14th International Conference

Joint Conference

SINTES 14 • SACCS 10 • SIMSIS 14

System Theory and Control

PROCEEDINGS

October 17-19, 2010
SINAIA - ROMANIA

Editura Universitaria Craiova

Proceedings
of the
14th International Conference on
System Theory and Control

(Joint conference of SINTES14, SACCS10, SIMSIS14)

October 17 – 19, 2010
Sinaia, ROMANIA

ISSN 2068-0465

Editor: Emil PETRE

Organizers:

- **University of Craiova, Faculty of Automation, Computers and Electronics, Automatic Control Research Center**
- **"Gheorghe Asachi" Technical University of Iasi, Faculty of Automatic Control and Computer Engineering**
- **"Dunarea de Jos" University of Galati, Faculty of Computer Science**
- **IPA CIFATT Craiova – Institute for Research and Engineering in Automation**

Technical co-sponsored by the IEEE - CSS
Control Systems Society

COMMITTEES

INTERNATIONAL PROGRAM COMMITTEE

Vladimir RĂSVAN — **General Chair**
Department of Automatic Control, University of Craiova, Romania

Chaouki T. ABDALLAH — **Program Chair**
University of New Mexico, United States

Silviu Iulian NICULESCU — **CSS Representative**
Laboratoire des signaux et systemes (L2S), Supelec, France

Dan POPESCU — **Vice-Chairman**
Department of Automatic Control, University of Craiova, Romania

Vasile MANTA — **Vice-Chairman**
Department of Computer Engineering,
"Gheorghe Asachi" Technical University of Iaşi, Romania

Adrian FILIPESCU — **Vice-Chairman**
Department of Automation and Industrial Informatics,
"Dunarea de Jos" University of Galaţi, Romania

MEMBERS

Mihail Abrudean (RO)
Tihamer Adam (HU)
Steve Banks (UK)
Viorel Barbu (RO)
Andrzej Bartoszewicz (PL)
Costin Badica (RO)
Theodor Borangiu (RO)
Pierre Borne (FR)
Paul van den Bosch (NL)
Sergiu Caraman (RO
Petru Cascaval (RO)
Emil Ceanga (RO)
Arben Cela (FR)
Ali Charara (FR)
Silviu Ciochina (RO)
Dorian Cojocaru (RO)
Vladimir Cretu (RO)
Valentin Cristea (RO)
Radu Dobrescu (RO)
Viorel Dugan (RO)
Ioan Dumitrache (RO)
Gerhard Freiling (DE)
Emilia Fridman (IL)
Cornelia Gordan (RO)
Eugen Iancu (RO)
Mircea Ivanescu (RO)
Robin De Keyser (BE)
Vladimir L. Kharitonov (RU)
Peter Kopacek (AT)
Karol Kostur (SK)
Gheorghe Lazea (RO)
Corneliu Lazar (RO)
Rogelio Lozano (FR)
Marin Lungu (RO)
Gheorghe Marian (RO)
Qinghao Meng (CH)
Viorel Minzu (RO)
Sabine Mondie (MX)
Sergiu Nedevschi (RO)
Roberto Oboe (IT)
Sorin Olaru (FR)
Nejat Olgac (USA)
Octavian Pastravanu (RO)
Emil Petre (RO)
Dumitru Popescu (RO)
Alexander Poznyak (MX)
Radu-Emil Precup (RO)
Stefan Preitl (RO)
Dorina Purcaru (RO)
Werner Purgathofer (AT)
Sergey Ryvkin (RU)
Bogdan Sapinski (PL)
Lubomir Smutny (CZ)
Nikos Tsourdveloudis (GR)
Nicolae Tapus (RO)
Erik I. Verriest (USA)
Gabriel Vladut (RO)
Mihail Voicu (RO)

NATIONAL ORGANIZING COMMITTEE

Eugen Bobasu (Chairman - RO)
Emil Petre (Program Editor - RO)
Adrian Burlacu (RO)
Daniela Cernega (RO)
Daniela Danciu (RO)
Augustin Ionescu (RO)
Marius Marian (RO)
Dorin Popescu (RO)
Elvira Popescu (RO)
Monica Roman (RO)
Dan Selisteanu (RO)
Dorin Sendrescu (RO)
Razvan Solea (RO)
Florina Ungureanu(RO)

WELCOME MESSAGE

The Conference whose Proceedings are in front of the reader is only apparently a new one. In fact it is a "bunch" of 3 traditional conferences or symposia in the field of Systems and Control which had been organised primarily around the eighties of the 20th Century in three different centres of Romania: Iaşi, Galaţi, Craiova. All these centres had in common prestigious Academic institutions as well as economic units with control and computer basics and applications as objects of activity.

For theses reason it had been decided at that time to organise specific National (at that time) Symposia, with well defined fields. We speak about the Symposium on Modelling, Simulation and Identification (SIMSIS) organized in Galaţi since 1978 and which had 13 editions, Symposium on System Theory (SINTES) organised in Craiova since 1980 and which also had 13 editions and Symposium on Automatic Control and Computer Science (SACCS) organised in Iaşi since 1987 and which had 9 editions.

Along some third of century all these meetings noticed a sensible growth in problem areas, number of participants per event and especially they acquired on international dimension in the last two decades.

Taking this into account and in order to increase the scientific and technical level of all Symposia, the decision has been taken for their merging into a larger International Conference called 14th Joint International Conference on System Theory and Control.

The conference is endorsed by all its traditional organisers and also enjoys the technical sponsorship of the IEEE Control System Society.

It is hoped that we succeeded to recover and to improve all features that each of the three Symposia had separately. Since only the participants and/or the readers of the papers may judge on the accomplishment we meet them with: WELCOME, distinguished colleagues!

LIST OF AUTHORS AND PAPERS

Output Analysis of Digital Interface Plants Based on FPGA Rapid Prototyping

Bogdan Alecsa and Alexandru Onea

Abstract— **This paper presents a method to analyze the output of digital interface plants by means of custom hardware implemented in FPGA and FPGA design and analysis tools. The novelty of the method is the use of a logic analyzer hardware core implemented together with the control hardware in an FPGA device to observe, extract and analyze the digital signals provided by the controlled plant. Experimental results present the method application to the case study of brushless DC motor control, using a digital output optical encoder for feedback.**

I. Introduction

FIELD programmable gate arrays (FPGA) have become powerful tools in prototyping, testing and even implementing highly demanding computational applications, which include control algorithms [1], [2]. They offer the advantage of configurability: the application has software-like programmability at hardware speed.

Also, FPGAs provide the possibility to develop and test custom hardware, tailored to the application, with any degree of parallelism. Unlike with processors and microcontrollers, where the hardware resources are limited and designed to serve general purposes, an FPGA provides enough hardware for hundreds of counters, timers, serial registers, address counters, all totally independent and fully customizable. One notable advantage of the custom designed hardware is the fact that the designer knows exactly what is needed and builds exactly that, which sometimes can be faster that learning about all the configuration registers and bits of general purpose hardware on a readily built processing system.

The main advantage FPGAs offer over the traditional processing systems is the hardware parallelism: different tasks can be executed in parallel and totally independent. This can lead to great performance advantages over usual embedded systems, based on microcontrollers or even digital signal processors (DSP). Adding to this the fact that modern FPGAs offer multiple embedded hardware multipliers or even DSP cells based on multiply-accumulate hardware [3], from tens to hundreds, and with low propagation times, makes them very suitable for computationally demanding control applications.

While these advantages are obvious and undeniable, the use of FPGAs in automatic control is still a matter of research [2], [3], [4]. This is due to the fact that FPGA design flow is different than usual embedded systems design flow, using different tools and being based on different knowledge. Even though FPGA producers and third parties offer a wide range of tools [5], [6] there is still need for a refined design methodology for automatic control.

This is the trend this paper follows: defining a methodology for FPGA use in automatic control. In this direction, this paper presents a method to extract and analyze digital signals provided by a controlled plant using FPGA hardware and FPGA design and analysis tools.

As a case study, a brushless DC (BLDC) motor, with a digital output optical encoder was chosen. Commutation hardware was designed and tested, but a method to determine the motor response was needed. This is presented in the next sections.

II. BLDC Speed Measurement

The plant used as a case study in this paper is a BLDC motor. The motor is equipped with 3 Hall effect sensors and an optical encoder. While the Hall effect sensors provide rotor position information sufficient for determining the commutation sequence of the stator windings, this information is not accurate enough for a position or speed feedback loop. Because of this, the motor is equipped with a 500 lines optical encoder, providing 2 quadrature encoded output signals. These signals can be used to determine the speed and direction of rotation of the rotor. Also, incremental rotor position can be determined.

The main problem in controlling the motor consists in determining the controller parameters. The major difficulty in verifying the controller is observing the output of the motor, which is the rotation speed.

There are several ways to observe the motor speed:

1) Connect the motor to a generator and read the voltage at the generator output. The advantage of the method is that it can provide a graphical view of the speed variation. However, it has two major disadvantages:

Manuscript received May 7, 2010. This work was partially supported by Continental Automotive Romania S.R.L. and by the National Centre for Programs Management from Romania under the research grant SICONA – 12100/2008.

B. Alecsa is with the Automatic Control and Applied Informatics Department, Technical University "Gh. Asachi", Iasi, Romania (phone: +40-721602160; fax: +40-232-230751; e-mail: balecsa@ac.tuiasi.ro).

A. Onea is with the Automatic Control and Applied Informatics Department, Technical University "Gh. Asachi", Iasi, Romania (e-mail: aonea@ac.tuiasi.ro).

the generator output presents voltage ripple which should be filtered out, and the generator itself is a load for the motor, modifying the output characteristics.

2) Observe the frequency of the quadrature encoded signals of the optical encoder. While this method is non-invasive, not affecting the plant transfer function, it has the disadvantage that it does not provide a graphical view of the speed variation. Additional hardware, namely a frequency to voltage converter, should be used for this.
3) Digital circuitry can be used to count the fronts of the quadrature encoded signals. The counter can be read and reset at a fixed interval and the value read from the counter can provide information about the motor speed. While this method is preferred, as this circuitry is used also in the control loop, the disadvantage from the previous method remains: it is difficult to obtain a graphical view of the speed variation.

This last method was improved and applied together with a specific FPGA design and debug method to obtain the desired graphical representation.

III. Method for Obtaining the BLDC Motor Speed Characteristic Graphic Representation

A. Hardware module for BLDC motor commutation

Before the motor speed could be estimated, a hardware module for generating the commutation sequence was designed and implemented in the FPGA.

The reader should resort to existing literature for information on driving methods for BLDC motors [7]. For the experiments presented in this paper, a three phase inverter bridge with MOSFET transistors was used. The commutation module must activate the right transistors in the inverter bridge in order to energize one stator winding with the right polarity. The transistor pair is chosen based on the current rotor position, which is determined by using 3 Hall sensors built in the motor. Additionally to the 3 input from the Hall sensors, a validation input is provided. So, the commutation module has 4 inputs and 6 outputs. Each output represents the state on 1 transistor in the inverter bridge.

The commutation logic is implemented by 6 read only memory (ROM) cells of 16 bits each. The 3 Hall sensors signals and the validation input are the address bits of the ROM cells. The content of the ROMs represent the state of the inverter bridge transistors associated with the given rotor position (each ROM cell corresponds to 1 transistor). The order of the address inputs is (from least to most significant bit): HALL_1, HALL_2, HALL_3, VALID. The ROM contents are presented in Table I.

The advantage of the FPGA implementation over traditional microcontroller implementation is obvious in this case: there is no delay in the sequence generation (only the ROM propagation delay), but, more importantly, the commutation module is absolutely independent. A microcontroller implementation would need an interrupt on change of the Hall signals and a software routine to compute the next output value. This would mean some delay, but this would also mean that the main software program stops execution in response to the interrupt. In an FPGA, while the commutation module changes the output, other modules can perform other tasks, without interruption.

TABLE I
Commutation Module ROM Contents

ROM Cell	ROM Contents
ROM_Q1	0x000A
ROM_Q2	0x0050
ROM_Q3	0x0044
ROM_Q4	0x0022
ROM_Q5	0x0030
ROM_Q6	0x000C

In order to avoid overcurrents because of the transistors switching time, a dead band is inserted at each commutation change. This means a delay between the switch off of the previously activated transistor and the switch on of the next transistor, if they are on the same inverter side (high side or low side).

The dead band generating module is also independent. It controls the validation input of the commutation hardware, effectively controlling the ROM stored value which will be outputted. It consists of a digital delay, implemented by a 5 bit counter. For the experimental setup, a dead band of at least 0.5μs was necessary. The counter, fed by the 50MHz system clock, gives a dead band of 0.64μs. The counter is reset when a change in the Hall signals is detected, starting the dead band measurement. At the same time, the VALID signal is deactivated. When the counter overflows, the dead time has ended, and the VALID signal is activated. This scheme is implemented with only one flip-flop controlled by the counter overflow signal.

B. BLDC motor speed estimation circuitry

The speed estimation circuitry block diagram is presented in Fig. 1. For better resolution of the speed measurement, both rising and falling edges of both quadrature encoded pulse (QEP) signals are counted. An edge detection circuitry is used for each edge. The edge detection circuitry is a simple Mealy finite state machine (FSM), implemented with only one D type flip-flop (DFF), as presented in [8]. Each edge detection circuit output activates when the monitored input and the value stored in the register, which is the input value at the previous clock rising edge, are in the desired sequence (0-1 or 1-0). All outputs are ored and connected to the clock enable input of a binary counter. This way, the edges of the QEP signal are

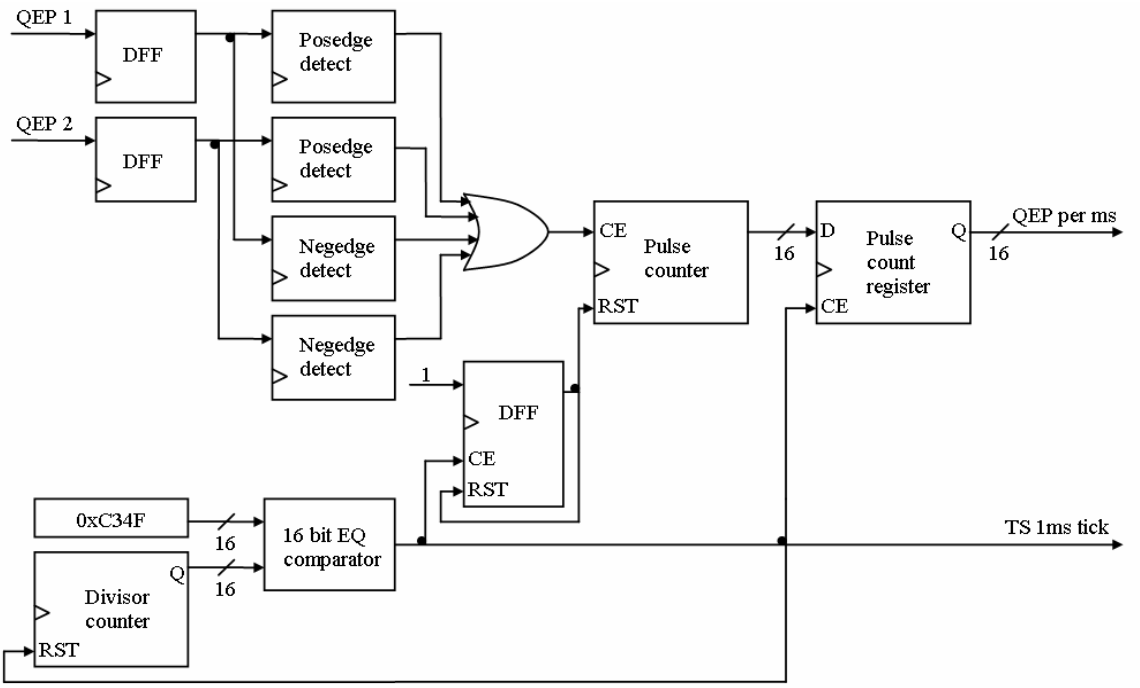


Fig. 1. Block diagram of speed estimation circuitry.

counted by incrementing the binary counter at each edge detection pulse.

Because the QEP signals are coming from an external, unsynchronized source, the edge of the QEP signal could come very close to the clock signal rising edge. The QEP edge is detected, meaning the output of the edge detection circuitry activates, but the output pulse propagation to the clock enable input of the counter could be longer than the difference between the QEP signal edge and the clock signal rising edge. This hazard could lead to the counter missing increments, even though the edge detection circuitry works properly. To address this problem, the QEP inputs were synchronized by adding DFFs to the signal paths.

This way, all signals used in the design are synchronized to a single clock signal. The clock signal is generated inside the FPGA from an external 50MHz quartz oscillator. The fact that all circuitry uses the same clock signal is an advantage for the FPGA design: the FPGA internal structure provides special global routing resources for the clock signal, ensuring the lowest possible clock skew [5]. In order that these resources are allocated properly by the FPGA implementation and routing software, the design must be properly done.

In Fig. 1, all the flip-flops and registers are synchronous and use the same 50MHz clock. The clock signal is not shown on the figure, in order to make it easier to read.

The contents of the pulse counter are saved into a register at every sample period. The sample period is obtained by dividing the clock frequency by a suitable proportion.

For a clock frequency

$$f_{clock} = 50\,MHz\ , \tag{1}$$

meaning a clock period

$$T_{clock} = 20\,ns\ , \tag{2}$$

and a desired sample period

$$T_s = 1ms\ , \tag{3}$$

the divisor ratio is

$$DIV\_CONST = T_s / T_{clock} = 50000\ . \tag{4}$$

The clock divisor is implemented by a 16 bit binary counter, fed by the 50MHz clock. When the output reaches the desired combination, detected by a 16 bit equality comparator (16 XOR gates ored together), the counter is reset and the subsequent circuits receive a one clock period *TS* signal. The sample time generation circuit is, in fact, a FSM with the number of states set by the comparison constant and the output active only in the last state. Taking into account the first state is 0, the comparison constant is, in fact, given by:

$$COMP\_CONST = DIV\_CONST - 1 = 0\text{xC34F} . \quad (5)$$

At the same time when the pulse counter value is saved in the register, the counter should be reset. To keep the outputs stable, the reset is delayed by one clock period. In this respect, a DFF is used: a 1 is loaded into the DFF when the *TS* signal is activated. The DFF output is used to reset the pulse counter, and, at the same time, the DFF itself. This way, the DFF repeats the *TS* pulse, but one clock cycle later.

All register and FF resets are synchronous with the system clock.

C. Logic analyzer software and core usage for FPGA internal signals monitoring

The design of hardware control structures in FPGA devices has a great advantage over traditional embedded systems using microcontrollers: the unused FPGA logic can be used to implement additional hardware for debugging. The simplest example is replicating the internal signals of the design on unused I/O pins for oscilloscope analysis.

A more advanced way of debugging FPGA hardware is the usage of a "virtual" logic analyzer to capture the internal signals of the design. The logic analyzer uses the FPGA internal memory to store samples of selected signals. The capture length is defined by the user, as well as the trigger condition that starts the capture.

The contents of the memory is read via a JTAG (Joint Test Action Group) connection by special software and displayed on a host computer.

The "virtual" logic analyzer is available as an intellectual property (IP) core from FPGA producers (like Xilinx, Altera and Lattice) or development tools producers (for example, Altium, Synplicity, First Silicon Solutions) [5].

For the experiments, a Spartan-3E FPGA device produced by Xilinx was used, and the ChipScope Pro logic analyzer, also developed by Xilinx. The design was done at the schematic capture level. In the design, the logic analyzer is inserted as two cores: the ILA (Integrated Logic Analyzer) core and the ICON (Integrated Controller) core. The ICON core is used to connect to the host computer via JTAG interface and can control up to 15 ILA cores. The ILA core defines the signals to be monitored, the number of samples taken at a time and the logic used to implement the trigger. The trigger condition can be specified at the debug time, through the host computer user interface provided by ChipScope Pro.

The ILA core is synchronous. The clock signal used to drive the ILA core specifies the period at which samples are taken. So, unlike a true logic analyzer, the "virtual" analyzer can not observe glitches that are shorter than the clock period. Through a convenient setting of the clock period different characteristics of the signals can be observed.

Fig. 2 presents a capture of the signals from the speed estimation circuitry. The motor is running at constant speed, so the QEP signals have constant period. The ILA core clock is the 50MHz system clock.

It can be observed that every edge on any of the QEP signals produces a pulse on the CE input of the counter, and the counter increments.

When the TS signal is asserted, the current counter value is saved to the register, and a new counting begins, by resetting the counter.

D. Obtaining the graphic representation of the motor speed characteristic

The motor speed can be determined form the value stored in the speed estimation register. Taking into account that the optical encoder wheel has a number of lines

$$N = 500 , \quad (6)$$

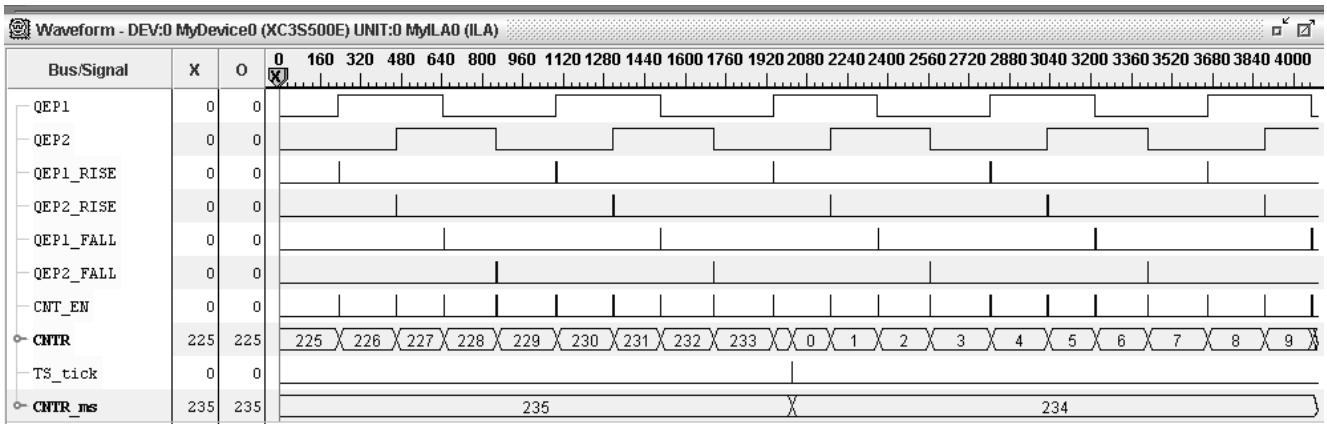


Fig. 2. Speed estimation circuit signals.

and for each line the QEP counter is incremented 4 times (4 QEP edges):

$$N_{edge} = 4\,, \tag{7}$$

it follows that to obtain the motor speed in rotations per minute (RPM) the next relation is needed:

$$S_{RPM} = QEPC \cdot \frac{60 \cdot 1000}{N \cdot N_{edge}} = QEPC \cdot 30, \tag{8}$$

where *QEPC* is the pulses count, given by

$$QEPC = CNTR\_VAL + 1\,, \tag{9}$$

because the counter starts from 0.

The motor used in the experiments is a 19.1*V* Pittman 3411 series BLDC servo motor, produced by Ametek. The motor characteristics are presented in Table I. From the table, it can be depicted that the motor has a no load rated speed of

$$S_{nl} = 7150\ RPM\,, \tag{10}$$

and a mechanical time constant

$$\tau_m = 8.3ms. \tag{11}$$

For the graphic representation deriving, the ILA core clock period was set to 1*ms*. This way, at each sample, only the *CNTR_ms* signal has a meaning.

The motor supply was cut and the ChipScope trigger was set for magnitude comparison of *CNTR_ms* to a low value. When the motor was supplied again, the *CNTR_ms* signal depicted the speed response of the motor to a step excitation.

The signals viewed in ChipScope were exported to a file as ASCII data. The *CNTR_ms* signal was imported from the file to the Matlab environment, incremented as stated in (9) and multiplied by 30 according to (8), and then plotted against the sample number.

Fig. 3 shows the response as plotted in Matlab. It can be observed that the motor speed rises to around 7100*RPM* in about 30*ms*, with a maximum slope of around 7100*RPM*/8*ms*. This is in accordance with the motor technical data of (10) and (11).

IV. Conclusion

In this paper, a method to extract process information in a useful format from digital interface plants was presented. The novelty of the method is the use of custom hardware implemented in an FPGA and FPGA design and debugging tools to capture data and analyze it on a host computer.

TABLE II
19.1*V* Pittman 3411 Motor Characteristics

Motor Data	Symbol	Units	Value
Supply Voltage (reference)	V_S	V	19.1
Continuous Torque	T_c	Nm	0.029
Motor Constant	K_M	Nm/$\sqrt{\text{W}}$	0.011
Torque Constant	K_T	Nm/A	0.0250
Voltage Constant	K_E	V/kRPM	2.62
Terminal Resistance	R_{mt}	Ω	5.25
Inductance	L	mH	0.46
No-Load Speed	S_{nl}	RPM	7150
Electrical Time Constant	τ_e	ms	0.09
Mechanical Time Constant	τ_m	ms	8.3
Rotor Inertia	J_r	$\text{kg} \cdot \text{m}^2$	9.9E-07

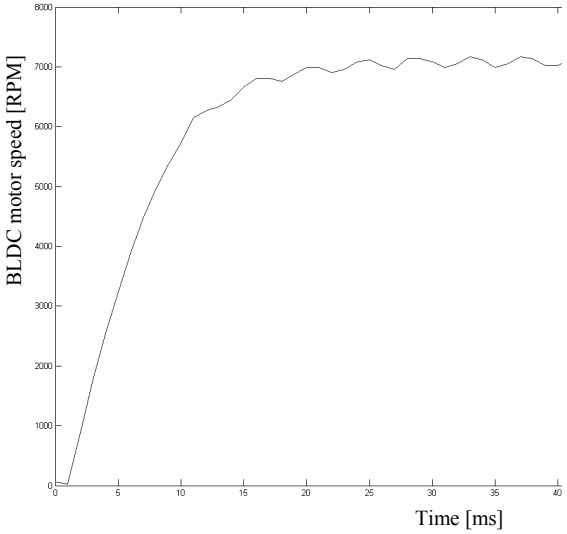


Fig. 3. BLDC motor speed response to input step voltage

The capture hardware was designed, implemented and tested for the case study of speed response analysis of a BLDC motor equipped with an optical encoder. The QEP outputs of the encoder are counted and the counter value is strobed every 1*ms*.

The ChipScope Pro "virtual" logic analyzer was used to capture the counter values for some milliseconds after a step voltage is applied to the motor. The captured data was exported as an ASCII file, imported in Matlab and displayed.

While the use of the ChipScope Pro software is well established in the literature for debugging and analysis of digital systems [9], [10], the originality of this paper is the usage of such tools for automatic control data analysis.

Also, the digital processing of motor resolver signals

using FPGAs is a matter of research [11], [12].

The described process can constitute a methodology for output analysis of any digital output plant, and this is the main result of the paper. It puts the basis for a new direction of FPGA rapid prototyping usage, in a domain where microcontroller based embedded systems have a long tradition.

Due to the multitude of resources that the FPGAs offer nowadays, they can be used further to implement the entire control system. In this respect, Matlab Simulink can be used as a powerful design and validation tool [13].

BLDC motor control using an FPGA device has been proposed with a digital model of the motor and a simple 2 states PWM technique [14]. A classical control algorithm may be more appropriate and will be a future development of this paper.

REFERENCES

[1] C. Maxfield, *FPGAs: World Class Designs*, Newnes Elsevier, 2009.

[2] E. Monmasson, M. N. Cirstea, "FPGA Design Methodology for Industrial Control Systems - A Review", *IEEE Transactions on Industrial Electronics*, vol. 54, no. 4, August 2007.

[3] M.-W. Naouar, E. Monmasson, A. A. Naassani, I. Slama-Belkhodja, N. Patin, "FPGA-Based Current Controllers for AC Machine Drives – A Review", *IEEE Transactions on Industrial Electronics*, vol. 54, no. 4, August 2007.

[4] A. Molino, M. Martina, F. Vacca, G. Masera, A. Terreno, G. Pasquettaz, G. D'Angelo, "FPGA Implementation of Time-Frequency Analysis Algorithms for Laser Welding Monitoring", *Microprocessors and Microsystems*, vol. 33, no. 3, May 2009.

[5] R. Woods, J. McAllister, G. Lightbody, Y. Yi, *FPGA-Based Implementation of Signal Processing Systems*, John Wiley and Sons, 2008.

[6] J. J. Rodriguez-Andina, M. J. Moure, M. D. Valdes, "Features, Design Tools, and Application Domains of FPGAs", *IEEE Transactions on Industrial Electronics*, Vol. 54, No. 4, 2007.

[7] A. Emadi, *Handbook of Automotive Power Electronics and Motor Drives*, CRC Press, 2005.

[8] P. P. Chu, *FPGA Prototyping by Verilog Examples*, John Wiley and Sons, 2008.

[9] O. Oltu, P. L. Milea, A. Simion, "Testing of Digital Circuitry Using Xilinx Chipscope Logic Analyzer", *Proceedings of CAS 2005 International Semiconductor Conference*, 2005.

[10] K. Arshak, E. Jafer, C. Ibala, "Testing FPGA Based Digital System Using XILINX ChipScope Logic Analyzer", *Proceedings of 29th International Spring Seminar on Electronics Technology*, 2006.

[11] K. Bouallaga, L. Idkhajine, A. Prata, E. Monmasson, "Demodulation Methods on Fully FPGA-Based System for Resolver Signals Treatment", *Proceedings of European Conference on Power Electronics and Applications*, 2007.

[12] L. Idkhajine, E. Monmasson, M. W. Naouar, A. Prata, K. Bouallaga, "Fully Integrated FPGA-Based Controller for Synchronous Motor Drive", *IEEE Transactions on Industrial Electronics*, vol. 56, no. 10, October 2009.

[13] B. Alecsa, A. Onea, "An FPGA Implementation of the Time Domain Deadbeat Algorithm for Control Applications", *Proceedings of 2009 Norchip*, Trondheim, Norway, 2009.

[14] A. Sathyan, N. Milivojevic, Y.-J. Lee, M. Krishnamurthy, A. Emadi, "An FPGA-based novel digital PWM control scheme for BLDC motor drives", *IEEE Transactions on Industrial Electronics*, vol. 56, no. 8, August 2009.

A FPGA Implementation of an Active Noise Cancellation System

Bogdan Alecsa, Bogdan Radu, Alexandru Onea, *Member, IEEE,* and Alexandru Bârleanu

***Abstract*— The paper presents an implementation of a single channel active noise cancellation (ANC) system. The focus was placed on the study of a methodology to use a field programmable gate array (FPGA) circuit for the development of adaptive filters, working at high sampling rates. The design environment is the LabView FPGA software because it provides a very intuitive and powerful graphical design interface. Experimental results show that it is possible to implement relatively long adaptive filters working in parallel, and the effectiveness of the ANC working at high sample frequencies. The implementation of the ANC with a FPGA shows that the FPGA is suitable for high speed digital signal processing (DSP) applications.**

I. Introduction

The acoustic noise sources, like engines, blowers, fans, compressors, are increasing in number in our environment, thus resulting in an ever higher need for systems that can reduce the noise.

There are two types of noise that exist in the environment. One is caused by turbulence and is random; this is called broadband noise because it is evenly distributed in the audio spectrum. The second type has its energy concentrated only around some frequencies and is called narrowband noise. The last one is typically almost periodic and is produced by rotating machines, frequently present in industry.

The approaches of noise control can be passive, that use sound barriers, and active, that consist of an electroacoustic device that cancels the unwanted noise by generating the same noise with opposite phase (an anti-noise). Passive noise cancellers are effective over a broad frequency spectrum, but they become relatively large and expensive if they are designed to attenuate low frequencies. For low frequencies the ANC system remains the best choice. The majority of ANC systems are adaptive, meaning that they can adapt to changes in the environment of the sound field that has to be cancelled.

Manuscript received on May 15, 2010. This work was partially supported by Continental Automotive Romania S.R.L. and by the National Centre for Programs Management from Romania under the research grant SICONA – 12100/2008.

B. Alecsa is with the Department of Automatic Control and Applied Informatics, Technical University "Gh. Asachi" of Iaşi, IS 700050, Romania (phone: 0721-602-160; fax: +40 232 230751, e-mail: balecsa@ac.tuiasi.ro).

B. Radu is with the Department of Automatic Control and Applied Informatics, Technical University "Gh. Asachi" of Iaşi, IS 700050, Romania (e-mail:neoradu@yahoo.com).

A. Onea is with the Department of Automatic Control and Applied Informatics, Technical University "Gh. Asachi" of Iaşi, IS 700050, Romania (e-mail: aonea@ac.tuiasi.ro).

A. Bârleanu is with the Department of Computer Engineering, Technical University "Gh. Asachi" of Iaşi, IS 700050, Romania (e-mail: alexb@cs.tuiasi.ro).

A specific application for an ANC system is the reduction of unwanted noise in a duct, where a single channel controller is used. A second usage refers to attenuation of noise in a local zone of a room. In this case a multichannel controller is mandatory due to the complex sound field.

Mainly, there are two types of controllers: some use only the residual noise for generating the anti-noise; the others use the initial noise as well. The first category is called feedback ANC system [1] and the second, feedforward ANC system [1].

ANC systems based on adaptive filter theory were developed as early as 1980s, but the technology has become practical with the introduction of low cost high speed digital signal processors (DSP) [2].

Development of such systems was realized on floating point DSP and results are presented in [1], [3], and [4]. The advances that were made in FPGA technology are making highly tempting the usage of FPGAs in ANC systems. Modern FPGAs offer embedded hardware multipliers and even DSP cells based on multiply-accumulate hardware, as well as blocks of random access memory (RAM) and hundreds of thousands of equivalent logic cells, which makes them very appropriate for DSP applications [5].

Also, the design tools have evolved to offer a high level of abstraction, making it easy for the designer to address the problem at the system level [5].

This paper presents an ANC system implementation on a FPGA board.

II. Single Channel Feedforward ANC System

The basic structure of a single channel feedforward ANC system is presented in Fig.1.

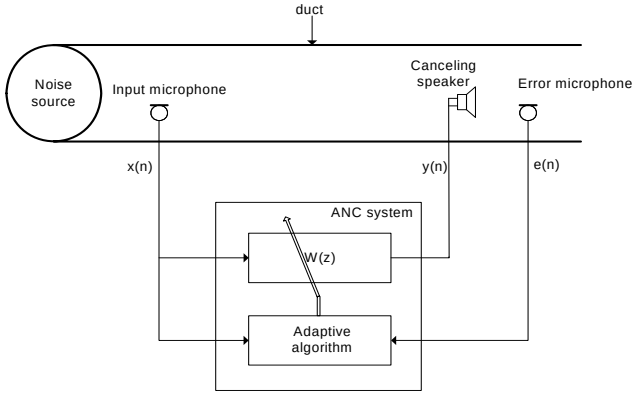


Fig.1. Basic structure of ANC system

A reference signal $x(n)$ is sensed by the input microphone that is placed near the noise source. This is used by the

controller to generate the control signal $y(n)$. The error signal $e(n)$ is sensed by the error microphone and is used by the adaptive algorithm to adjust the coefficients of the filter $W(z)$. This way the error is minimized and the filter models the acoustic path from the input microphone to the error microphone.

A more detailed block diagram is presented in Fig. 2. Here the controller can be viewed in a system identification framework. If we consider $C(z)=1$, $\hat{C}(z)=1$, $F(z)=0$ and $\hat{F}(z)=0$, the unknown acoustic path $P(z)$ is estimated by the adaptive filter $W(z)$ in order to minimize the error signal $e(n)$ .In the z domain, this leads to:

$$E(z)=D(z)+Y(z) = X(z)\,[P(z)+W(z)]\,, \tag{1}$$

where $E(z)$ is the error signal and $Y(z)$ is the adaptive filter output. After the adaptive filter has converged the error becomes zero. That implies:

$$P(z)=-W(z). \tag{2}$$

As it can be seen from Fig.1, the error signal is recorded downstream from the canceling speaker. The error signal is modeled in Fig. 2 by the secondary transfer path $C(z)$.

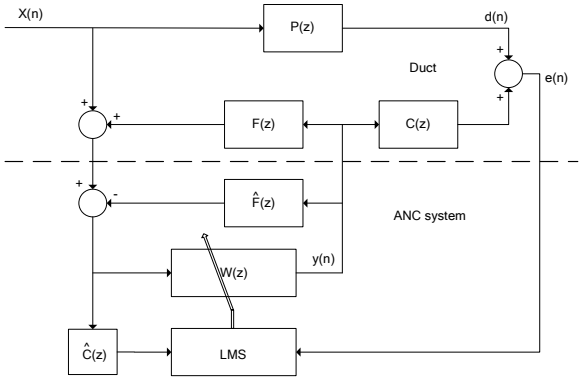


Fig.2. Detailed representation of ANC system

The existence of this acoustic path implies that $E(z)$ is expressed as:

$$E(z)=X(z)\,[P(z)+C(z)W(z)] \tag{3}$$

Assuming that the error is zero after convergence, the adaptive filter has to realize the transfer function:

$$W(z)=\frac{-P(z)}{C(z)} \tag{4}$$

If the least mean squares (LMS) [6] adaptive algorithm is used, the presence of $C(z)$ leads to instability. This happens because of the impossibility to invert the delay caused by $C(z)$, if the primary path $P(z)$ does not contain a delay at least equal in length. The solution to this problem is to use a modified form of the LMS algorithm, the filtered-X LMS (FXLMS) [7].

In the standard LMS algorithm the output $y(n)$ is given by the following equation:

$$y(n)=w(n)^T x(n)=\sum_{i=1}^{N} w_i(n)x(n-i)\,. \tag{5}$$

Equation (5) represents a finite impulse response (FIR) filter of length N, where the output $y(n)$ is a linear combination of present and past inputs and the filter weighs $w(n)$.

The cost function that is used in the LMS algorithm is the mean square error:

$$\xi(n)=\frac{1}{2}e^2(n)\,. \tag{6}$$

The desired goal is to minimize the instantaneous square error $e^2(n)$ [8].This is achieved by

$$w(n+1)=w(n)-\frac{\mu}{2}\nabla e^2(n)\,. \tag{7}$$

Equation (7) is equivalent with

$$w(n+1)=w(n)+\mu e(n)x(n) \tag{8}$$

In (8), μ is the adaptation rate, and dictates the speed of convergence. A big value of this parameter can make the algorithm to diverge. A detailed study of the stability is available in [6].

To solve the problem that resulted from (4) with the FXLMS algorithm, the signal $x(n)$ used in (8) is filtered through an estimate of $C(z)$, $\hat{C}(z)$, as it can be seen from Fig. 2. Thus, (8) becomes

$$w(n+1)=w(n)+\mu e(n)x'(n)\,, \tag{9}$$

where $\hat{C}(z)$ is a FIR filter of length N, and so $x'(n)$ is:

$$x'(n)=\sum_{i=1}^{N}\hat{c}_i(n)x(n-i)\,. \tag{10}$$

Because the signal $e(n)$ is obtained as a sum of two sound waves, not electrical subtraction, in order to obtain the minimum of the cost function (6), (9) becomes:

$$w(n+1) = w(n) - \mu e(n) x'(n). \quad (11)$$

From Fig. 1 it can be observed that there is also an acoustic transfer path from the canceling speaker to the input microphone. If the gain of this path is consistent, the positive reaction formed can lead to instability. The solution is to use an estimate of $F(z)$, $\hat{F}(z)$, to subtract the filtered $y(n)$ from the input signal. This is shown in Fig. 2.

Both transfer ways $C(z)$ and $F(z)$ are unknown and time varying in practice, due to the aging of the speaker or to the temperature variation and they need to be estimated. This can be done on-line or off-line during a training stage. Both transfer paths can be estimated at the same time using the standard LMS algorithm, with a setup like in Fig. 3.

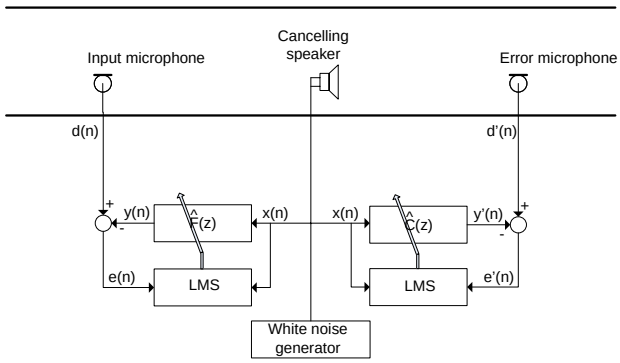


Fig.3. Identification setup

Here $d(n)$ and $d'(n)$ represent the desired responses for $\hat{F}(z)$ and $\hat{C}(z)$ filters. The resulted error $e(n)$ and $e'(n)$ is used by the LMS algorithm, according to (8), to update the filters coefficients.

III. Experimental Results

The ANC system was implemented on the Spartan-3E Starter Kit board. This board has a Xilinx XC3S500E Spartan-3E FPGA, a four-channel digital-to-analog converter (DAC), a two-channel analog-to-digital converter (ADC) with a programmable-gain pre-amplifier. The DAC, the ADC and the amplifier use a serial SPI interface. These features make this board a good choice for applications that involve digital signal processing.

The design was realized in LabView FPGA 2009. This software allows the usage of the LabView intuitive graphical design interface to implement very fast and truly parallel hardware modules on FPGA. The LabView Adaptive Filter Toolkit can automatically generate the hardware graphical description of a fixed point LMS adaptive filter. The generated description can be easily modified to fulfill the designer needs.

For the ANC system presented three adaptive filters were generated. Fixed point format was used throughout the entire project, LabView FPGA 2009 offering good support for this.

A signed fixed point format with a word width of 16 bits and one bit for the integer part was chosen for the signal lines between filters, because the ADC has 14 bit resolution.

The filter coefficients are words of 24 bit width and one bit for the integer part for a better resolution.

All three adaptive filters were generated using LabView Adaptive Filter Toolkit and then modified. The filters used at the identification of the secondary transfer paths have the possibility of modifying on-line the number of taps. A hardware module was added to copy the filter coefficients to an external video buffer when an external trigger signal is received. This last added feature is used in conjunction with a video block that generates VGA signals. This gives the possibility to view the impulse response of the adaptive filter directly on a computer monitor.

The main filter has the possibility to modify its length and a separate memory was added for it. The extra memory was used to store filtered input $x'(n)$ used by the FXLMS algorithm.

The LabView diagram of the main filter is shown in Fig. 4. The filter is composed of two loops. The first one is executed only once, and is used to write the inputs $x(n)$ and $x'(n)$ to their designated memories. The address index of the current sample is generated also here. This is passed to the second loop, where it is used by the controller.

The second loop is executed $2*N+8$ times, where N is the filter length. First the filter output is calculated and then the weighs are updated using the FXLMS algorithm. The structure of the adaptive filter can be better visualized if it is represented as blocks, as in Fig. 4.

Block 1 represents the memories for the input signals: the coefficients and the filtered input. Block 2 and block 3 can be viewed together as a multiply-accumulate (MAC) unit, where the first one is the multiplier and the second one is the accumulator. The MAC unit is used to calculate, in the first phase, the filter output according to (5).

The block 4 has two multipliers and one subtracter that implement (11). In the block 5 overflow of the new weights resulted in previous block is checked and an external boolean signal is set if overflow is detected.

There are also three case structures in Fig. 4. In case 1, the external variable that triggers the refresh of the video buffer is cleared. In case 2 the new filter coefficients that result from block 5 are written in memory. There is also the case 3 structure, which copies the filter coefficients to the video buffer when the external trigger is received.

An offset is added to the coefficients before they are written to the video buffer. This is $-1*\min(w)$ and scales the values to fit the weighs on a computer monitor with a resolution of 640x480 pixels. The minimum is calculated when no trigger signal is present, in the case 3 structure.

The z^{-1} operators in the second loop help to reduce the maximum propagation time. In this way a 20ns execution time could be achieved. They are essentially pipelining registers.

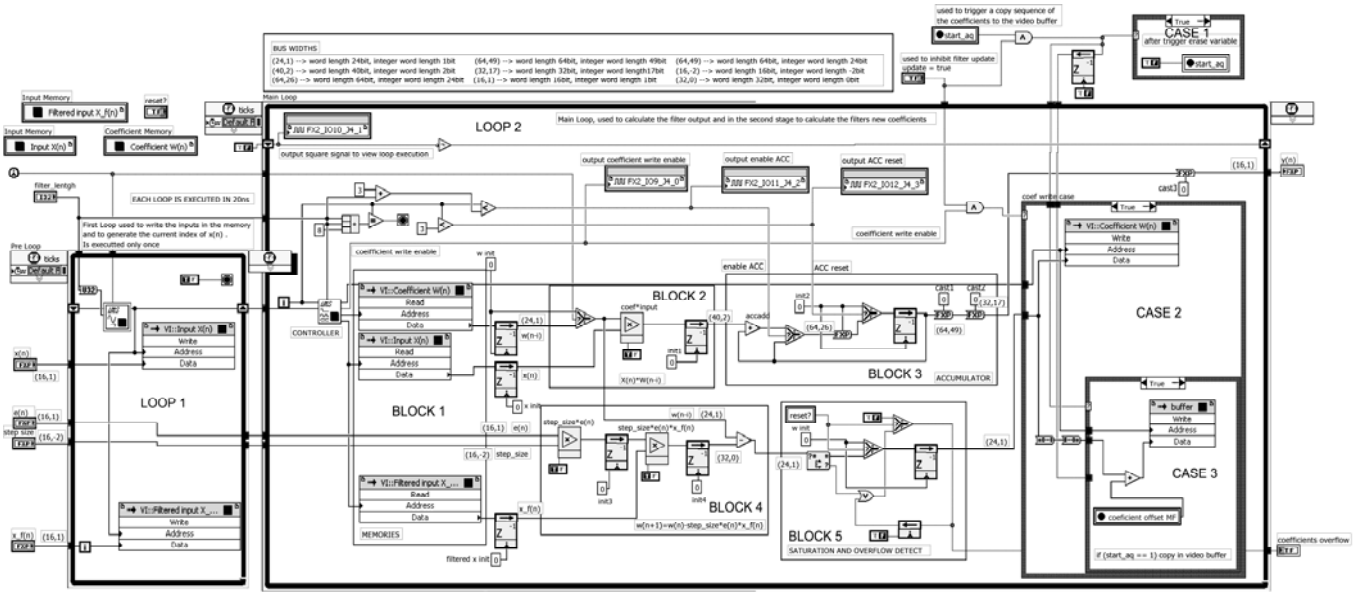


Fig.4. Main filter LabView diagram

The whole structure is a finite state machine with data path (FSMD) [9]. The states are generated by the controller, and from here the control signals are generated. Data paths are in fixed point format and have different widths that are stated at the top of Fig. 4.For a better understanding of how the filter works, 4 signals where routed to FPGA pins and outputted on a port of the board to be monitored on an oscilloscope. They can be viewed in Fig. 5. The capture was taken for a filter with 6 taps and a correspondence between channels in Fig. 5 and signals in Fig. 4 is given in table 1.

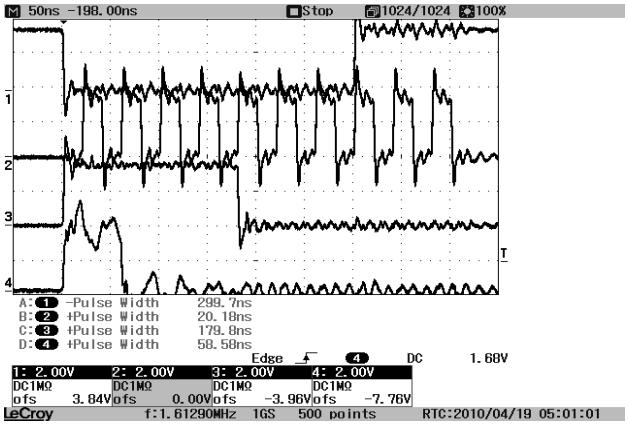


Fig.5. Main filter signals

The system has two main modes of operation. The first mode is the training stage, in which a white noise generator is fed to the canceling speaker, and the identification of the $C(z)$ and $F(z)$ transfer paths (Fig. 3) is done with the standard LMS algorithm. The filter impulse response to be viewed on the monitor can be also selected. This feature helps because it gives a feedback about the evolution of the filter and it can be seen clearly when the filter coefficients have reached steady-state.

The second mode starts when the adaptation of the secondary filters is stopped and these are used in the FXLMS algorithm with the main filter.

To take full advantage of the FPGA platform the acquisition of the analog signals and all three filters are executed in parallel. Because the DAC is on the same SPI bus as the ADC, the communication with the DAC is realized after all the processing of one sample has taken place. To speed up communication time with the DAC a SPI clock of 25 MHz was chosen.

TABLE I
CORRESPONDENCE BETWEEN OSCILLOSCOPE CHANNELS IN FIG. 5 AND SIGNALS IN FIG. 4

Fig. 5 channel	Fig. 4 signal
Channel 1	FX2_IO9_J4_0
Channel 2	FX2_IO10_J4_1
Channel 3	FX2_IO11_J4_2
Channel 4	FX2_IO12_J4_3

A better view of the execution times is obtained by using four signals from an expansion connector. A detailed oscilloscope capture is presented in Fig. 6.

The timing presented in Fig. 6 shows signals from the 3 filters. The filters used to estimate the secondary transfer paths have a length of 500 taps and the main filter has a length of 640 taps. The maximum sampling rate that can be achieved is 36.4 kHz. This is presented in the figure by the Δt between cursors. Channel 1 is pulled high when the ADC conversion has completed and it is pulled low after the communication with the DAC has completed. Channel 2 represents the main filter execution time, and subtracting this time from the total sample time results that sending the output to the DAC takes 1.64 µs. Channels 3 and 4 are representing the execution times of the filters used to model the secondary transfer paths.

The setup for the ANC system was built as a duct with the diameter 180 mm and 3m long, covered at one end. The

noise source was provided from a speaker placed at the covered end. The cancellation speaker was placed at the opposite end of the noise source, near to the error microphone.

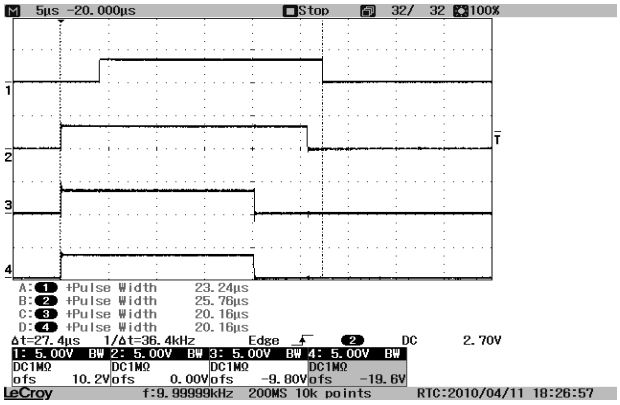


Fig.6. Filters execution times

Although the maximum sampling rate that could be achieved is 36.4 kHz, a 10 kHz sample rate was chosen to maintain the filter length at a reasonable value. For the anti-aliasing and the anti-imaging filters, one pole low-pass filters with a pass band of 4 kHz were used.

Secondary transfer paths were identified firstly. The experiments show that a filter of 500 taps is needed for the estimation of $F(z)$. This filter models very well the amplitude and phase characteristics of the real system. The model is used to cancel the influence of the control signal on the input signal and a bad model may well lead to instability of the whole system. An adaptation step of 0.025 was used. This allows the filter coefficients to become large enough to correctly model the gain of the transfer path.

At the same time with the white noise used as input signal for identification, the error signal can also be monitored on the oscilloscope because there are four DAC channels available on the development board. The oscilloscope capture is presented in Fig. 7. The channel 1 shows the moment when the training begins, and the channel 4 shows the evolution of the error signal.

The acoustic path from the canceling speaker to the error microphone was modeled using a filter of 120 taps. Because of the high gain due to the short length of this path a 0.0005 adaptation step was used. It was demonstrated that the FXLMS algorithm is converging if the phase error of the model is lower than 90° [10].

The system performance was evaluated using three noise signals, each of them composed of five sinusoid signals. The experimental results are presented in Table 2.

TABLE II
EXPERIMENTAL ATTENUATION MEASUREMENT IN THREE CASES

Nr.	Frequency components of noise [Hz]					Attenuation achieved [dBm]
1	290	310	330	340	360	32.9
2	240	260	280	300	310	27.5
3	330	350	380	420	430	32.1

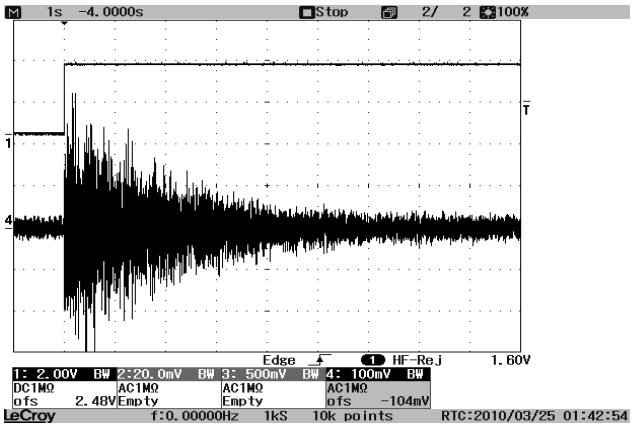


Fig.7. Evolution of the identification error

The noise was generated using a PC, MATLAB Simulink and Signal Processing Blockset. The noise was taken directly from the error microphone and the measurements were done using the LeCroy WaveJet 334 oscilloscope. The FFT operator was used together with a VonHann window function and screen captures are presented in Fig. 8, 9, 10, 11, 12, and 13. Fig. 8 represents the noise spectrum resulted at the error microphone for the first noise test.

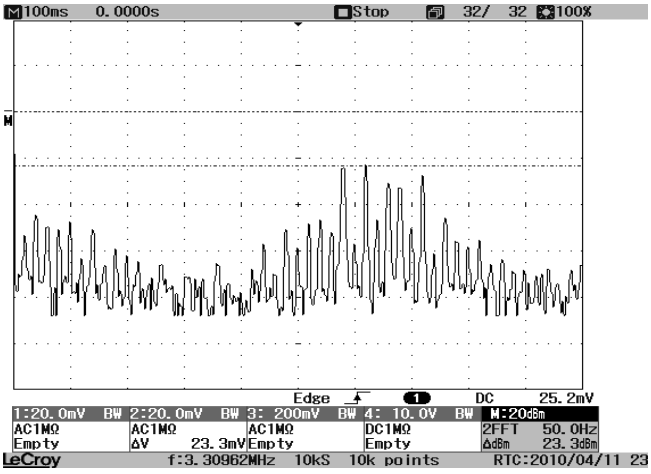


Fig.8. PSD of the first test noise

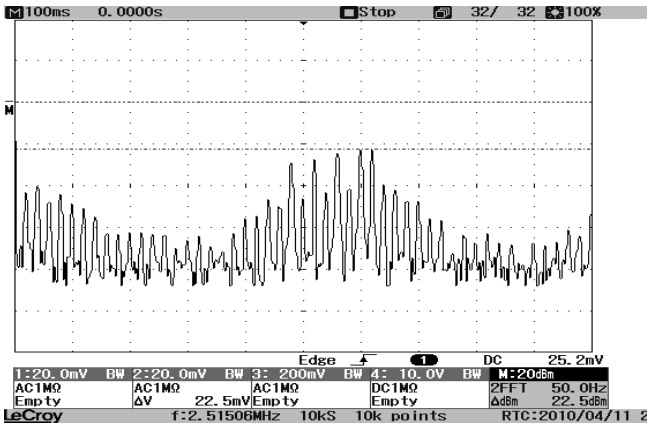


Fig.9. PSD of the second test noise

Fig. 9 and 10 show the noise spectrum resulted for the second and the third noise test. The highest level of noise is measured between cursors an it can be read in the lower right side of the screen captures as ΔdBm.

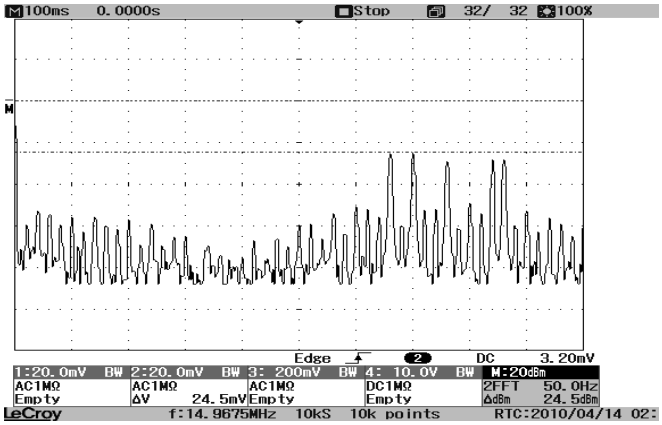


Fig.10. PSD of the third test noise

After the convergence of the main filter using an adaptation step $\mu = 0.0005$, the residual noise spectra for the three test signals are presented in Fig. 11, 12, and 13.

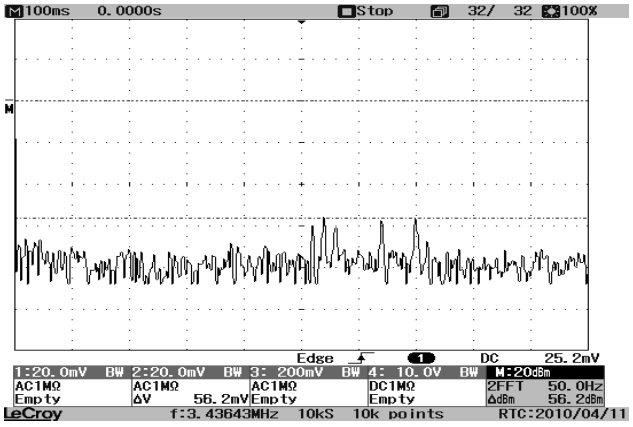


Fig.11. PSD of error signal for the first test noise

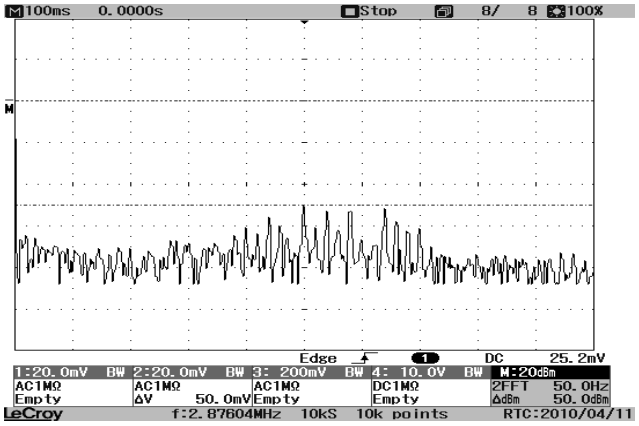


Fig.12. PSD of error signal for the second test noise

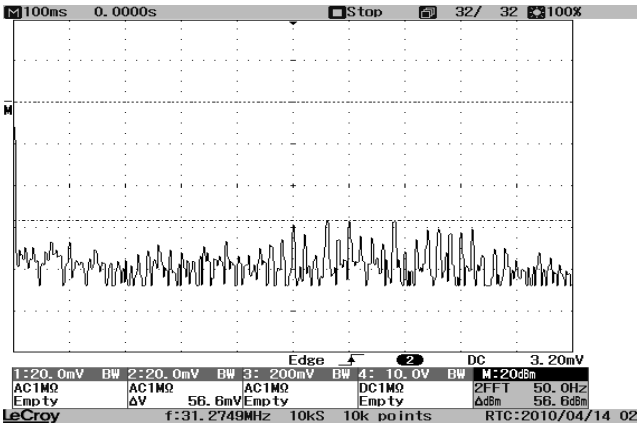


Fig.13. PSD of error signal for the third test noise

IV. CONCLUSION

A real-time implementation of an ANC system on a FPGA circuit was presented, together with design tools and methodology details and experimental results. The FXLMS algorithm was used, designed in LabView FPGA, and implemented on Spartan-3E Starter Kit board. The FPGA has proved to be a good choice for very fast and parallel digital signal processing algorithms. The level of noise attenuation archived by the system was higher than 27 dB, enough to justify the use of the system as a practical application.

REFERENCES

[1] S. M. Kuo and D.R. Morgan, *Active Noise Control Systems: Algorithms and DSP Implementations*, New York: John Wiley & Sons, 1996.

[2] S. M. Kuo and C. Chen, "Implementation of adaptive filters with the TMS320C25 or the TMS320C30", in *Digital Signal Processing Applications with the TMS320 Family*, Vol. 3, P. Papamichalis, Ed. Englewood Cliffs, NJ: Prentice-Hall, 1990, pp. 191-271.

[3] S. Elliot, *Signal Processing for Active Control*, London: Academic Press, 2001.

[4] S. Elliot, *Down With Noise,* IEEE Spectrum,1999, Vol. 36, pp.54-61.

[5] R. Woods, J. McAllister, G. Lightbody, Y. Yi, *FPGA-Based Implementation of Signal Processing Systems*, John Wiley & Sons, 2008.

[6] S. Haykin, *Adaptive Filter Theory, 4th Edition*, Upper Saddle River, NJ: Prentice-Hall, 2002.

[7] S. M. Kuo, I. Panahi, K. M. Chung, T. Horner, M. Nadeski, and J. Chyan, "Design of Active Noise Control Systems with the TMS320 Family", Digital Signal Processing Products – Semiconductor Group, Texas Instruments, 1996.

[8] D. G. Manolakis, V. K. Ingle, and S. Kogon, *Statistical and Adaptive Signal Processing*, McGraw- Hill, 2000.

[9] P. P. Chu, *FPGA Prototyping by Verilog Examples*, John Wiley and Sons, 2008.

[10] D. R. Morgan, "Analysis of Multiple Correlation Cancellation Loops With a Filter in the Auxiliary Path," *IEEE Transactions on Acoustics, Speech and Signal Processing*, Vol. 28, No. 4, August 1980, pp. 454–467.

Inter-VLAN Routing using VLAN Trunk Protocol and Spanning Tree Protocol

D. M. Anghelina, *University of Craiova*

***Abstract*—The present paper describes two methods dedicated to basic inter-VLAN routing, using VLAN Trunk Protocol and Spanning Tree Protocol. The example network has been designed and an addressing scheme has been documented, based on the requirements. The network devices, as well as the PC have been configured in order to ensure connectivity throughout the network. A network management strategy has been designed and simulated, based on Virtual Local Area Networks (VLANs), which allow a logical segmentation of communications networks, in order to enable the management of network devices.**

I. Introduction

DIGITAL communication, using data, voice and video is critical for the small- and medium-sized business. Therefore, a properly designed local area network and the selection of appropriate devices to support the network specifications are fundamental for surviving in the business.

Hierarchical network design involves identifying discrete layers of the network, each layer fulfilling specific functions that define its role within the network. Hierarchical network designs have many benefits, among which scalability, redundancy and high performance [1]-[2].

II. Virtual Local Area Networks and Protocols

A. Hierarchical Network Design

There are three layers that can be identified in a typical hierarchical design model: access, distribution and core [3]-[6]. The main purpose of the access layer is providing a means of connecting devices to the network and controlling which devices may communicate on the network. It includes end devices and routers, switches, bridges, hubs and wireless access points. The distribution layer represents the means of controlling the flow of network traffic using policies and broadcast domains by performing routing functions between virtual local area networks (VLANs) defined at the access layer. The core layer of the hierarchical design represents the high-speed backbone of the network and is critical for the core to be highly available and redundant.

A Layer 2 switch only allows switching and filtering based on the MAC address, which is characteristic to the OSI data link layer (Layer 2) and is completely transparent to network protocols and user applications. On the other hand, a Layer 3 switch can also use IP address information and is capable of performing Layer 3 routing functions, thus reducing the number of dedicated routers in a Local Area Network [4].

Manuscript received May 15, 2010.

D. M. Anghelina is with the University of Craiova, Faculty of Automation, Computers and Electronics, Craiova, DJ, 200585, Romania (phone: 0040745205109; e-mail: anghelina.diana@yahoo.com).

B. Virtual Local Area Networks (VLANs)

A Virtual Local Area Network (VLAN) mainly represents a logically separate IP subnetwork and allows the coexistence of multiple IP networks and subnets on the same switched network [9]. In order to communicate on the same VLAN, computers must have an IP address and a subnet mask that is consistent for that VLAN. The switch needs to be configured with the VLAN and each port in the VLAN must be assigned to the VLAN. The communication between devices on two separate networks and subnets must be supported by a router, regardless of the use of VLANs [5].

The management of Virtual Local Area Networks involves verifying VLANs and port memberships, by using specific *show* commands.

C. VLAN Trunk Protocol (VTP)

VLANs may be extended across an entire network by using VLAN trunks. Trunks are links between the switches and routers of VLANs and do not belong to any specific VLAN, as in [6].

VLAN Trunk Protocol (VTP) reduces administration tasks in a switched network and allows the configuration of a switch so that it will propagate VLAN configurations to other switches in the network. Switches may be configured either in the role of a VTP server or a VTP client. The VTP server has the main goal of distributing and synchronizing VLAN information to VTP-enabled switches throughout the switched network. VTP stores VLAN configurations in the VLAN database called vlan.dat. VLAN configuration consistency across the network and accurate tracking and monitoring of VLANs are among the most important benefits of the VLAN Trunk Protocol, but dynamic reporting and trunk configuration of added VLANs across a network are also of high importance [6].

D. The Spanning Tree Protocol (STP)

The Spanning Tree Protocol (STP) ensures that there is only one logical path between all destinations on the network by intentionally blocking redundant paths that could cause a loop. Blocking the redundant paths is critical to preventing loops in a network. The physical paths still exist to provide redundancy, but these paths are disabled in order to prevent the loops from occurring. If the path is ever needed to

compensate for a network cable or switch failure, STP recalculates the paths and unblocks the necessary ports to allow the redundant path to become active, as in [6].

STP uses the Spanning Tree Algorithm (STA) to determine which switch ports on a network need to be configured for blocking to prevent loops from occurring. The Spanning Tree Algorithm (STA) designates a single switch as the root bridge and uses it as the reference point for all path calculations.

III. Inter-VLAN Routing Using VLAN Trunk Protocol (VTP)

Fig. 1 presents the network proposed for the experiment:

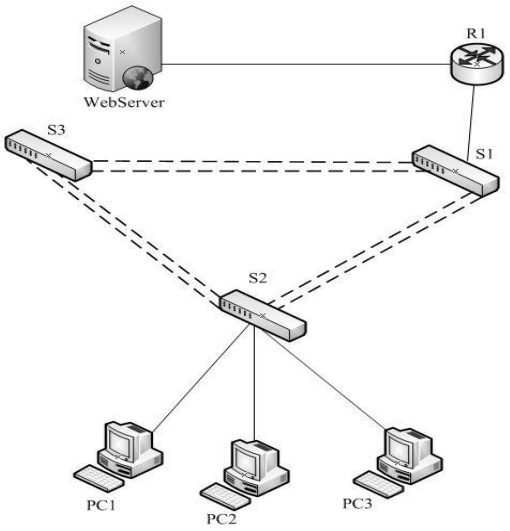


Fig.1.The network topology used for the experiment.

Starting with given network requirements, the aim of the experiment is to perform basic switch configurations, configure addressing on PCs, configure VTP and inter-VLAN routing. Notations: R1 – router; S1, S2. S3 – switches; PC1, PC2, PC3 – computers.

The first step in this experiment is designing an appropriate addressing scheme for the network, based on the requirements shown in the topology [10]. The result is Table I, which presents the IP addresses' assignment for the switches, PCs, Server and router, including the subnet masks and the default gateway addresses.

TABLE I
Address Assignment for the Network Devices

Device	Interface	IP Address	Subnet Mask	Default Gateway
S1	VLAN99	172.17.99.11	255.255.255.0	172.17.99.1
S2	VLAN99	172.17.99.12	255.255.255.0	172.17.99.1
S3	VLAN99	172.17.99.13	255.255.255.0	172.17.99.1
R1	Fa0/0	According to the interfaces' configuration table.		N/A
	Fa0/1	172.17.50.1	255.255.255.0	N/A
PC1	NIC	172.17.10.21	255.255.255.0	172.17.10.1
PC2	NIC	172.17.20.22	255.255.255.0	172.17.20.1
PC3	NIC	172.17.30.23	255.255.255.0	172.17.30.1
Server	NIC	172.17.50.254	255.255.255.0	172.17.50.1

The resulting topology diagram is presented in Fig. 2.

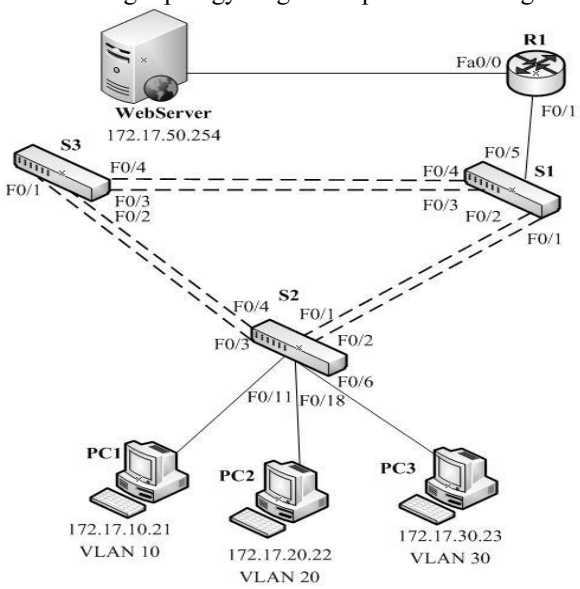


Fig.2.The topology diagram for the experiment.

For switch S2 the port assignments are documented in Table II.

TABLE II
Port Assignment for Switch S2

Port	Assignment	Network
Fa0/1 - 0/5	802.1q Trunk (VLAN 99 Native)	172.17.99.0/24
Fa0/6 - 0/10	VLAN 30	172.17.30.0/24
Fa0/11 – 0/17	VLAN 10	172.17.10.0/24
Fa0/18 – 0/24	VLAN 20	172.17.20.0/24

Table III describes the configuration of the subinterfaces on router R1.

TABLE III
Subinterface Configuration for Router R1

Port	Assignment	IP Address
Fa0/0.1	VLAN 1	172.17.1.1/24
Fa0/0.10	VLAN 10	172.17.10.1/24
Fa0/0.20	VLAN 20	172.17.20.1/24
Fa0/0.30	VLAN 30	172.17.30.1/24
Fa0/0.99	VLAN 99	172.17.99.1/24

The next step in the experiment involves configuring the switches with basic configurations. The configuration of switches S1, S2 and S3 according to the addressing table I requires a series of basic tasks, among which: configuring the switch hostname, disable DNS lookup, configuring the default gateway and configuring passwords for EXEC mode, console connections and vty connections, as in (1).

```
Switch>enable
Switch#config term
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#hostname S1
S1(config)#enable secret cisco
S1(config)#no ip domain-lookup
S1(config)#ip default-gateway 172.17.99.1
S1(config)#line console 0
S1(config-line)#password cisco
S1(config-line)#login
S1(config-line)#line vty 0 15
```
(1)

```
S1(config-line)#password cisco
S1(config-line)#login
S1(config-line)#end
%SYS-5-CONFIG_I: Configured from console by console
S1#copy running-config startup-config
Destination filename [startup-config]? [enter]
Building configuration...
```

The Ethernet Interfaces on the host PCs PC1, PC2 and PC3 must also be configured with the IP addresses from the addressing Table I, as in [7]. The next main goal of the experiment is configuring the VLAN Trunk Protocol (VTP) on switches S1, S2 and S3, by first enabling the user ports on the switches in access mode, as in (2).

```
S2(config)#interface fa0/6
S2(config-if)#switchport mode access
S2(config-if)#no shutdown
S2(config-if)#interface fa0/11                    (2)
S2(config-if)#switchport mode access
S2(config-if)#no shutdown
S2(config-if)#interface fa0/18
S2(config-if)#switchport mode access
S2(config-if)#no shutdown
```

The VTP configuration on the three switches will be accomplished using Table IV and as described in (3).

TABLE IV
VTP CONFIGURATION FOR SWITCHES S1, S2 AND S3

Switch Name	VTP Operating Mode	VTP Domain	VTP Password
S1	Server	Infofer	cisco
S2	Client	Infofer	cisco
S3	Client	Infofer	cisco

```
S1(config)#vtp mode server
Device mode already VTP SERVER.
S1(config)#vtp domain Infofer
Changing VTP domain name from NULL to Infofer
S1(config)#vtp password cisco
Setting device VLAN database password to cisco    (3)
S1(config)#end
S2(config)#vtp mode client
Setting device to VTP CLIENT mode
S2(config)#vtp domain Infofer
Changing VTP domain name from NULL to Infofer
S2(config)#vtp password cisco
Setting device VLAN database password to cisco
S2(config)#end
S3(config)#vtp mode client
Setting device to VTP CLIENT mode
S3(config)#vtp domain Infofer
Changing VTP domain name from NULL to Infofer
S3(config)#vtp password cisco
Setting device VLAN database password to cisco
S3(config)#end
```

The next step in setting up VTP on the switches is configuring the trunking ports and designating the native VLAN for the trunks, as in [6].

The Fast Ethernet interfaces Fa0/1 through Fa0/5 are configured as trunking ports, and VLAN 99 is designated as the native VLAN for these trunks. These ports were disabled and must be re-enabled now using the no shutdown command. Only the commands for the FastEthernet0/1 interface on each switch are shown, but the commands will be applied up to the FastEthernet0/5 interface, as in (4).

```
S1(config)#interface fa0/1
S1(config-if)#switchport mode trunk
S1(config-if)#switchport trunk native vlan 99
S1(config-ino shutdown
f)#S1(config)#end
S2(config)#interface fa0/1
S2(config-if)#switchport mode trunk                (4)
S2(config-if)#switchport trunk native vlan 99
S2(config-if)#no shutdown
S2(config-if)#end
S3(config)#interface fa0/1
S3(config-if#switchport mode trunk
S3(config-if)#switchport trunk native vlan 99
S3(config-if)#no shutdown
S3(config-if-#end
```

The VLANs in Table V will be configured on the VTP server, as in (5).

TABLE V
VTP SERVER VLANS

VLAN	VLAN Name
VLAN 99	management
VLAN 10	craiova
VLAN 20	timisoara
VLAN 30	cluj

```
S1(config)#vlan 99
S1(config-vlan)#name management
S1(config)#vlan 10
S1(config-vname craiova
lan)#S1(config)#vlan 20                            (5)
S1(config-vlan)#name timisoara
S1(config)#vlan 30
S1(config-vlan)#name cluj
S1(config-vlan)#end
```

The creation of the VLANs can be verified by using the *show vlan brief* command, according to Fig. 3. The VLANs creation on S1 must be verified for correct distribution to the client switches, by using the *show vlan brief* command on S2 and S3, as in Fig. 4.

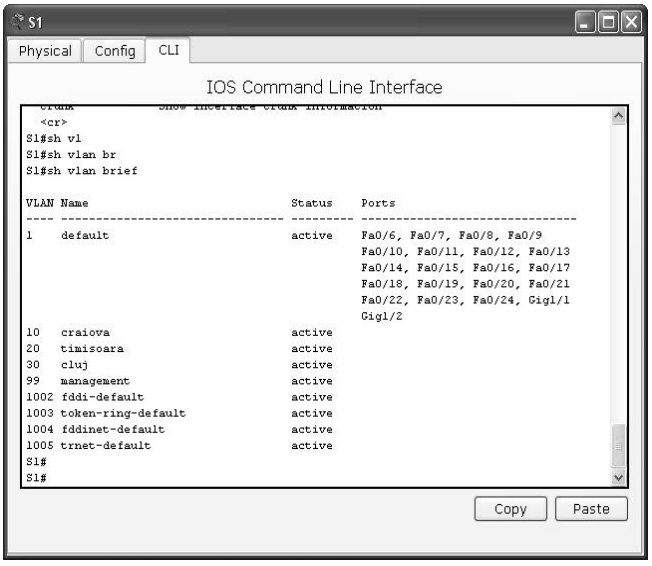


Fig.3.The *show vlan brief* command on switch S1.

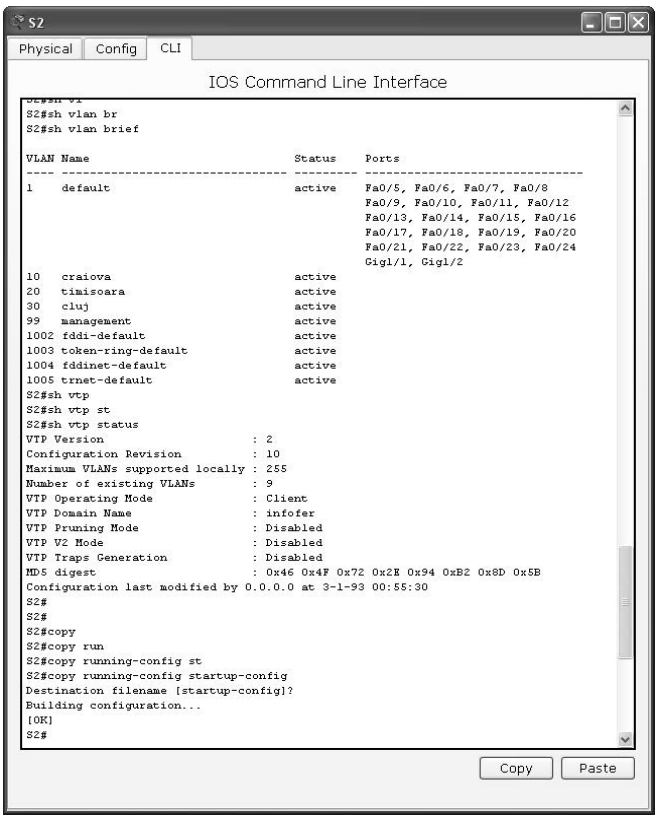


Fig.4.The *show vlan brief* command on switch S2.

The management interface address will be configured on all three switches, as in (6).

```
S1(config)#interface vlan99
S1(config-if)#ip address 172.17.99.11 255.255.255.0
S2(config)#interface vlan99                                    (6)
S2(config-if)#ip address 172.17.99.12 255.255.255.0
S3(config)#interface vlan99
S3(config-if)#ip address 172.17.99.13 255.255.255.0
```

In order to verify that the switches have been correctly configured, the best method is pinging between them. From S1, we shall ping the management interface on S2 and S3. From S2, we shall also ping the management interface on S3, as described in Fig. 5.

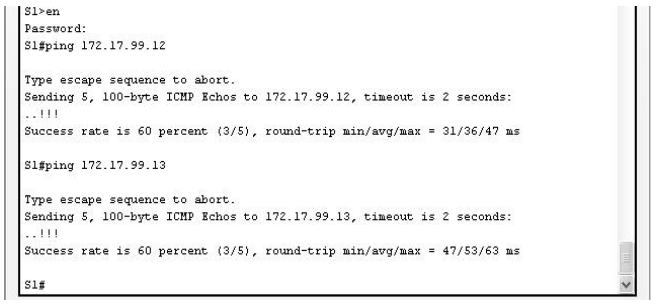


Fig.5. The *ping* command on switch S1.

The assignments of switch ports to VLANs on S2 are listed in Table II, but we shall only assign the first port from each range, as in (7).

```
S2(config)#interface fa0/6
S2(config-if)#switchport access vlan 30
S2(config-if)#interface fa0/11
S2(config-if)#switchport access vlan 10
S2(config-if)#interface fa0/18                                 (7)
S2(config-if)#switchport access vlan 20
S2(config-if)#end
S2#copy running-config startup-config
Destination filename [startup-config]? [enter]
Building configuration...
[OK]
```

Connectivity between VLANs can be verified by pinging between the three PCs from the command prompt.

At first, the *ping* commands are not successful, because inter-VLAN routing has not been yet configured.

In order to configure the Router and the Remote Server LAN, a basic configuration on the router must be set, as in [8]-[9]. The router will be configured with hostname R1, DNS lookup will be disabled and passwords will be configured for the EXEC mode, console connections and vty connections. Next, the trunking interface on R1 must be configured.

Connectivity between VLANs requires routing at the network layer, which can be configured by means of two different methods, as in [6]. The first method is that a router or a Layer 3 capable switch may be connected to a LAN switch with multiple connections, using a separate connection for each VLAN that requires inter-VLAN connectivity. The limitations of this method are excessive wiring and manual configuration, the lack of sufficient Fast Ethernet ports on routers and the fact that few of the ports on the Layer 3 switches and routers will be used.

An alternative method of configuring inter-VLAN connectivity would be creating one or more Fast Ethernet connections between the router and the distribution layer switch and configuring these connections as dot1q trunks, thus allowing all inter-VLAN traffic to be carried to and from the routing device on a single trunk. This approach requires that the Layer 3 interface be configured with multiple IP addresses, by creating virtual interfaces, called subinterfaces, on one of the router Fast Ethernet ports and configuring them to be use dot1q.

The subinterface configuration approach requires entering subinterface configuration mode, establishing trunking encapsulation, associating a VLAN with the subinterface and assigning an IP address from the VLAN to the subinterface, as in (8).

```
R1(config)#interface fastethernet 0/0
R1(config-if)#no shutdown
R1(config-if)#interface fastethernet 0/0.1
R1(config-subif)#encapsulation dot1q 1
R1(config-subif)#ip address 172.17.1.1 255.255.255.0
R1(config-subif)#interface fastethernet 0/0.10
R1(config-subif)#encapsulation dot1q 10
R1(config-subif)#ip address 172.17.10.1 255.255.255.0
R1(config-subif)#interface fastethernet 0/0.20                 (8)
R1(config-subif)#encapsulation dot1q 20
R1(config-subif)#ip address 172.17.20.1 255.255.255.0
R1(config-subif)#interface fastethernet 0/0.30
R1(config-subif)#encapsulation dot1q 30
R1(config-subif)#ip address 172.17.30.1 255.255.255.0
```

R1(config-subif)#interface fastethernet 0/0.99
R1(config-subif)#encapsulation dot1q 99 native
R1(config-subif)#ip address 172.17.99.1 255.255.255.0

These configuration commands enable the physical interface by using the no shutdown command, because router interfaces are down by default. The subinterface will then be up by default. The subinterface numbers are identical to the numbers of the VLANs. The native VLAN must be specified on the L3 device so that it is consistent with the switches, because otherwise, VLAN 1 is native by default, and there is no communication between the router and the management VLAN on the switches.

The server LAN interface on R1 will also be configured, according to (9).

```
R1(config)#interface FastEthernet0/1
R1(config-if)#ip address 172.17.50.1 255.255.255.0
R1(config-if)#description server interface                    (9)
R1(config-if)#no shutdown
R1(config-if)#end
```

Six networks will be configured, as in (10).

```
R1#show ip route
<output omitted>
Gateway of last resort is not set
172.17.0.0/24 is subnetted, 6 subnets
C 172.17.1.0 is directly connected, FastEthernet0/0.1
C 172.17.10.0 is directly connected, FastEthernet0/0.10        (10)
C 172.17.20.0 is directly connected, FastEthernet0/0.20
C 172.17.30.0 is directly connected, FastEthernet0/0.30
C 172.17.50.0 is directly connected, FastEthernet0/1
C 172.17.99.0 is directly connected, FastEthernet0/0.99
```

Inter-VLAN routing can be verified by pinging the remote server (172.17.50.254) and the other two hosts (172.17.20.22 and 172.17.30.23) from PC1, as in [9].

IV. Inter-VLAN Routing Using Spanning Tree Protocol (STP)

Considering the network in Fig. 1, the Spanning Tree Protocol (STP) will be configured in the present topology [6]. Switch S3 is chosen as the switch with the lowest bridge ID (BID) and automatically becomes the root bridge for the Spanning Tree Algorithm (STA) calculations, as in Fig. 6.

The STA calculates the shortest path to the root bridge. Each switch uses the STA to determine which ports to block. When the STA has determined which paths are to be left available, it configures the switch ports into distinct port roles. The root ports are the switch ports closest to the root bridge. In this example network, the root port on switch S1 is F0/1 configured for the trunk link between switch S3 and switch S1. The root port on switch S2 is F0/1, configured for the trunk link between switch S3 and switch S2.

The designated ports are all the non-root ports that are still permitted to forward traffic on the network [6]-[9]. In the experiment, switch ports F0/1, F0/2, F0/3 and F0/4 on switch S3 are designated ports. Switch S1 also has its ports F0/3 and F0/4 configured as designated ports.

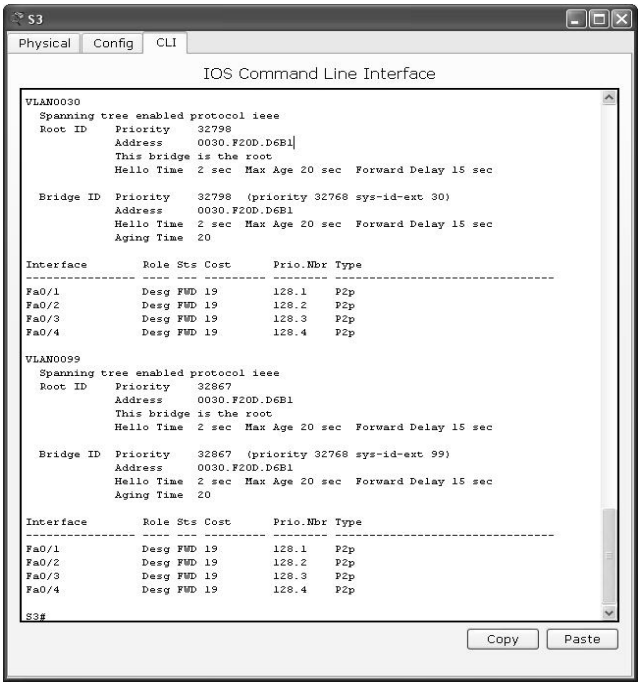


Fig.6. Switch S3 becomes the root bridge.

The non-designated ports are all the ports configured to be in a blocking state to prevent loops. In the present network topology, the STA configured ports F0/3, F0/4 on switch S2 in the non-designated role. Ports F0/2, F0/3 and F0/4 on switch S2 are in the blocking state, as in Fig. 7.

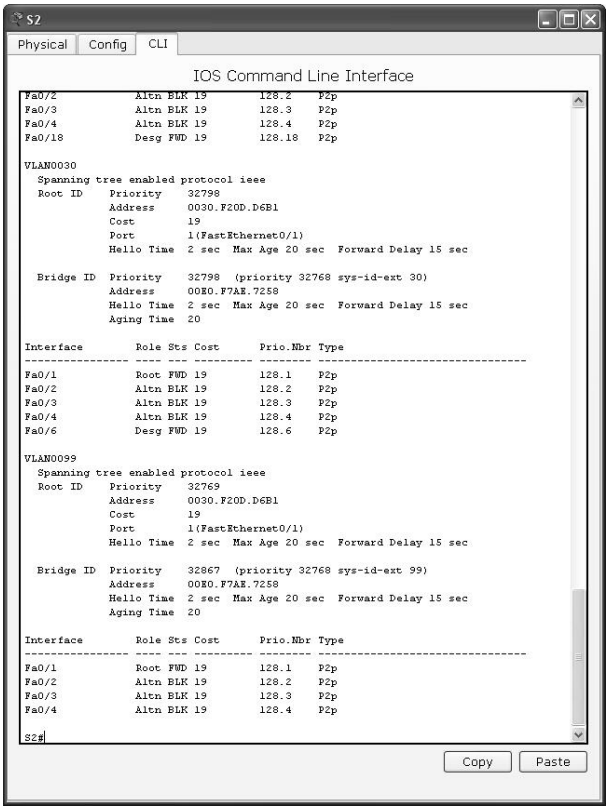


Fig.7. Ports state on Switch S2 before blocking ports on Switch S1.

Considering the scenario when ports F0/1 and F0/2 on S1 are forced into shut down mode and will not participate anymore in the STP protocol, ports F0/3 and F0/4 on S1 change state to active mode and the traffic will be redirected through these ports, as described in Fig. 8.

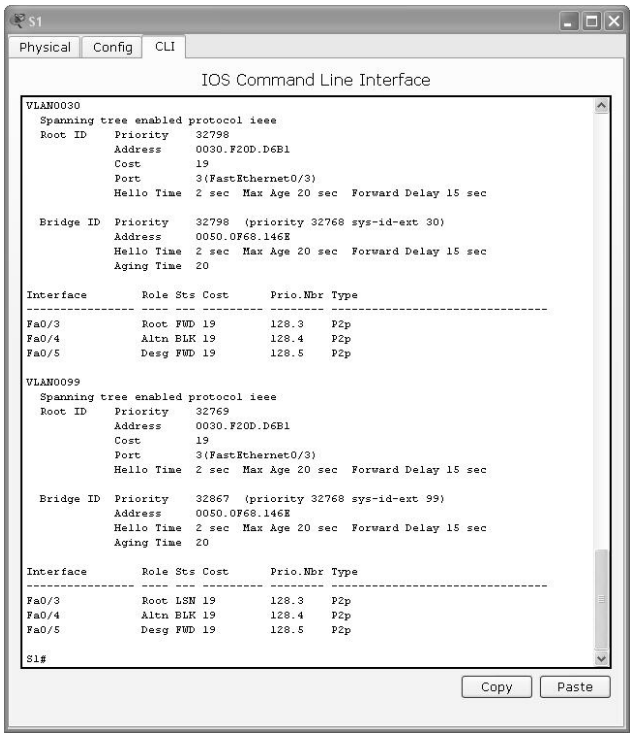


Fig.8. Ports F0/1 and F0/2 on Switch S1 are in shutdown mode.

Consequently, ports F0/3 and F0/4 change state to designated forwarding state. After blocking the ports on switch S1, the ports on switch S2 will be in a state presented by Fig. 9.

Therefore, in the scenario when ports F0/1 and F0/2 on S1 are forced into shut down mode, the traffic will be redirected through ports F0/3 and F0/4, which will change state to active mode.

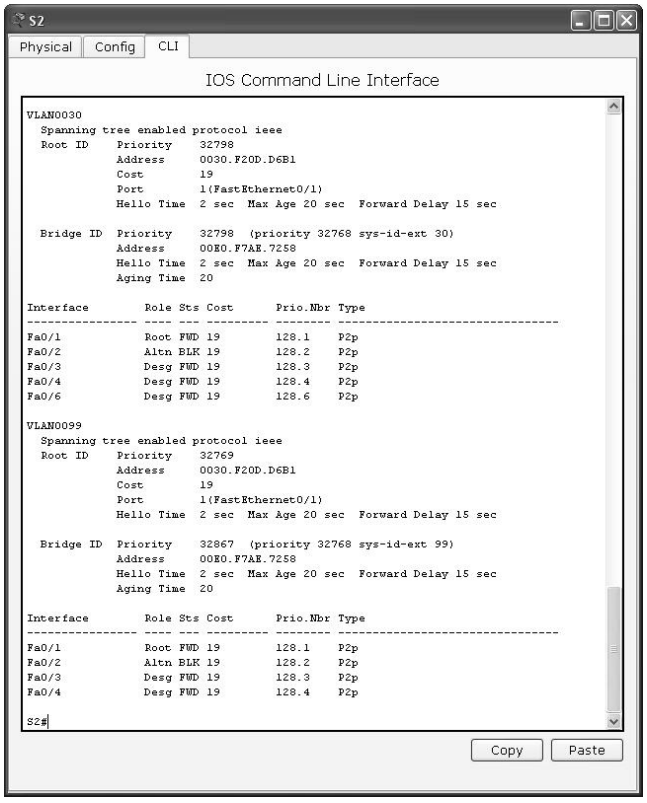


Fig.9. Ports state on Switch S2 after blocking ports on Switch S1.

V. CONCLUSION

Nowadays network performance is a critical factor in the productivity and reputation of any organization. The present paper describes a network management strategy based on Virtual Local Area Networks (VLANs). The example network topology supports two methods dedicated to basic inter-VLAN routing, which use VLAN Trunk Protocol and Spanning Tree Protocol.

REFERENCES

[1] A. Clemm, "Network Management Fundamentals", *Cisco Press*, pp.417-433, 2006.

[2] G. Shields, "Network Management for the Mid-Market", *Realtimepublishers*, pp.29-58, 2009.

[3] J. Doyle, J. DeHaven, "Routing TCP/IP", *Cisco Press*, vol. A247, pp.239-270, 2008.

[4] J. D. McCabe, "Network Analysis, Architecture, and Design", 3rd ed., *Morgan Kaufmann Publishers*, pp.249-299, 2007.

[5] CCIE Fundamentals: "Network Design and Case Studies", *Cisco Press Publications*, pp.536-557, 632-684, 1999.

[6] CCNA Exploration: "LAN Switching and Wireless", *Cisco Networking Academy*, 2007.

[7] G. Held, "Ethernet Networks: Design, Implementation, Operation, Management", 4th ed., *John Wiley & Sons*, pp.365-407, 2003.

[8] A. Tannenbaum, "Computer Networks", 4th ed., *Prentice Hall PTR*, pp.490-567, 2002.

[9] A. Limoncelli, C. Hogan, S. Chalup, "Practice of System and Network Administration", 2nd ed., *Addison-Wesley Professional*, pp.213-270, 2007.

[10] "Understanding IP Addressing: Everything You Ever Wanted To Know", *3Com Corporation*, 2001.

Nonsmooth μ synthesis

Pierre Apkarian

***Abstract*— We revisit robust complex- and mixed-μ synthesis problems based on upper bounds and show that they can be recast as specially structured controller design programs. The proposed reformulations suggest a streamlined handling of μ synthesis problems using recently developed (local) nonsmooth optimization methods where both scalings or multipliers and a controller of given structure are obtained simultaneously. A preliminary version of the code will be released through the MATLAB *Robust Control Toolbox* developed by The MathWorks, Inc.**

INTRODUCTION

Most if not all control engineering problems motivate consideration of plants subject to uncertainties. Uncertainties describe plant variations in the frequency domain and this is often referred to as complex or Linear Time-Invariant uncertainties. They can also capture deviations of some physical quantities from their nominal values and in that case, parametric uncertainties is the appropriate terminology. In a practical situation the nominal plant undergoes variations with respect to both types of uncertainty, this is the so-called mixed uncertainty case. Synthesizing feedback controllers achieving robustness in face of mixed uncertainties remains a very challenging problem at the core of modern robust control theories. When the controller is fixed, assessing the robustness properties of the closed-loop system though apparently much simpler than the synthesis problem is NP-hard in general [1]. A number of potent theories with associated practical tools have been developed over the past 30 to answer the analysis question with mild conservatism. We are pointing here to μ-analysis [2] or to Lyapunov-based analysis techniques where LMIs (Linear Matrix Inequalities) form the central computational component [3]. No such methods and tools with comparable merit and efficiency are available so far for the synthesis problem. The inherent non-convexity of robust synthesis problems is certainly the culprit cause. Among numerous attempts to estimate solutions of robust control problems, μ-synthesis based on upper bounds was probably the most explored avenue. In this approach, the true robustness indicator is replaced by mildly conservative upper bounds. The synthesis problems is then decomposed into analysis and synthesis both of which are covered by efficient numerical tools. At this stage it seems important to pinpoint potential sources of weakness in those techniques

- the biconvex nature of the synthesis problem suggests alternating between analysis and synthesis phases till nor further progress is observed. The term $D-K$ or alike is often used in this context. This immediately raises questions about the convergence of such schemes.
- in conventional μ-synthesis, scalings are computed pointwise in frequency and must be fitted by state-space realizable systems [4]. This again questions the numerical stability and accuracy of such procedures.
- in the same approach either scalings are restricted to be stable or stable factorizations are searched for with detrimental impact on conservatism and size inflation in the state dimension of the problem.

In this work we suggest a different strategy bypassing most of the aforementioned difficulties. The order and structure of the controller is set beforehand as well as the order of scalings and multipliers. Scalings and multipliers are sought in non-necessarily stable state-space realizable form. We do not rely on curve fitting nor on stable factorizations and nor on ad-hoc basis selection. We establish that both complex-μ and mixed-μ syntheses based on upper bounds can be reformulated as synthesis programs with an augmented controller of special structure. The augmented controller encapsulates the original structured controller along with the scalings or multipliers. It follows from the latter cast that the synthesis problem can be solved (locally) with recently available nonsmooth optimization techniques [5]. Synthesis is accomplished in one shot which means the controller together with the scalings or multipliers are obtained simultaneously. The proposed technique is endowed with a sound *local* convergence certificate and thus has potential to overcome premature terminations arising in biconvex $D-K$ iteration schemes. Using our nonsmooth techniques to compute local solutions is certainly less ambitious than global approaches [6], [7] but is more efficient numerically and supported by a wide range of experiments. See [8] and references therein.

NOTATION

We use a few concepts from nonsmooth analysis covered by [9]. For a locally Lipschitz function $f:\mathbb{R}^n\to\mathbb{R}$, $\partial f(x)$ denotes its Clarke subdifferential at x. We shall LFT and star product notations $\mathcal{F}_l$, $\mathcal{F}_u$, $\star$, respectively, as defined in [10].

ONERA-CERT, Centre d'études et de recherches de Toulouse, Control System Department, 2 av. Edouard Belin, 31055 Toulouse, France - and - Institut de Mathématiques, Université Paul Sabatier, Toulouse, France - Email: apkarian@cert.fr - Tel: +33 5.62.25.27.84 - Fax: +33 5.62.25.27.64

We introduce the bilinear transformation operator $\mathcal{B}_q := [I_q \quad -\sqrt{2}I_q; \sqrt{2}I_q \quad -I_q]$ which is the unit element for the Redheffer star product. The notation H_ω is used to designate arbitrary frequency-dependent matrix-valued functions while the notation $H(j\omega)$ implicitly assumes an underlying state-space realization.

To save a space terms denoted $(.)^H$ will be induced by Hermitian symmetry. For a square matrix $A \in \mathbb{R}^{n\times n}$, $\alpha(A)$ stands for is spectral abscissa.

I. Complex- and mixed-μ syntheses

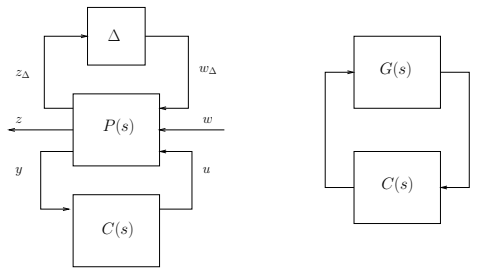


Fig. 1. μ synthesis interconnection & nominal stability loop

The general robust synthesis interconnection discussed throughout the paper is shown in figure 1, where $P(s)$ is a linear time invariant plant, $C(s)$ a linear time invariant controller to be designed and Δ refers to uncertainties. The signal pair $(w_\Delta, z_\Delta) \in \mathbb{R}^N \times \mathbb{R}^N$ is the uncertainty channel, $(w, z) \in \mathbb{R}^{m_1} \times \mathbb{R}^{m_1}$ is the performance channel, $(u, y) \in \mathbb{R}^{m_2} \times \mathbb{R}^{p_2}$ is the control-measurement channel. We also assume throughout the plant P is state-space realizable in the form

$$P(s) := \left[\begin{array}{c|ccc} A & B_\delta & B_1 & B_2 \\ \hline C_\delta & D_{\delta\delta} & D_{\delta 1} & D_{\delta 2} \\ C_1 & D_{1\delta} & D_{11} & D_{12} \\ C_2 & D_{2\delta} & D_{21} & D_{22} \end{array}\right] . \tag{1}$$

The plant $P(s)$ describes how uncertainties Δ impact the system dynamics, how the controller component $C(s)$ acts on these dynamics, and determines a performance channel (w, z) reflecting practical specifications. It is constructed from an underlying system $G(s)$ with state-space realization

$$G(s) := C_G(sI - A_G)^{-1}B_G + D_G\,,\ D_G \in \mathbb{R}^{p_2\times m_2}\,.$$

Generally speaking, $P(s)$ is built from $G(s)$ by adding weighting filters which emphasize appropriately chosen frequency ranges [11], [12]. For future use, we shall denote $A(G(s), C(s))$ the state-space A-matrix resulting from the positive feedback loop of $G(s)$ and $C(s)$. We infer that nominal stability of the interconnection in figure 1 can be expressed as $\alpha\{A(G(s), C(s))\} \le -\epsilon$ for some small enough $\epsilon > 0$.

As usual [2], [13], [14], we consider structural restrictions on the uncertainty block Δ that is $\Delta \in \mathbf{\Delta}$ with the definition $\mathbf{\Delta} :=$

$$\{\Delta = \mathrm{diag}(\delta_1 I_{r_1}, \ldots \delta_q I_{r_q}, \Delta_P) :\ \delta_i \in \mathbb{F},\ \Delta_P \in \mathbb{C}^{m_1\times m_1}\}\,,$$

with $\mathbb{F} = \mathbb{C}$ or $\mathbb{R}$ depending on the context and $N = \sum_{i=1}^q r_i + m_1$. We will refer to $\delta_i I_{r_i}$ with $\delta_i \in \mathbb{C}$ as repeated scalar LTI uncertainties, while $\delta_i \in \mathbb{R}$ designates repeated parametric uncertainties, and Δ_P will stand for the performance block. To ease the notation, we also introduce for $\mathbb{F} = \mathbb{C}$ or $\mathbb{R}$ the reduced sets

$$\mathbf{\Delta}_\mathbb{F} := \{\Delta = \mathrm{diag}(\delta_1 I_{r_1}, \ldots \delta_q I_{r_q}) :\ \delta_i \in \mathbb{F},\ i = 1, \ldots, q\}\,,$$

and $\mathbf{\Delta_P} := \{\Delta = \Delta_P \in \mathbb{C}^{m_1\times m_1}\}$. For $\mathbf{\Delta}$, the notation $\Delta \in \mathbf{B\Delta}$ means $\Delta \in \mathbf{\Delta}$ and $\overline{\sigma}(\Delta) \le 1$ and similarly for $\mathbf{\Delta}_\mathbb{F}$ and $\mathbf{\Delta_P}$.

A key ingredient in assessing robust performance of the interconnection in figure 1 is the Main Loop Theorem [13, p. 289] which states that assuming nominal internal stability, the interconnection is well-posed, internally stable for all $\Delta \in \mathbf{B\Delta}_\mathbb{F}$ and meets robust performance constraints

$$\|\mathcal{F}_u(\mathcal{F}_l(P(s), C(s)),\, \Delta)\|_\infty < 1, \qquad \forall \Delta \in \mathbf{B\Delta}_\mathbb{F}\,, \tag{2}$$

if and only if

$$\mu_\mathbf{\Delta}(\mathcal{F}_l(P(s), C(s))) < 1,\ \forall s = j\omega, \omega \in \mathbb{R} \cup \{\infty\}\,, \tag{3}$$

where here $\mathbf{\Delta}$ is the diagonally augmented structure $\mathbf{\Delta} = \mathrm{diag}(\mathbf{\Delta}_\mathbb{F}, \mathbf{\Delta_P})$.

The quantity $\mu_\mathbf{\Delta}$ is the structured singular value computed with respect to the uncertainty structure $\mathbf{\Delta}$. This result means that we can assess robust performance of a closed-loop system by using a frequency evaluation of $\mu_\mathbf{\Delta}$. Also, the peak value on the $\mu_\mathbf{\Delta}$ plot captures the inverse of the size of the uncertainty against which the loop maintains robust performance.

II. μ synthesis with dynamic D-scalings

In this section, the focus is on the case $\delta_i \in \mathbb{C}$ for $i = 1, \ldots, q$. Except in special instances [15], [16], μ is essentially intractable and is replaced by an easily computable upper bound. That is, the robust performance condition (3) holds whenever there exist for each ω, D-scaling matrices D_ω such that

$$\overline{\sigma}(D_\omega \mathcal{F}_l(P(j\omega), C(j\omega))D_\omega^{-1}) < 1,\ \forall \omega \in \mathbb{R} \cup \{\infty\}\,, \tag{4}$$

where for all ω, $D_\omega \in \mathbf{D_\Delta} :=$

$$\{D :\ D \in \mathbb{C}^{N\times N}, \det(D) \ne 0,\ \Delta D = D\Delta, \forall \Delta \in \mathbf{\Delta}\}\,.$$

The infinite sequence $(D_\omega)_\omega$ neither needs to be state-space realizable nor to correspond to a stable system. Tractability becomes apparent if one reformulates (4), $\forall \omega \in \mathbb{R} \cup \{\infty\}$

$$\mathcal{F}_l(P(j\omega), C(j\omega))^H S_\omega \mathcal{F}_l(P(j\omega), C(j\omega)) - S_\omega \prec 0\,, \tag{5}$$

where $S_\omega := D_\omega^H D_\omega \succ 0$. What we have obtained is an infinite set of LMI constraints indexed by ω. In the classical μ synthesis approach the left-hand side of (4) is minimized by alternating minimizations over $C(s)$ and over $D(s)$ with either one of these terms fixed. This is the so-called $D-K$

iteration. $C(s)$ is obtained as the result of an H_∞ synthesis run for the scaled plant in (4) whereas state-space realizable D-scalings are built by

- computing estimates D_ω of optimal scalings on a discrete grid of frequencies.
- fitting the finite family $(D_\omega)_\omega$ with stable and inversely stable transfer functions $D(s)$.

The rationale in enforcing stable and minimum phase D-scalings originates in the inherent limitations of H_∞ synthesis which by definition must achieve internal stability of the closed-loop scaled plant $D(s)\mathcal{F}_l(P(s),C(s))D(s)^{-1}$. Apart notable exceptions [17]–[19], the overall procedure is restricted to a few complex full blocks $\Delta_i \in \mathbb{C}^{f_i\times f_i}$ since D-scalings simplify to $D_i(s)=d_i(s)I_{r_i}$. The outlined scheme would become considerably more complex and costly if a couple of scalar blocks $\delta_i I_{r_i}$ were present as D-scalings are now full unstructured LTI systems $D(s)$. Another hindrance concerns the size inflation in the controller order since H_∞ synthesis is a full-order method both the plant order in (1) and the order of the D-scaling accumulate at the outcome in the order of $C(s)$. Despite apparent difficulties and sources of numerical trouble, the whole procedure has proven to work well in a number of examples [4], [20]–[23].

Hereafter, we overcome most of the above-mentioned difficulties by rearranging D-scaling μ synthesis as a nonsmooth program involving a specially structured controller thereby allowing simultaneous construction of both the controller and the scaling in one shot. The resulting program is supported by a sound local optimality certificate established in [5], [24].

For complex uncertainties $\Delta \in \mathrm{diag}(\mathbf{\Delta}_\mathbb{C},\, \mathbf{\Delta_P})$ the suggested rearrangement trick is simple and illustrated in figure 2.

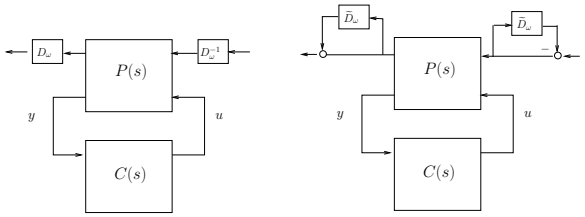


Fig. 2. translation on D-scalings

We introduce new scalings $\widetilde{D}_\omega = D_\omega - I_N$ and infer $\widetilde{D}_\omega\Delta = \Delta\widetilde{D}_\omega$ whenever $D_\omega \in D_{\mathbf{\Delta}}$. With the new translated scalings, the left-hand side diagram of figure 2 clearly is an LFT in $C(s)$ and $\widetilde{D}_\omega$ which is readily derived by computing the transfer function around the latter components. We have

$$D_\omega\mathcal{F}_l(P(j\omega),C(j\omega))D_\omega^{-1} = \mathcal{F}_l\left(P_c(s),\begin{bmatrix} C(s) & 0 & 0\\ 0 & \widetilde{D}_\omega & 0\\ 0 & 0 & \widetilde{D}_\omega\end{bmatrix}\right) \tag{6}$$

with the definition $P_c(s) := L_cP(s)R_c + Q_c :=$

$$\left[\begin{array}{cc} I_N & 0\\ \hline 0 & I_{p_2}\\ 0 & 0\\ I_N & 0\end{array}\right] P(s) \left[\begin{array}{c|ccc} I_N & 0 & -I_N & 0\\ 0 & I_{m_2} & 0 & 0\end{array}\right] + \left[\begin{array}{c|ccc} 0 & 0 & 0 & I\\ \hline 0 & 0 & 0 & 0\\ I & 0 & -I & 0\\ 0 & 0 & 0 & 0\end{array}\right].$$

We refer to L_c, R_c and Q_c as left, right outer factors and constant. Note the partitioning above emphasizes the structure of the LFT in (6). Accepting for a moment the seemingly mild conservatism induced by searching over state-space realizable D-scalings, the complex μ synthesis problem becomes a design problem with specially structured controllers. Incorporating nominal stability, a nonsmooth cast is obtained as

$$\begin{array}{ll}\underset{K(s)}{\text{minimize}} & \sup_{\omega\in\mathbb{R}^+}\ \overline{\sigma}\left(\mathcal{F}_l\left(P_c(j\omega),K(j\omega)\right)\right)\\ \text{subject to} & \text{nominal stability}\\ & K(s) := \left[\begin{array}{ccc} C(s) & 0 & 0\\ 0 & \widetilde{D}(s) & 0\\ 0 & 0 & \widetilde{D}(s)\end{array}\right],\ \widetilde{D}(s)\in\widetilde{\mathbf{D}}_{\mathbf{\Delta}}\,,\end{array}$$

with the shifted scaling set $\widetilde{\mathbf{D}}_{\mathbf{\Delta}}$ defined as

$$\{\widetilde{D}:\ \widetilde{D}\in\mathbb{C}^{N\times N},\ \det(I_N+\widetilde{D})\neq 0,\ \Delta\widetilde{D}=\widetilde{D}\Delta,\ \forall\Delta\in\mathbf{\Delta}\}\,.$$

Nominal stability can be handled in different ways [25]. Adding a constraint on the spectral abscissa of the closed-loop $(G(s),C(s))$ is one possibility: $\alpha\{A(G(s),\,C(s))\}\leq -\epsilon$.

Summing up the above, complex μ synthesis is recast as the nonsmooth program with block-diagonal controller $K(s) := \mathrm{diag}(C(s),\,\widetilde{D}(s),\,\widetilde{D}(s))$:

$$\left|\begin{array}{ll}\underset{K(s)}{\text{minimize}} & \sup_{s=j\omega,\,\omega\in\mathbb{R}^+}\ \overline{\sigma}\left\{\mathcal{F}_l\left(P_c(s),K(s)\right)\right\}\\ \text{subject to} & \alpha\{A(G(s),\,C(s))\}\leq -\epsilon\\ & K(s) := \left[\begin{array}{ccc} C(s) & 0 & 0\\ 0 & \widetilde{D}(s) & 0\\ 0 & 0 & \widetilde{D}(s)\end{array}\right],\ \widetilde{D}(s)\in\widetilde{\mathbf{D}}_{\mathbf{\Delta}}\,.\end{array}\right. \tag{7}$$

REMARKS: A few comments are in order. The supremum in (7) is easily computed using the standard Hamiltonian technique for H_∞ norm computation [26], [27]. It also delivers all the necessary information to construct subdifferential information for program (7).

It is important at this stage to stress that D-scalings $D(s)$ are not necessarily stable hence reducing the conservatism of classical μ-synthesis schemes. The above analysis is easily generalized to multiple complex full uncertainty blocks upon re-defining D-scalings accordingly.

We note that setting $\widetilde{D}(s)=0$ reduces program (7) to a pure H_∞ synthesis problem while keeping $C(s)$ fixed leads to a μ-analysis problem.

Last but not least, the suggested scheme can be used to compute finely structured controllers $C(s)$ including PIDs, decentralized or lead-lag compensators, etc. This attractive feature of the nonsmooth approach is not accessible with conventional $D-K$ iteration procedures since synthesis phases rely on full-order controllers.

III. MIXED-μ SYNTHESIS WITH MULTIPLIERS

This section is concerned with developing an analogue of section II when parametric uncertainties are present. To this aim, we assume $\mathbb{F} = \mathbb{R}$, so that $\delta_i \in \mathbb{R}$ for $i = 1, \ldots, q$. The robust performance problem stated in (3) becomes a mixed-μ synthesis problem. Part of the material in this section is borrowed from [13], [28], [29].

The extension of (5) to parametric uncertainties is the infinite set indexed by ω of LMI constraints

$$F(j\omega)^H S_\omega F(j\omega) + j(G_\omega F(j\omega) - F(j\omega)^H G_\omega) - S_\omega \prec 0, \tag{8}$$

with the shorthand $F(j\omega) := \mathcal{F}_l(P(j\omega), C(j\omega))$ and where G_ω scalings are defined as $G_\omega \in \mathbf{G_\Delta} :=$

$$\left\{G : G = G^H \in \mathbb{C}^{N\times N},\, G\Delta = \Delta^H G,\, \forall \Delta \in \mathbf{\Delta}\right\} .$$

Characterization (8) again seems to advocate for a resolution technique where S_ω, G_ω scalings are computed pointwise, curve fitting is used to built rational representations, and stable factors of scalings are derived, followed by an H_∞ synthesis, and so forth, till no further progress is observed. Unfortunately as the author himself admits, this scheme is not practical and prone to failure when there are repeated scalar blocks [30].

Invoking a bilinear transformation applied to $\mathcal{F}_l(P(j\omega), C(j\omega))$, and denoting $M_\omega = -(S_\omega + jG_\omega)$, inequality (8) can be turned into the negative real condition

$$M_\omega \mathcal{B}_N \star \mathcal{F}_l(P(j\omega), C(j\omega)) + (.)^H \prec 0,\ \forall \omega \in \mathbb{R} \cup \{\infty\},$$

which by associativity of the star product is easily rewritten as

$$M_\omega \mathcal{F}_l(\mathcal{B}_N \star P(j\omega), C(j\omega)) + (.)^H \prec 0,\ \forall \omega \in \mathbb{R} \cup \{\infty\}. \tag{9}$$

Note $\mathcal{B}_N \star P$ is well-posed as soon as the LFT model defined in (1) is well-defined on $\mathbf{B\Delta}$ what we assume throughout. By construction M_ω enjoys the same spatial structure as D-scalings with the positive real property $M_\omega + M_\omega^H \prec 0$, for all ω. For future use, we denote $\mathbf{M_\Delta}$ the set of all such multipliers.

All put together, and proviso nominal stability of the closed-loop underlying plant, the robust performance condition (2) holds whenever $\forall \omega \in \mathbb{R} \cup \{\infty\}$

$$\begin{bmatrix} M_\omega \mathcal{F}_l(\mathcal{B}_N \star P(j\omega), C(j\omega)) & 0 \\ 0 & M_\omega \end{bmatrix} + (.)^H \prec 0\,. \tag{10}$$

Introducing appropriate left and right outer factors L_r, R_r, and constant Q_r and denoting $P_r(s) := (L_r(\mathcal{B}_N \star P(s))R_r + Q_r)$, inequality in (10) is re-expressed as

$$\mathcal{F}_l\left(P_r(j\omega), \begin{bmatrix} C(j\omega) & 0 & 0 \\ 0 & M_\omega & 0 \\ 0 & 0 & M_\omega \end{bmatrix}\right) + (.)^H \prec 0\,. \tag{11}$$

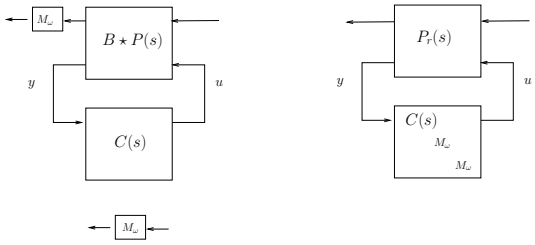


Fig. 3. mixed-μ synthesis as structured controller design

Restricting the search to state-space realizable multipliers $M_\omega := M(j\omega)$, leaves us with the nonsmooth cast:

$$\left|\begin{array}{ll} \underset{K(s)}{\text{minimize}} & \sup_{s=j\omega,\, \omega\in\mathbb{R}^+} \overline{\lambda}\left\{\mathcal{F}_l\left(P_r(s), K(s)\right) + (.)^H\right\} \\ \text{subject to} & \alpha\{A(G(s),\, C(s))\} \le -\epsilon \\ & K(s) := \begin{bmatrix} C(s) & 0 & 0 \\ 0 & M(s) & 0 \\ 0 & 0 & M(s) \end{bmatrix},\ M(s) \in \mathbf{M_\Delta}\,. \end{array}\right. \tag{12}$$

This establishes that mixed-μ upper bound synthesis may be viewed as synthesizing a controller with special (repeated) block-diagonal structure and that the conventional distinction between analysis and synthesis is merely an artefact due to the lack of resolution methods. The case of time-varying parameters with arbitrary rates of variation is easily recovered by restricting the search for multipliers $M(s)$ to real-valued static multipliers with complying structure, $M(j\omega) = M$ for all ω with $M + M^T \prec 0$. Although not developed further here this result has also obvious consequences for the synthesis of robust controllers in the presence of slowly varying parameters and for a number of variants involving multiplier-based characterizations. In the proposed nonsmooth optimization approach the orders of the multipliers as well as the order and structure of the controller is set beforehand. A very favorable feature is therefore that complexity in scalings and multipliers does not meddle in the controller complexity. Multipliers are sought in non-necessarily stable state-space realizable form and do not rely on expansion on finite basis as is often the case with current techniques.

If conservatism is an issue in a specific application the outlined approach is easily generalized to synthesis with multiple frequency bands. By this we mean that a single structured controller $C(s)$ but different multipliers are employed, say, in the low, medium and high frequency ranges.

IV. NONSMOOTH TECHNIQUES

μ-synthesis problems discussed in this paper are solved using tailored nonsmooth optimization techniques. Obviously the full details are outside the scope of this paper, and we refer the reader to [5], [31] for comprehensive

expositions on line-search-based methods and to [32] and references therein for those adopting a trust region strategy. The reader is also referred to [33] for alternative techniques.

V. Examples

A. Lightly damped plant with parametric uncertainties

The example below is an excerpt from [30] which has been modified to yield an unstable nominal system. This is a somewhat contrived example as it has been built to stress issues due to (real) parametric uncertainties. The uncertain plant is described as the second-order transfer function

$$G(s,\delta) := \frac{s^2 + 2\zeta\omega_n s + \omega_n^2(1+\delta_1)}{s^2 - 2\zeta\omega_n s + \omega_n^2(1+\delta_2)}, \qquad (13)$$

where $\delta_1,\ \delta_2$ are uncertain real parameters, i.e. $\delta_1,\ \delta_2 \in \mathbb{R}$ and $\omega_n = 1, \zeta = 0.2$. Such descriptions are typical of lightly damped modes with uncertain natural frequency. This can be rearranged into the LFT format of figure 1 upon defining an appropriate $P(s)$ and and $\Delta = \mathrm{diag}(\delta_1,\ \delta_2)$.

The aim in this illustration is the synthesis of a robust controller maximizing the parametric margin, that is, the radius of the square in the $(\delta_1,\ \delta_2)$-space where the closed-loop system remains stable. Note that for $(\delta_1,\ \delta_2) = (-1,\ -1)$, system (13) has an unstable non-minimal pole at $s = 0$. We infer the maximum reachable parametric margin for this system is bounded above by one no matter the technique used.

Results are collected in table I where various techniques are compared. The classical $D-K$ and $D-G,\ K$ iteration schemes with curve fitting and the nonsmooth approach as discussed in section III. The nonsmooth method is initialized from scratch using random generation of state-space systems, the 'rss' function from Matlab. Both techniques are run in default mode. Various controller orders were prescribed for the nonsmooth technique and we have also included the case of a proportional derivative controller denoted 'PD' in the table. Multiplier state dimensions were set to 4 for each uncertainty channel. The order of the overall controller $K(s)$ is therefore $\dim x_C + 4 + 4$, where x_C designates the state of $C(s)$.

TABLE I

ACHIEVED PARAMETRIC MARGINS WITH DIFFERENT SYNTHESIS TECHNIQUES

technique	controller order for $C(s)$	parametric margin
$D-K$ complex μ-synthesis	10	0.59
$D-G, K$ real μ-synthesis	26	0.82
$D-G, K$ reduced	14	0.82
nonsmooth real μ-synthesis	0	0.999
nonsmooth real μ-synthesis	1	0.999
nonsmooth real μ-synthesis	2	0.999
nonsmooth real μ-synthesis	3	0.999
nonsmooth real μ-synthesis	PD	0.999

We refer the reader to the full version of the paper for a detailed description of both the controllers and the multipliers. As clarified earlier in section III, the order of $C(s)$ is chosen independently of the multiplier orders which is an advantage in realistic applications. We note the $D-G, K$ controller can be reduced to order 14 without perceptible deterioration of robustness but further reduction results in unstable closed-loop systems.

B. Plant with a repeated parameter

A major impediment in μ synthesis problems is when repeated scalar blocks are present. Repeated scalar blocks have in correspondance full-block multipliers which entails substantial size inflation in terms of decision variables. Another issue with the conventional approach is that curve fitting is essentially a SISO procedure which means all entries of D and G scalings must be fitted. The overall scaling orders accumulate those of the individual entries so that subsequent synthesis phases become numerically challenging or even may succumb. In the sequel, we consider an example of this type and we show the nonsmooth approach holds promise to overcome the difficulty.

We consider an example borrowed from [34]. Parametric uncertainties are restricted to $\rho \in [-\pi/4,\ \pi/4]$. A reasonably accurate LFT approximation of the model is obtained by using 3rd-order Taylor series expansion of trigonometric functions. This yields an LFT representation with a repeated real scalar block of size 12. Performance is expressed as minimizing effects of disturbance f on measurements y in the L_2-gain sense. Overall the uncertainty structure can be written as $\Delta := \mathrm{diag}(\delta I_{12},\ \Delta_P)$ with Δ_P a (complex) performance block and δ a normalization of ρ to the unit interval $[-1,\ 1]$.

Conventional mixed-μ synthesis techniques based on curve fitting quickly succumb on such problems as they are facing augmented plants of orders 212 and 462 if restricting $D-G$ scalings to orders 1 and 2, respectively.

Our resolution strategy is based on program (12). We have explored various multiplier and controller orders with the nonsmooth technique and results are gathered in table II.

TABLE II

ACHIEVED PARAMETRIC MARGINS WITH VARIOUS MULTIPLIER/CONTROLLER COMBINATIONS

multiplier order for $M(s)$	controller order for $C(s)$	mixed-μ margin
static	4	0.95
1	4	1.53
2	5	1.67

Pure H_∞-synthesis yields a mixed-μ margin of 0.75 and falls short of achieving robust performance. In the first row of table II, the nonsmooth synthesis was conducted with static multipliers which implicitly assumes that uncertainties are time-varying $\rho = \rho(t)$. The second and third rows involves multipliers and controllers of increasing complexity. In both cases robust performance has been achieved beyond

the whole range of parameter variations. Note that mixed-μ related computations were double-checked using the skew-μ toolbox [35].

VI. Conclusion

Complex- and mixed-μ syntheses remains very challenging problems if sizeable practical applications is the aim. In this work, we have proposed a different line of attack where complex- and mixed-μ syntheses based on upper bounds are formalized as structured controller design problems which are then solved using a specialized nonsmooth technique with local convergence certificate. The nonsmooth approach allows to bypass a number of issues attached to more conventional methods and seems to offer a valid alternative according to preliminary testing.

Acknowledgement

This work was supported by research grant *TECHNICOM* from la Fondation d'Entreprise EADS & ONERA.

References

[1] O. Toker and H. Ozbay, "On the NP-hardness of the purely complex μ computation, analysis/synthesis, and some related problems in multidimensional systems," in *Proc. American Control Conf.*, 1995, pp. 447–451.

[2] M. K. H. Fan, A. L. Tits, and J. C. Doyle, "Robustness in the presence of mixed parametric uncertainty and unmodeled dynamics," *IEEE Trans. Aut. Control*, vol. AC-36, no. 1, pp. 25–38, Jan. 1991.

[3] S. Boyd, L. ElGhaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in Systems and Control Theory*, ser. SIAM Studies in Applied Mathematics. Philadelphia: SIAM, 1994, vol. 15.

[4] G. J. Balas, J. C. Doyle, K. Glover, A. Packard, and R. Smith, *μ-Analysis and synthesis toolbox : User's Guide*. The MathWorks, Inc., 1991.

[5] P. Apkarian and D. Noll, "Nonsmooth H_∞ synthesis," *IEEE Trans. Aut. Control*, vol. 51, no. 1, pp. 71–86, 2006.

[6] M. Fukuda and M. Kojima, "Branch-and-cut algorithms for the bilinear matrix inequality eigenvalue problem," *Comput. Optim. Appl.*, vol. 19, no. 1, pp. 79–105, 2001.

[7] K. C. Goh, M. G. Safonov, and G. P. Papavassilopoulos, "Global optimization for the biaffine matrix inequality problem," *Journal of Global Optimization*, vol. 7, no. 4, pp. 1573–2916, 1995.

[8] A. Simoes, P. Apkarian, and D. Noll, "A nonsmooth progress function for frequency shaping control design," *IET Control Theory & Applications*, vol. 2, no. 4, pp. 323–336, April 2008.

[9] F. H. Clarke, *Optimization and Nonsmooth Analysis*, ser. Canadian Math. Soc. Series. New York: John Wiley & Sons, 1983.

[10] W. M. Lu, K. Zhou, and J. C. Doyle, "Stabilization of LFT Systems," in *Proc. IEEE Conf. on Decision and Control*, Brighton, England, 1991, pp. 1239–1244.

[11] S. Skogestad and I. Postlethwaite, *Multivariable feedback design - analysis and design*. Wiley, 1996.

[12] D. McFarlane and K. Glover, "A loop shaping design procedure using H_∞ synthesis," *IEEE Trans. Aut. Control*, vol. 37, no. 6, pp. 759–769, 1992.

[13] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*. Prentice Hall, 1996.

[14] J. Doyle, A. Packard, and K. Zhou, "Review of LFT's, LMI's and μ," in *Proc. IEEE Conf. on Decision and Control*, vol. 2, Brighton, Dec. 1991, pp. 1227–1232.

[15] G. Meinsma, Y. Shrivastava, and M. Fu, "A dual formulation of mixed mu and the losslessness of (D,G)-scaling," *IEEE Trans. Aut. Control*, vol. 42, no. 7, pp. 1032–1036, 1997.

[16] A. Packard and J. Doyle, "The complex structured singular value," *Automatica*, vol. 29, no. 1, pp. 71–109, 1993.

[17] P. M. Young, "controller design with mixed uncertainties," *Proc. American Control Conf.*, vol. 2, pp. 2333– 2337, 1994.

[18] M. G. Safonov and R. Y. Chiang, "Real/complex K_m-synthesis without curve fitting," in *Control and Dynamic Systems*, C. T. Leondes, Ed. New York: Academic Press, 1993, vol. 56, pp. 303–324.

[19] R. Y. Chiang and M. G. Safonov, "Real K_m-synthesis via generalized Popov multipliers," in *Proc. American Control Conf.*, vol. 3, Chicago, Jun. 1992, pp. 2417–2418.

[20] R. J. Adams and S. S. Banda, "Robust Flight Control Design Using Dynamic Inversion and Structured Singular Value Synthesis," *IEEE Trans. Circuits Syst.*, vol. 1, no. 2, pp. 80–92, June 1993.

[21] A. Packard, J. C. Doyle, and G. J. Balas, "Linear, multivariable robust control with a μ perspective," *ASME Journal on Dynamics, Measurements and Control, Special Edition on Control*, vol. 115, no. 2b, pp. 426–438, Jun. 1993.

[22] J. Reiner, G. Balas, and W. Garrard, "Design of a flight control system for a highly maneuverable aircraft using μ synthesis," in *AIAA Guid., Nav., and Control Conf.*, Monterey, CA, Aug. 1993, pp. 710–719.

[23] J. C. Doyle, K. Lenz, and A. Packard, "Design examples using μ-synthesis: space shuttle lateral axis FCS during reentry," in *Proc. IEEE Conf. on Decision and Control*, Athens,Greece, 1986, pp. 2218–2223.

[24] P. Apkarian, D. Noll, and O. Prot, "A trust region spectral bundle method for nonconvex eigenvalue optimization," *SIAM J. on Optimization*, vol. 19, no. 1, pp. 281–306, 2008.

[25] P. Apkarian and D. Noll, "Nonsmooth optimization for multidisk H_∞ synthesis," *European J. of Control*, vol. 12, no. 3, pp. 229–244, 2006.

[26] N. A. Bruinsma and M. Steinbuch, "A fast algorithm to compute the H_∞-norm of a transfer function matrix," *Syst. Control Letters*, vol. 14, no. 5, pp. 287–293, 1990.

[27] S. Boyd and V. Balakrishnan, "A regularity result for the singular values of a transfer matrix and a quadratically convergent algorithm for computing its L_∞-norm," in *Proc. IEEE Conf. on Decision and Control*, Dec. 1989, pp. 954–955.

[28] A. Helmersson, "Applications of mixed μ-synthesis using the passivity approach," Dept of EE. Linkping University, SE-581 83 Linkping, Sweden, Tech. Rep. LiTH-ISY-R-1688, Feb. 1994.

[29] J. Ly, M. G. Safonov, and R. Y. Chiang, "Real/complex multivariable stability margin computation via generalized Popov multiplier - LMI approach," in *Proc. American Control Conf.*, Baltimore, Ma, Jun. 1994, pp. 425–429.

[30] P. M. Young, "Controller design with real parametric uncertainty," *Int. J. Control*, vol. 65, no. 3, pp. 469–509, 1996.

[31] P. Apkarian and D. Noll, "Nonsmooth optimization for multiband frequency domain control design," *Automatica*, vol. 43, no. 4, pp. 724–731, April 2007.

[32] D. Noll, O. Prot, and P. Apkarian, "A proximity control algorithm to minimize nonsmooth and nonconvex semi-infinite maximum eigenvalue functions," *Jounal of Convex Analysis*, vol. 16, no. 3 & 4, pp. 641–666, 2009.

[33] S. Gumussoy, M. Millstone, and M. L. Overton, "H_∞ strong stabilization via HIFOO, a package for fixed-order controller design," in *Proc. IEEE Conf. on Decision and Control*, Cancun, Mexico, 2008, pp. 4135–4140.

[34] F. Wu, X. Yang, A. Packard, and G. Becker, "Induced L_2 norm control for LPV systems with bounded parameter variation rates," *Int. J. Robust and Nonlinear Control*, vol. 6, no. 9, pp. 983–998, 1996.

[35] G. Ferreres and J. Biannic, "A Skew Mu Toolbox (SMT) for robustness analysis," *available on the authors' homepages*, 2003-2009.

Object Tracking in Real Time Video Sequences Using a Fast Level Set Method

Bogdan Apostol and Vasile Manta

Abstract— **We propose a method for determining the shape (2D outline of the object) and follow-up position (set of simple geometric transformations on position), using a level set based curve evolution and combining the benefits of using the pixel-wise posterior term with a fast level-set algorithm to approximate curve evolution. The pixel-wise posterior term allows us to marginalize the model parameters at pixel level, and the fast level-set algorithm avoids the need of solving the time consuming partial differential equations (PDEs). Our proposed implementation can accurately process a higher number of frames per second, bringing real-time performance on standard hardware systems.**

I. Introduction

OBJECT tracking and shape adaptation, according to changes in the appearance of moving objects and/or camera movement, is an intensely studied topic of various applications in computer vision. The process of finding the contour and adapting the position of an object in the scene in video sequences is composed of image segmentation and pose geometric transformations.

The basic idea in active contour based segmentations (introduced by Kass, Witkins and Terzopoulos [1]) is to evolve a curve, under the influence of image data constraints, in order to separate the relevant objects from the background. We start with an initial approximation of the segmentation: a curve around the object. This moves in the normal direction and stops at the object contour. Classical snake models involve an edge detector to stop the evolving curve at the border of the object.

As shown in [2], the snake model has the following problems: it doesn't allow the contour to easily undergo topological changes, it lacks a probabilistic interpretation and can it be quite sensitive at initialization (the algorithm tends to get stuck in undesirable local minima).

We represent the contour of the object as the zero level set of an implicit function ϕ defined in a higher dimensional space, usually mentioned as the *level set function*, and evolve the curve according to a partial differential equation (PDE). This approach has the advantage that the level set can separate and merge naturally during evolution, thus topological changes are handled automatically. The level set formulation also permits us the use of statistical region-based methods, thus there are less local minima in the image data and the segmentation schemes are far less sensitive to noise and to initialization varying.

A region based method models the foreground and the background membership probabilities. The standard method of Mumford and Shah is extended in [3] where a per pixel log likelihood is computed from the per region histogram. Another method is that of [4] where a per pixel posterior term is computed from the foreground and background aspect models. This method provides better results than [2, 3] as it allows us to marginalise out model parameters at a pixel level and allow the shape to change online.

In this paper, we combine the benefits of using the pixel-wise posterior term [4], as opposed to a likelihood, with the benefits of using an algorithm for the approximation of level-set-based curve evolution [5]. This, unlike [4], avoids the need to solve the PDE at every step and thus speeds up the algorithm.

Similar to [4], we define a probabilistic framework based on two aspect models, respectively the foreground model and background model. Our models are represented by RGB histograms using 32 bins per channel. We initialize the models using a user imputed region of interest. The values of the pixels inside the region of interest are used to build the foreground model, and the values of the pixels outside the region of interest are used to build the background model. The two initial distributions are then used to generate an initial segmentation, and then to rebuild the probabilistic model. This step is repeated until the shape converges. Similar to [4] we adapt the foreground and background models online, by using a linear model with learning rates.

The level-set implementation proposed in [4] is still slow on standard hardware, because the curve evolution process is based on the solution of certain PDEs. This takes significant computational time for each curve evolution step. In this paper we use the fast level set approximation [3], where two linked lists are kept that represent a narrow band of one pixel width, around the actual contour. The level set evolution is simply a switching mechanism between the two lists.

The outline of this paper is as follows: Section II describes the use of level-set segmentation and pixel-wise

Manuscript received May 12, 2010.

Bogdan Apostol is with the Faculty of Automatic Control and Computer Engineering, Technical University "Gheorghe Asachi" Iasi, Str. Prof. dr. doc. Dimitrie Mangeron, nr. 27, Iasi, cod 700050, Romania (e-mail: apostol_bgdn@yahoo.com).

Vasile Manta is with the Faculty of Automatic Control and Computer Engineering, Technical University "Gheorghe Asachi" Iasi, Str. Prof. dr. doc. Dimitrie Mangeron, nr. 27, Iasi, cod 700050, Romania (e-mail: vmanta@cs.tuiasi.ro).

posteriors; Section III outlines the implementation details and shows the experimental results of the method both in real-time videos and post-processed image sequences and Section IV concludes with a summary and discussion of further development possibilities.

II. Theoretical Considerations

A. Level Set Method

The main idea of the level set methodology is to embed the propagating curve as the zero level of Lipchitz continuous functions ϕ. Let $\{(x,y)\in\Re^2:\phi(x,y)=0\}$ be the embedding function for the level set function. If we associate a velocity field $\mathbf{v}$ and by limiting it to the curve we determine the speed of the curve, then at least for a moment time *t*, the evolution can be described with a Cauchy approach [8]:

$$\phi_t+\mathbf{v}\cdot\nabla\phi=0,\qquad \phi(\mathbf{x},0)=\phi_0(\mathbf{x}),\quad (1)$$

where ϕ_0 embeds the initial curve position.

Enforcing the $\gamma(s,t)$ parameterised curve to be the zero level of function ϕ in equation (1), independent of the time variable, so that $\phi(\gamma(s,t),t)=0$, whatever $t\geq 0$, we can write:

$$\phi_t+\frac{\partial\gamma}{\partial t}\cdot\nabla\phi=0,\ (2)$$

where $\frac{\partial\gamma}{\partial t}$ is the known dynamics of the curve.

Extending $\frac{\partial\gamma}{\partial t}$ continuously throughout the entire domain will create such a field of velocities. Generally, the velocity field $\mathbf{v}$ can be a function of position and other geometric quantities of the curve. We can rewrite equation (2) using the normal velocity:

$$\phi_t+v_n|\nabla\phi|=0,\quad v_n=\mathbf{v}\cdot\frac{\nabla\phi}{|\nabla\phi|}.\quad (3)$$

In the level set formulation, the surface integral of a function *f* along the zero level is defined by [8]:

$$\int_{\Re^d} f(x)\delta(\phi)|\nabla\phi|dx,\quad (4)$$

where $\delta(\phi)$ is the delta Dirac function.

If $f\equiv 1$, the result of this integral is the arc length for curves in two-dimensional spaces, the surface area in three-dimensional space and equation (4) becomes:

$$\int_{\Re^2}\delta(\phi)|\nabla\phi|dx.\quad (5)$$

Volume integrals are defined as:

$$\int_{\Re^3} H(\phi)|\nabla\phi|dx\quad (6)$$

where H is defined as the step function:

$$H(\phi)=\begin{cases}1, & \text{for } x\geq 0\\ 0, & \text{for } x<0\end{cases}\quad (7)$$

1) Reshaping the level set function. The level set function develops steep and smooth gradients, generating problems for numerical approximations, and it becomes necessary to reshape the level set function in a usable form, but without changing the zero level location. This means reshaping the level set around the interface and leaving the interface as is.

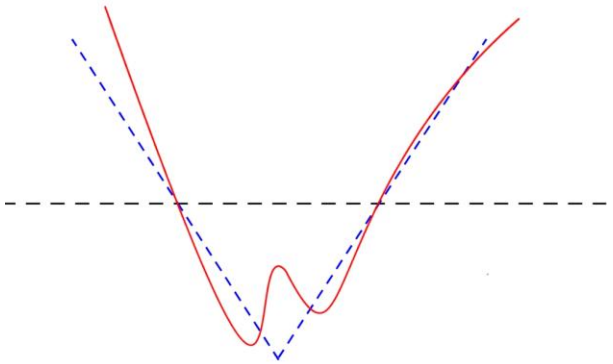


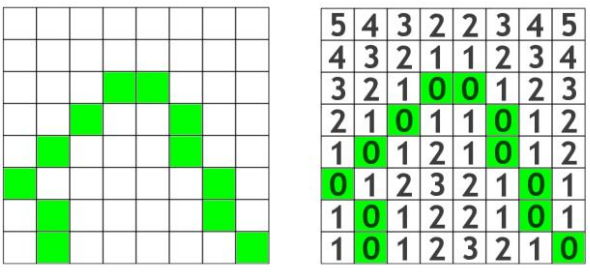
Fig. 1. Reshaping the function

A method of remodelling is reinitializing the distance by evolving the PDE function (partial equations) to a steady state [8]:

$$\phi_t+\text{sgn}(\phi_0)(|\nabla\phi|-1)=0,\ \phi(\mathbf{x},t=0)=\phi_0(\mathbf{x}).\ (8)$$

If it evolves towards a steady state solution on the computational domain, the solution of ϕ becomes the signed distance transform function for the interface $\{\phi_0=0\}$. In the region where ϕ_0 is positive, ϕ_t <0 when $|\nabla\phi|$ > 1 and therefore, the value of ϕ will decrease and thus $|\nabla\phi|$ will become closer to 1. We note that ϕ_t = 0 when ϕ_0 = 0, because $\text{sgn}(\phi_0)=0$.

Standard level set uses a common numerical method, which implies the reshaping the distance form ϕ (Fig.1.)



(a) (b)

Fig. 2. Distance transform method.
(a) $\phi=0$ interface, (b) Value of each pixel as the distance to the nearest pixel on interface

with a higher order more accurate method for a very short time interval, so that a small band around the $\phi = 0$ interface. This method is called the distance transform function (Fig.2.) and the values of ϕ inside the band become the values of the distance transform.

B. Fast Approximation of Level Set

The level set method represents the interface as the zero level of a function ϕ, defined over a regular grid D of k dimensions. We assume the grid is sampled uniformly and the default sampling is of uniform distance 1. The coordinates of a point in the grid are given as $\mathbf{x} = \{x_1, x_2, \ldots, x_k\}$. We define the list of inside neighbouring grid points in the band L_{inside} and outside neighbouring grid points in the band $L_{outside}$ for the object regions as follows:

$$L_{inside} = \{\mathbf{x},\ \mathbf{x} \in \Omega \text{ and } \exists \mathbf{y} \in N(\mathbf{x}) \text{such that } \mathbf{y} \in D \setminus \Omega\}$$
$$L_{outside} = \{\mathbf{x},\ \mathbf{x} \in D \setminus \Omega \text{ and } \exists \mathbf{y} \in N(\mathbf{x}) \text{such that } \mathbf{y} \in \Omega\} \quad (9)$$

where $N(\mathbf{x})$ is a discrete neighbourhood of $\mathbf{x}$.

The interface moves inward or outward and can be split into two curves, as the value of ϕ changes from positive to negative. To move the curve in one or the other direction one needs to solve the PDE equations, which are time and resource consuming.

The idea that [5] proposes is to only take in account the border between the object and the background area, and to achieve the same result if we use the relation between the two lists, L_{inside} and $L_{outside}$. This achieves the movement of the curve using minimal computation, by simply moving a grid point from one list to another, L_{inside} to $L_{outside}$ if the curves moves outward and $L_{outside}$ to L_{inside} if the curves moves inward.

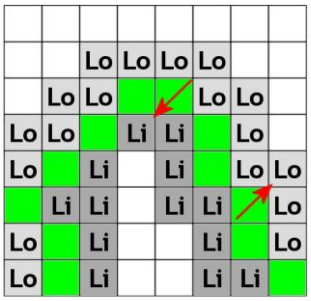


Fig. 3. Moving the curve by moving points from L_{inside} to $L_{outside}$ and from $L_{outside}$ to L_{inside}

The level set function values are locally approximated to a signed distance transform:

$$\hat{\phi}(\mathbf{x}) = \begin{cases} 3, & \text{for } \mathbf{x} \text{ outside } C, \mathbf{x} \notin L_{outside} \\ 1, & \text{for } \mathbf{x} \in L_{outside} \\ -1, & \text{for } \mathbf{x} \in L_{inside} \\ -3, & \text{for } \mathbf{x} \text{ inside } C, \mathbf{x} \notin L_{inside} \end{cases}, \quad (10)$$

and the evolution speed F is represented by an integer-value array $\hat{F}(\mathbf{x})$ and is defined by:

$$\hat{F}(\mathbf{x}) = \begin{cases} 1, & F(\mathbf{x}) > 0, \\ 0, & F(\mathbf{x}) = 0, \\ -1, & F(\mathbf{x}) < 0. \end{cases} \quad (11)$$

To keep the smoothness of the curve, [5] proposes a two step algorithm based on the fact that the evolution speed F can be split in two components. The first one, F_d, is a data-dependent speed, which is a function of the image data and depends on up to the first order geometric properties of the curve, and the second one, F_{int}, is a smoothing regularization speed, which is proportional to the mean curvature. The two component speeds must keep:

$$F(\mathbf{x}) = F_d(\mathbf{x}) + F_{\text{int}}(\mathbf{x}). \quad (12)$$

The data-dependent speed F_d is represented by the integer-value array $\hat{F}_d$, and the smoothing speed F_{int} is represented by the integer-value array $\hat{F}_{\text{int}}$, using the sign representation in equation (11).

1) Fast two-cycle algorithm. This subsection will describe the four steps of the algorithm corresponding to evolving the curve using the sign of the data-dependent speed $\hat{F}_d$ and respectively evolving the curve using the sign of smoothing speed $\hat{F}_{\text{int}}$.

Step 1: initializes the arrays $\hat{\phi}$, $\hat{F}_d$, $\hat{F}_{\text{int}}$ and the lists L_{inside} and $L_{outside}$, using equations (10), (11), (12).

Step 2: represents cycle one, data-dependent speed evolution:

- compute F_d in each point stored in the two lists, L_{inside} and $L_{outside}$, and store it's sign in $\hat{F}_d$;
- evolve the curve outward and copy each point $\mathbf{x} \in L_{outside}$ in L_{inside} if $\hat{F}_d(\mathbf{x}) > 0$, and then eliminate duplicates in L_{inside};
- evolve the curve inward and copy each point $\mathbf{x} \in L_{inside}$ in $L_{outside}$ if $\hat{F}_d(\mathbf{x}) < 0$, and then eliminate duplicates in $L_{outside}$;
- check stopping condition and if satisfied continue with step 3, otherwise restart step 2.

Step 3: represents cycle two, smoothing speed evolution using Gaussian filtering:

- compute F_{int} in each point stored in the two lists, L_{inside} and $L_{outside}$, and store it's sign in $\hat{F}_{\text{int}}$;
- evolve the curve outward and copy each point $\mathbf{x} \in L_{outside}$ in L_{inside} if $\hat{F}_{\text{int}}(\mathbf{x}) > 0$, and then eliminate duplicates in L_{inside};
- evolve the curve inward and copy each point $\mathbf{x} \in L_{inside}$ in $L_{outside}$ if $\hat{F}_{\text{int}}(\mathbf{x}) < 0$, and then eliminate duplicates in $L_{outside}$;

Step 4: if stopping condition from step 2 not satisfied then return to step 2.

C. Fast implementation of PWP

Traditional region based segmentations compute the overall likelihood $P(I \mid M)$ as the product of the pixel-wise likelihoods functions in all grid points $P(I \mid M) = \prod_{i=1}^{N} P(I(\mathbf{x_i}) \mid M_i)$. Using the notations:

- $\mathbf{x_i} = (x_i, y_i)$ a pixel in the image,
- $M = \{M_f, M_b\}$ foreground model or background model,
- $I(\mathbf{x_i})$ the image pixel value,
- $P(y \mid M_f)$ foreground model over pixel values y (RGB value),
- $P(y \mid M_b)$ background model over pixel values y (RGB value),

we define the foreground probability P_f of a pixel, and the background probability P_b as in [4] :

$$P_f = \frac{P(y \mid M_f)}{\eta_f P(y_i \mid M_f) + \eta_b P(y_i \mid M_b)}, \qquad P_b = \frac{P(y \mid M_b)}{\eta_f P(y_i \mid M_f) + \eta_b P(y_i \mid M_b)} \tag{13}$$

where:

$$\eta_f = \sum_{i=1}^{N} H_\varepsilon(\phi(\mathbf{x_i})) \text{ and } \eta_b = \sum_{i=1}^{N} (1 - H_\varepsilon(\phi(\mathbf{x_i})))$$

and H_ε is the blurred Heaviside step function used in most level-set methods.

Next we define the data-dependent speed F_d from equation (12) as the difference between P_f and P_b :

$$F_d(\mathbf{x}) = P_f(\mathbf{x}) - P_b(\mathbf{x}) \tag{14}$$

$$\begin{aligned} \hat{F}_d(\mathbf{x}) &\geq 0, \quad \forall \mathbf{x} \in L_{inside} \\ \hat{F}_d(\mathbf{x}) &\leq 0, \quad \forall \mathbf{x} \in L_{outside} \end{aligned} \tag{15}$$

The smoothing speed $\hat{F}_{\text{int}}$ is defined as follows for all the boundary points in the $L_{outside}$ list:

$$\hat{F}_{\text{int}}(\mathbf{x}) = \begin{cases} 1, & \text{if } G \otimes H_\varepsilon > \frac{1}{2}, \\ 0, & \text{otherwise} \end{cases}, \tag{16}$$

where H_ε is the blurred Heaviside step function, G is the Gaussian filter and $\otimes$ is the convolution operation.

The smoothing speed $\hat{F}_{\text{int}}$ is defined as follows for all the boundary points in the L_{inside} list:

$$\hat{F}_{\text{int}}(\mathbf{x}) = \begin{cases} 1, & \text{if } G \otimes H_\varepsilon < \frac{1}{2}, \\ 0, & \text{otherwise} \end{cases}, \tag{17}$$

D. Visual tracking

As it is shown in [4] it is possible to treat the tracking problem directly in a segmentation framework. For visual tracking of the object we will use a general geometric model, the notation $\mathbf{x}_0 = [x_0, y_0]^T$ for pixel image position before applying the geometric transformations, and $\mathbf{x} = [x, y]^T$ for pixel image position after applying the geometric transformations.

Using homogeneous coordinates we can describe a general model for possible transformations in an image:

$$\mathbf{x} = T(t_x, t_y) \cdot R(\theta) \cdot S(s_x, s_y) \cdot \mathbf{x}_0, \tag{18}$$

where $T(t_x, t_y)$ is the translation in both x and y directions, $R(\theta)$ is the rotation of θ angle and $S(s_x, s_y)$ is the scaling according to the two axes x and y .

Detailing the model in equation (18):

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix} \tag{19}$$

In this paper we will use the unit matrix for the scale and rotation matrixes, because they are automatically handled by adapting the curve on-line. Thus, we simplify equation (19) and define the model as:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix} \quad (20)$$

and $p = \{t_x, t_y\}$ the positioning parameters of the object.

Differencing the energy function, as shown by [6], regarding the positioning parameters p_i, we evolve the curve in a space defined by the positioning parameters:

$$\frac{\partial P(\phi \mid \Omega)}{\partial p_i} = P(\phi \mid \Omega) \Sigma \frac{P_f - P_b}{H_\varepsilon(\phi) P_f + (1 - H_\varepsilon(\phi) P_b)} \cdot \frac{\partial H_\varepsilon(\phi)}{\partial p_i} \quad (21)$$

where

$$\frac{\partial H_\varepsilon(\phi(x_i, y_i))}{\partial p_i} = \delta_\varepsilon(\phi) \begin{bmatrix} \frac{\partial \phi}{\partial x_i} & \frac{\partial \phi}{\partial y_i} \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial x_i}{\partial p_i} \\ \frac{\partial y_i}{\partial p_i} \end{bmatrix}$$

and $i = \{1,2\}$.

E. Online Learning

The online adaptation of the histogram models is done using the linear learning model proposed by [4]:

$$P_t(y \mid M_i) = (1 - \alpha_i) P_{t-1}(y \mid M_i) + \alpha_i P_t(y \mid M_i), \quad (22)$$

where $i = \{f, b\}$ and α_i is the learning rate.

Thus, if the learning rate value is small, even if the current pixel distribution does introduce errors, they will be corrected by previous histogram models.

III. Implementation And Experimental Results

This section will present the implementation details and some of the qualities of the method, such as flexibility in tracking different kinds of objects, resilience to noise and high real time performance on a standard hardware system. We will also show the invariance of the implementation to changes in appearance and camera angle. Finally we will make a comparative study with the PWP implementation proposed by [4], varying the segmentation parameters, the limit number of iterations per frame, and the online learning rates.

The aspect models in equation (13) are computed using the total area of the image and not restricted only on the region of interest, as the curve is. Thus for the foreground histogram model all the pixels that have a positive level-set value are taken in account. For the background histogram model, in contrast to [4], we take in account both the negative level-set value pixels and the pixels outside the region of interest. This is shown in Fig.4.

Unless otherwise specified, we use the following parameter values for the learning rates of the two histogram models in equation (22): $\alpha_f = 0.001$ and. $\alpha_b = 0.002$.

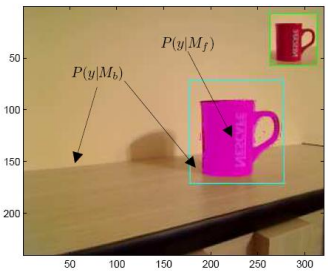


Fig. 4. Foreground and background aspect models

Our implementation implies an initial user registration, by placing a square region of interest (ROI) around the object and an initial segmentation process. Thus we begin with a circular contour and evolve the curve until we reach partial stability. Stability is reached fast because, instead of computing the PDE function for every step of the initial segmentation, we use the fast-level-set algorithm evolution speed in equation (12) to move the curve inward or outward. The foreground and background aspect models are also adapted online in this step (equation (22)) and then used to compute the data-dependent speed using equation (14).

Next we compute the smoothing speed in equations (16) and (17) and move the curve by simply moving pixels between the inside list and the outside list depending on the orientation of the evolution speed.

The translation of the contour and the region of interest is done by obtaining the position parameter $p = \{t_x, t_y\}$ in equation (21). For each step in the tracking algorithm we evolve the curve and adapt the RGB histogram models online. Because segmentation is now an easy process we can compute several segmentation steps for each tracking step.

The application described in this paper has been tested mostly on live video sequences, but also on a set of recorded sequences and then processed frame by frame. These sequences contain: objects which show rapid movements, posture changes regarding the video camera, camera moving towards and outwards, scene illumination variations and image noise.

The proposed implementation was tested on computer generated video sequences (three dimensional computer scenes) and on camera feed images. In testing on video sequences obtained by real-time camera feed, we used a Windows Vista compatible web camera, which uses the USB 2.0 connectivity. In this section from now on when we refer to the video source used in experiments, we refer to a Logitech QuickCam Pro 9000, which captures images at a resolution of 640x480, and a capture rate of 15 frames per second. The application was written in C++ programming

language and uses a managed user interface in C# to adjust the parameters and register the results. It was tested on an Intel Core2Duo 2.16GHz machine with an ATI Radeon 3470 video card.

First we present a qualitative evaluation for two image segmentations, Fig.5. The inner surface of the contour is displayed in magenta color and is overlaid over the real model in the image. The first set of frames (Fig.5.a, Fig.5.b, Fig.5.c) represent the curve evolution until it reaches the real mug contour. For this experiment we choose a 100x100px region of interest and an initial circle contour of 80px diameter that was placed in the middle of the region of interest. In the second set of frames (Fig.5.d, Fig.5.e, Fig.5.f) we set the curve evolution for segmented a more complicated object as a hand. For this experiment we choose a 130x130px region of interest and an initial circle contour of 100px diameter that was placed in the middle of the region of interest.

Note that although the hand color distribution is close to the color of the background the contour does not evolve beyond the real edges of the hand figure. We also see the benefits of using a fast level-set implementation as we can observe the quick merging of the contour in both cases, creating a complete and stable contour over the edges of the objects. Because we use a two list band representation (L_{inside} and $L_{outside}$) isolated contours cannot appear as in the classical level-set implementation, where isolated regions can build up and pass through the zero level, thus creating isolated contours.

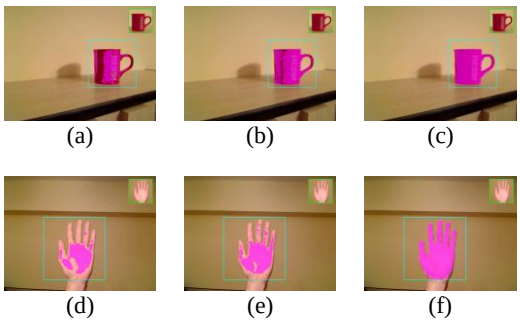


Fig. 5. Curve evolution in image segmentation. (a)-iteration 10, (b)-iteration 70, (c)-iteration 120, (d)-iteration 10, (e)-iteration 50, (f)-iteration 360.

In Fig.6 we selected a few frames obtained by real-time processing of a webcam image sequence. Note the contour evolution and on-line adaptation in the sequence of finding and tracking the human face, and that in time parts that were not segmented initially become part of the contour. Also, in the mug tracking sequence we see how the algorithm deals with scale adaptation (between (Fig.6.b) and (Fig.6.c))

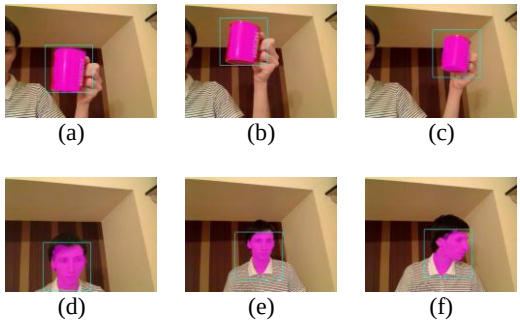


Fig. 6. Visual tracking and contour evolution for a real scene: (a, b, c) – tracking the map, (d, e, f) – tracking human face.

IV. Conclusion

In this paper, we have proposed a method of combining the benefits of using the pixel-wise posterior term, by evolving the curve depending on two Bayesian models, with the benefits of a fast-level-set based algorithm, that does not require us to solve the partial differential equations but rather approximate it. The pixel-wise-posterior term allows us to marginalize model parameters at pixel level, and the fast-level-set implementation permits us to evolve the curve by simply moving a pixel from one list to another. Thus, we obtain a highly accurate algorithm that can easily process more than 20 frames per second, using a web-camera live feed on a standard hardware system. Further algorithm development primarily involves increasing the processing by running jobs in parallel and by implementing the jobs directly on the Graphics Processing Unit.

References

[1] M. Kass, A.P.Witkin and D. Terzopoulos, “Snakes: Active contour models.” 1(4):321–331, January 1988.

[2] Daniel Cremers, Mikael Rousson and Rachid Deriche, “A review of statistical approaches to level set segmentation: Integrating colour, texture, motion and shape.” *Int. J. Comput. Vision*, 72(2):195–215, 2007.

[3] Luminita A. Vese, Tony F. Chan, “A multiphase level set framework for image segmentation using the mumford and shah model”. *International Journal of Computer Vision*, 50(3):271–293, December 2002.

[4] Charles Bibby and Ian Reid, “Robust real-time visual tracking using pixel-wise posteriors”.In *ECCV ’08: Proceedings of the 10th European Conference on Computer Vision*, pages 831–844, Berlin, Heidelberg, 2008. Springer-Verlag.

[5] Yonggang Shi, William Clem Karl, “A Real-Time Algorithm for the Approximation of Level-Set-Based Curve Evolution”. *IEEE Transactions on Image Processing*, vol. 17, no. 5, May 2008.

[6] Victor Prisacariu and Ian Reid, “PWP3D: Real-time segmentation and tracking of 3D objects”. *Proceedings of the 20th British Machine Vision Conference*, September 2009.

[7] Stanley J. Osher and Ronald P. Fedkiw, “Level Set Methods and Dynamic Implicit Surfaces”. *Springer*, October 2002.

[8] Richard Tsai and Stanley Osher, “Level set methods and their applications in image science”. *Bul. Comm. Math Sci*, 1:2003.

Parallel K-Means Revisited: A Hypercube Approach

Alexandru Archip, Vasile Manta and Gabriela Danilet

***Abstract*— Clustering represents the data mining process of partitioning or grouping a given set of objects into a set of disjoint clusters. The algorithms addressing this process are usually time consuming and often require efficient parallel models in order to achieve good response times. We present a new model for one of the most popular clustering algorithms - K-Means Clustering. This model focuses on using an efficient topology for MPI based approaches in order to reduce the communication times involved in data partitioning and in computing the results.**

I. Introduction

According to [1], [2], *clustering* is defined as a descriptive *data mining* technique that aims to divide a given set of objects into disjoint groups. The process of splitting the given data set is performed using the intrinsic similarity of the items with respect to a given set of *interesting attributes*. Clustering results should offer a better *representation* of the input data set, with respect to a given *similarity* metric (objects belonging to the same group resemble one another, while any two objects belonging to distinct groups should be significantly different). *Clustering methods* are generally considered as being a form of *unsupervised learning*[1], [2]. Given this perspective, Berkin concludes in [1] that resulting data represents a *hidden pattern* providing a new *representation* for the existing data.

Nowadays, an increasing number of applications employ clustering methods in order to achieve better results. Popular examples include document clustering [3] or information retrieval systems [4]. A common issue that must be solved for these applications is the increasing volume of data that needs to be processed. Clustering algorithms are mostly *lazy learners* and efficient parallel models should be used for obtaining reasonable response times and scalable applications.

This paper presents a study performed on one of the existing parallel models for the *K-Means Clustering* algorithm - *Parallel K-Means* [5] (PKMeans). The purpose of this study is to stress out the importance of choosing an adequate implementation technique in order to achieve good scalability. The following subsections review the general considerations regarding the target algorithm and the parallel model submitted to our tests. *Section II* presents a detailed analysis of the generic MPI approach of PKMeans given in [6]. *Section III* introduces the fundamentals of our approach regarding our own implementation of the parallel algorithm, while *section IV* depicts the test results we have obtained. The final section - *section V* - emphasizes the conclusion we have drawn from our tests and also presents some considerations on future tests and developments.

A. Archip is with The Faculty of Automatic Control and Computer Engineering, The "Gheorghe Asachi" Technical University of Iasi alexandru.archip@cs.tuiasi.ro

V. Manta is with The Faculty of Automatic Control and Computer Engineering, The "Gheorghe Asachi" Technical University of Iasi vmanta@cs.tuiasi.ro

G. Danilet is a MSc. student of The Faculty of Automatic Control and Computer Engineering, The "Gheorghe Asachi" Technical University of Iasi d_gabryelle86@yahoo.com

A. K-Means Clustering

K-Means Clustering (KMC) was first introduced by [7] in 1967. Currently, *KMC* is one of oldest and, according to [6], one the most popular clustering algorithms. The reasons for its popularity are the simplicity of implementation, the scalability and the speed of convergence. Also, provided that a adequate metric is used, the algorithm may be easily adapted to a variety of data types. *KMC* algorithm represents a partitioning method that attempts to split the N sized input data set into K partitions. The objects belonging to one partition should be similar to one another, while objects belonging to different partitions should be different. M. Joshi in [6], citing [5], presents the following generic stages for the *KMC* algorithm:

1) *Initialization*: select a set of k items from the input data set as the initial *centroids*
2) *Distance calculation*: for each *item* in the data set, compute the *distance* to each of the selected *centroids*; the *item* is assigned to the *closest centroid cluster*
3) *Centroid recalculation*: for each *cluster*, recompute the *centroid* as the average of the items assigned to it
4) *Convergence condition*: repeat *stage 2* and *3* until convergence is reached

From a logical perspective, the convergence condition for *stage 4* could be one of the following: *no items have been moved between the clusters* or *centroid coordinates have not changed in stage 3*. From a mathematical perspective, convergence is represented by the square distance computed according to (1). *Stage 4* may be in this case defined as obtaining the minimum value for [2]:

$$E = \sum_{k} \sum_{x_i \in C(k)} \|x_i - m_k\|^2. \quad (1)$$

An important note is that the *KMC* algorithm is a *greedy* approach for partitioning methods. For different selections of the initial *centroids* the minimum square distance obtained according to (1) may vary. Given k - the number of clusters, n - the number of items to be clustered and t - the number of iterations necessary to reach convergence, *KMC* algorithm has *O(nkt)* time complexity [2]. Under normal circumstances, the following relationship exists between n, k and t [2]:

$$k \ll n \text{ and } t \ll n. \quad (2)$$

Taking (2) into consideration, it can be easily noticed that the *total number of items n* is the factor with the greatest influence over the time response for any *KMC* implementation.

B. Parallel K-Means

Several attempts were made to efficiently parallelize the *KMC* algorithm. One of the most known and used models is *Parallel K-Means* (*PKMeans*) [5], [6], [8]. The most time consuming step is the distance calculation stage. This stage alone requires *O(nk)* steps in order to compute the distances between each item in the data set (comprised of *n* items) and each of the *k* selected clusters. Considering this observation and (2), the parallel model of *PKMeans* splits the entire input data set amongst processes[5], [6], [8]. Given *P* processes, each would receive *(n/P)* elements from the original input. This division of items would reduce the local stage of distance calculation to the time complexity given by:

$$O(\frac{n \cdot k}{P}). \quad (3)$$

The subsequent *stage 3* (presented in the previous section) implies that all processes participate in a reduction phase in order to determine the new centroids. M. Joshi summarizes in [6] the following steps for the *PKMeans*:

1) root process computes the initial means
2) initial centroids are replicated amongst all processes
3) each processor computes the distance of each local item to the given centroids
4) each processor assigns each local item to the closest centroid and local squared distance is computed
5) processes participate in a reduction phase in order to determine the global centroids and the global value for the squared distance.

II. Analysis of Parallel K-Means Implementations

The *PKMeans* model is based on a *Single Program Multiple Data* paradigm. One first important note is that this model does not apply only to a specific implementation method. Depending on various requirements, the *PKMeans* algorithm may be implemented using either *thread parallelism/shared memory* [9] or *process parallelism/distributed memory*[10], [11]. However, various authors argument that the latter should be preferred, especially when dealing with large volumes of data [6], [5], [8], [12] (such a case would be applying *KMC* based methods to text document clustering [2]).

When it comes to the actual implementation of *PKMeans* using a distributed memory scheme, the *Message Passing Interface* (MPI for short) standard seems to be the preferred approach [6], [12]. This choice is based on the following arguments [6]:

- MPI aims to be a *standard* for distributed memory, message passing parallel computing (currently, version 3 of the specifications are about to be released)
- it is portable (provided that applications are recompiled between various operating platforms), heterogeneous and provides good support for various communication topologies
- all parallelism is explicit i.e. the programmer is given complete freedom in implementing almost every form of parallelism using various MPI constructs
- there are a lot of implementations available (for example [10], [11], [13]) for all supported programming languages (such as C/C++ or FORTRAN).

A generic MPI approach is given in [12].

Algorithm 1 Generic MPI approach to PKMeans

1: $P := MPI\_Comm\_size()$
2: $id := MPI\_Comm\_rank()$
3: $MSE := largeValue$
4: **if** $id = root$ **then**
5: make initial selection for centroids $\{m_j\}_{j=1}^{k}$
6: **end if**
7: $MPI\_Bcast(\{m_j\}_{j=1}^{k}, root)$
8: **repeat**
9: $OldMSE := MSE$
10: $MSE^* := 0$
11: **for** $j := 1$ **to** k **do**
12: $m_j{}^* := 0$
13: $n_j{}^* := 0$
14: **end for**
15: **for** $i := id * (n/P) + 1$ **to** $(id+1) * (n/P)$ **do**
16: **for** $j := 1$ **to** k **do**
17: compute squared Euclidean distance $d^2(X_i, m_j)$
18: **end for**
19: find closest centroid m_l **to** item X_i
20: $m_l{}^* := m_l{}^* + X_i$
21: $n_l{}^* := n_l{}^* + 1$
22: $MSE^* := MSE^* + d^2(X_i, m_l)$
23: **end for**
24: **for** $j := 1$ **to** k **do**
25: $MPI\_Allreduce(n_j{}^*, n_j, MPI\_SUM)$
26: $MPI\_Allreduce(m_j{}^*, m_j, MPI\_SUM)$
27: $n_j := max(n_j, 1)$
28: $m_j := m_j / n_j$
29: **end for**
30: $MPI\_Allreduce(MSE^*, MSE, MPI\_SUM)$
31: **until** $MSE < OldMSE$

An MPI implementation based on Algorithm 1 must consider the following key aspects: *splitting* the input data evenly amongst the working processes (the *for* loop in line 15 of Algorithm 1) and efficiently performing the *reduction* operations needed to determine the *new centroids* and *MSE* (lines 24 through 30 of Algorithm 1).

In most real usage scenarios, the input data for *PKMeans* is initially accessible only to the *root* process (usually this

would be the process having *id = 0*). This implies that the input data should be split using communications between the *root* process and the other working processes. *Scatter* is the standard collective operation that performs such a split. This operation implies that the *root* process sends a unique message of size *rsize* to every other node [14]. Considering the address of the input buffer *sendbuf*, the target processing node given by *tid* and the size of the data type *datasize*, the *unique message* for each processing node is determined according to [11], [10]:

$$message_{tid} := sendbuf + tid \cdot rsize \cdot datasize. \quad (4)$$

The MPI standard implements the collective *scatter* operation through *MPI_Scatter* or *MPI_Scatterv* [10], [11]. Using these functions in order to split the data stored by the *root* process has two major drawbacks. First of all, these functions do not always ensure a *balanced* distribution of data. A standard *MPI_Scatter* call cannot be performed if the *length* of the input data (variable *n* in Algorithm 1) is not divisible by *the total number of processes* (variable *P* in Algorithm 1) [11], [10]. *MPI_Scatterv* might solve this issues, but it requires that a computational stage takes place prior to the actual communication in order to correctly determine the number of elements that need to be sent to any given process [11], [10]. The second important drawback is that *MPI_Scatter* or *MPI_Scatterv* involve *P-1* send operations for the *root* rank in order to divide the data between processes [10], [11]. Although data splitting is considered a pre-processing stage, it may lead to significant delays for the *root* rank.

The reduction stage at the end of Algorithm 1 (lines 24 through 30) must also be handled carefully. Although the MPI documentation does not provide any specific details on how *MPI_Reduce* and *MPI_Allreduce* functions are implemented, [15] states that these functions could reach *O(P)* time in the worst case and is dependent to the implementation of the MPI standard. The overall time complexity for computing new centroids (including communications) is given by:

$$O(k \cdot P). \quad (5)$$

Considering that the *reduction steps* need to be applied until no more objects are moved between the clusters, if we consider t loops before convergence is reached, then (5) becomes:

$$O(t \cdot k \cdot P). \quad (6)$$

The results presented in (3) and (6) show us that increasing the number of processes reduces the distance computing stages, but could also linearly increase the time required to determine the new centroids. This would result in an overall *inefficient* MPI implementation of *PKMeans* since the actual time reduction would be far less than the estimate given in (3).

III. Our Approach

Considering the disadvantages presented in *Section II*, an *efficient* implementation of *PKMeans* should focus on reducing the time spent in *data partitioning* and *reduction* phases. This may be achieved through the use of a communication cost effective topology. In our approach we have centered our efforts on the *hypercube* topology [14].

A. The Hypercube Topology

Given P processing nodes (*P must be a natural power of base 2*), the *hypercube topology* (*HC*) has $d = \log_2 P$ dimensions [14]. Each processing node (having the binary representation of its own *id* given in (7a)) has exactly d neighbors, each neighbor being distributed on one of the d dimensions of the cube (the binary codes for these neighbors is given by (7b, 7c)).

$$id = i_{d-1}...i_{b+1}i_b i_{b-1}...i_0 \quad (7a)$$

$$id^b = i_{d-1}...i_{b+1}\tilde{i_b}i_{b-1}...i_0 \quad (7b)$$

$$where\ \tilde{i_b} = 1 - i_b, 0 \leq b < d \quad (7c)$$

It can easily be observed that each d-dimensional cube is obtained by connecting the corresponding nodes of *two (d-1)*-dimensional cubes [14] (we will refer to these two cubes as *corresponding sub-cubes*). This property is called the *recursion property of a HC*. An important advantage of this property is that the maximum distance between any two processing nodes is at most d. This implies that a message transfer between any two processing nodes within a *HC* requires a maximum of d steps. The immediate consequence is that, if properly handled, all collective communications (including *scatter* and *reduction*) take *exactly d steps* to complete [14].

B. Initial Data Partitioning

In *Section II* we have shown that one key aspect for every MPI based implementation of *PKMeans* is the initial distribution of data amongst the processing nodes. The common approach is to use either *MPI_Scatter*, or *MPI_Scatterv* and balance the distribution if necessary. We have also shown that, according to various MPI implementations, these functions require *(P-1)* steps in performing this distribution.

In order to overcome this disadvantage, we resorted to modifying the basic *one-to-all* broadcast procedure over *HC* ([14], chapter 4). The reason behind choosing this algorithm is that the *one-to-all* broadcast procedure over a *HC* requires d steps to complete [14].

The communication pattern depicted in the *one-to-all* broadcast may be used in order to implement the basic *scatter* procedure over *HC* [14]. In order to achieve *scatter* like functionality using the basic *one-to-all* broadcast, the *message* to be "broadcasted" must contain both the data for the target process id (the *new root for the corresponding sub-cube*), as well as all the data for all the processes that belong to that same sub-cube. Therefore, in each stage during which a process acts as a *sending* process it must send the upper

half of its message. The *receiving* process on the other hand must determine exactly how many elements it will receive. This determination has to be performed without any supplementary communications and, also, without resorting to some complex computing stages. Considering the *HC* topology and the broadcast algorithm presented in [14], one can easily determine the *receiving* stage for each participating process. Using the *stage id* (variable *i*), the *length* of the original message and the dimension *d* of the *HC*, we have determined that each node in a *HC* receives *at most* the number of elements given by:

$$recvlen = \frac{length}{2^{d-i}} + 1. \quad (8)$$

Using (8), we have developed the procedure presented in Algorithm 2 in order to distribute an array over *d-dimensional HC* (a similar approach is also presented in [14], page 168). The difference from the classic approach in [14] is that Algorithm 2 requires no initial computation in order to determine how many elements will be received by each process.

Algorithm 2 Array distribution over a d-dimensional HC

procedure *CUBESplit (array, d, id)*

```
1:  mask := 2^d − 1
2:  for i := d − 1 down to 0 do
3:    mask := mask XOR 2^i
4:    if 0 = (id AND mask) then
5:      partner_id := id XOR 2^i
6:      if 0 = (id AND 2^i) then
7:        sendLen := length/2
8:        length := length − sendLen
9:        send message at array + length having sendLen
          elements to partner_id
10:     else
11:       length := length/2^(d−i) + 1
12:       receive message of max length from partner_id
13:       adjust length to map the exact number of re-
          ceived elements
14:     end if
15:   end if
16: end for
```

In Algorithm 2 the variable *length* denotes the *length of the original input array for all processes*. After each initial *receive stage*, each process uses variable *length* to store the actual number of elements it needs to split in further stages. When the procedure is finished, each process uses this variable to store the actual length of its corresponding partition from the original array.

Lemma 1: Algorithm 2 achieves a difference of at most *one* element between any two processes.

Proof: Let n be the length of the input array. Let d be the dimension of the *HC* having $P = 2^d$ processors. Let HC_{id} denote the *HC* having root *id*. In the first division stage of Algorithm 2, processor having $id = 0$ sends $[\frac{n}{2}]$ elements to processor $id_p = 0$ XOR 2^{d-1} and retains $n - [\frac{n}{2}]$ elements. If n is even, then HC_0 and HC_{id_p} have exactly $\frac{n}{2}$ elements. If n is odd, then HC_0 retains $[\frac{n}{2}] + 1$ elements and HC_{id_p} obtains $[\frac{n}{2}]$. Considering the fact that *HC* is a *recursive topology* and that Algorithm 2 behaves in a *recursive manner*, the relations given above are maintained through out all division stages. It results that the maximum difference between the number of elements between any two processors is at most one. ■

C. Reduce Operations

We have shown in *Section II* that according to [15] the total number of steps consumed by the reduction operation at the end of each stage in Algorithm 1 is linear with respect to the total number of processes P (see *Section II* - (6)). As suggested in [15], this may be improved. In [14] it is proven that all collective communications performed over a *HC* require exactly d steps to complete (d represents the dimension of the *HC*).

Taking all these observations into consideration, we resorted to the basic *all-to-all broadcast* presented in [14]. Basically, the single modification we performed to the original algorithm given in [14] is that the *message reunion* in the original algorithm has been adapted to map the calculation of the new centroids. This behavior is identical to the *all-reduce* communication primitive and various modifications are given in [14]. Algorithm 3 presents these modifications. Please note that *SUM* in Algorithm 3 denotes either the operation needed to compute the *coordinates* for the new centroids, either a simple *addition* operation that needs to be performed when computing the *MSE* value (see Algorithm 1).

Algorithm 3 Reduce operation over all-to-all broadcast on a d-dimensional HC

procedure *CUBEReduce (input, output, d, id)*

```
1: output := input
2: for id := 0 to d − 1 do
3:   partner_id := id XOR 2^i
4:   send output to partner
5:   receive input from partner
6:   output := SUM(output, input)
7: end for
```

It can be easily observed that the procedure in Algorithm 3 requires exactly d steps to complete (where $d = log_2 P$, P being the total number of processes). Also, it can be easily observed that Algorithm 3 performs the required *all-reduce* operation as needed by Algorithm 1.

The major advantage in using this approach for reduction operations is that it limits the over all time consumed by these steps. We have shown that in a classic approach, the time needed for a single reduction stage is given by (5). This is due to the fact that a single *all-reduce* operation consumes *O(P)* time (according to [15]). In our approach however, a single *all-reduce* operation consumes $O(\log_2 P)$ time to complete. This implies that the time complexity of a single reduction stage in Algorithm 1 is given by:

$$O(k \cdot \log_2 P). \quad (9)$$

If Algorithm 1 reaches convergence after t stages, using (9) we can determine that the overall time complexity for all the reduction stages would in our case be given by:

$$O(k \cdot t \cdot \log_2 P). \quad (10)$$

Comparing (6) with (10), it can easily be observed that our approach performs an efficient reduction stage with respect to increasing the total number of participating processes.

Considering Algorithm 1 and the modifications presented in *Subsections B* and *C*, Algorithm 4 presents our approach to *PKMeans*.

Algorithm 4 Hypercube-PKMeans

1: $P := MPI\_Comm\_size()$
2: $id := MPI\_Comm\_rank()$
3: $MSE := largeValue$
4: $length := \textbf{\textit{CUBESplit}}(X, \log_2 P, id)$
5: **if** $id = root$ **then**
6: make initial selection for centroids $\{m_j\}_{j=1}^{k}$
7: **end if**
8: $MPI\_Bcast(\{m_j\}_{j=1}^{k}, root)$
9: **repeat**
10: $OldMSE := MSE$
11: $MSE^* := 0$
12: **for** $j := 1$ **to** k **do**
13: ${m_j}^* := 0$
14: ${n_j}^* := 0$
15: **end for**
16: **for** $i := 1$ **to** $length$ **do**
17: **for** $j := 1$ **to** k **do**
18: compute squared Euclidean distance $d^2(X_i, m_j)$
19: **end for**
20: find closest centroid m_l **to** item X_i
21: ${m_l}^* := {m_l}^* + X_i$
22: ${n_l}^* := {n_l}^* + 1$
23: $MSE^* := MSE^* + d^2(X_i, m_l)$
24: **end for**
25: **for** $j := 1$ **to** k **do**
26: $\textbf{\textit{CUBEReduce}}({n_j}^*, n_j, \log_2 P, id)$
27: $\textbf{\textit{CUBEReduce}}({m_j}^*, m_j, \log_2 P, id)$
28: $n_j := max(n_j, 1)$
29: $m_j := m_j / n_j$
30: **end for**
31: $\textbf{\textit{CUBEReduce}}(MSE^*, MSE, \log_2 P, id)$
32: **until** $MSE < OldMSE$

IV. Experimental Results

In order to test Algorithm 4 against Algorithm 1, we have implemented both algorithms using C++ and LAM/MPI v. 7.1.2. For Algorithm 1, initial data distribution has been performed only using *MPI_Scatterv*. Tests have been performed both on a computer cluster with 12 workstations (Dell Optiplex 755, Intel 2.66 GHz, 64 bit, 2 GB RAM) and on an 8-core server (IBM 3800, 4 x XEON HT 3.66 GHz, 64 bit, 8 GB RAM). Due to these limitations, both Algorithm 1 and Algorithm 4 were run on 2, 4 and 8 processes. Test data-sets were synthetic and consisted of 5000, 7500 and 10000 points having 4 attributes (*coordinates on axes X, Y, Z, T*). For each of the three sets, we considered $k = 2000$ target clusters. Two types of tests have been conducted for each data-set.

TABLE I
Initial data partitioning

Data set length: 5000				
	Classic split		Cube split	
No. proc.	Items/proc	Avg. time	Items/proc	Avg. time
2	2500	4.90	2500	4.50
4	1250	4.50	1250	3.50
8	625	9.33	625	8.16
Data set length: 7500				
	Classic split		Cube split	
No. proc.	Items/proc	Avg. time	Items/proc	Avg. time
2	3750	7.00	3750	5.10
4	1875	7.20	1875	7.00
8	938: 0-3 937: 4-7	6.33	938: 0,2,4,6 937: 1,3,5,7	10.25
Data set length: 10000				
	Classic split		Cube split	
No. proc.	Items/proc	Avg. time	Items/proc	Avg. time
2	5000	8.90	5000	7.20
4	2500	8.67	2500	9.35
8	1250	11.18	1250	11.00

a) Initial data partitioning: The first type was performed to test our approach for the initial data partitioning against the classic use of *MPI_Scatterv*. Table I presents the results we have obtained (time results are given in milliseconds).

Although the results obtained for data splitting using the *HC* approach are not as expected, in most cases we obtained smaller response times than the classic *MPI_Scatterv* approach. These results gain an even greater significance if we consider the fact that we have used basic *Send/Receive* primitives whereas *MPI_Scatterv* usually achieves communications through direct interaction with *lamd* (the LAM/MPI control daemon thread [10]).

b) Comparative time results: The second type of tests were designed to compare the overall time performance of our approach (Algorithm 4) against the classic approach (Algorithm 1). The first set of tests has been performed on the computer cluster and has failed for both algorithms. This was an expected result since the test data-sets are quite small and in such cases the latencies induced by network communications are considerable. The second set of tests, performed on the 8-core server, provided an interesting set of results (Table II - time results are given in seconds).

Data has been divided evenly amongst all processes for both algorithms (see Table I) and local computation has not been altered (see Algorithm 1 and Algorithm 4). Despite the fact that *MPI_Allreduce* is implemented similarly to *MPI_Scatterv* (direct interaction with *lamd*) and the fact that

TABLE II

COMPARATIVE TIME RESULTS

Data length: 5000 Iteration count: 9				
Sequential	Generic PKMeans		Hypercube PKMeans	
	No. proc.	Avg. time	No. proc	Avg. time
20.8068	2	10.4555	2	10.3877
	4	5.3647	4	5.3015
	8	4.9862	8	4.8803
Data length: 7500 Iteration count: 11				
Sequential	Generic PKMeans		Hypercube PKMeans	
	No. proc.	Avg. time	No. proc	Avg. time
38.0796	2	19.2438	2	19.0708
	4	9.7368	4	9.7174
	8	9.1801	8	8.8869
Data length: 10000 Iteration count: 11				
Sequential	Generic PKMeans		Hypercube PKMeans	
	No. proc.	Avg. time	No. proc	Avg. time
50.6392	2	25.6233	2	25.4109
	4	12.9323	4	12.8905
	8	11.9807	8	11.8243

we have again used the basic *Send/Receive* primitives in order to implement the *CUBEReduce* presented in Algorithm 3, our approach outperformed the classic MPI implementation of *PKMeans* in every test.

V. CONCLUSION AND FUTURE WORK

In this paper we have presented a new approach for implementing the classic *PKMeans* algorithm. This approach makes use of the *HC* topology in order to efficiently perform the required collective operations. The results presented in *Section IV* have shown that, using en efficient communication topology like the *HC*, we can further improve the time responses of the given parallel model. The significance of these results is even greater if we consider the fact that in order to implement Algorithms 2 and 3 we have resorted to the basic *Send/Receive* primitives. Even though these two primitives involve a higher connection set-up time, restraining the communication time to a logarithmic dependency on the total number of processes proved to be more effective.

The next immediate development we plan is to implement the above mentioned algorithms using lower level functions provided by the LAM/MPI environment rather than *Send/Receive*. This should provide more accurate information on the actual time reduction gained through using a more efficient communication topology and on the actual scalability improvement.

We would also consider extending these tests to larger volumes of real rather than synthetic data. This would provide a more accurate insight on how well the *hypercube reduction* performs given that a more complex metric function is used.

ACKNOWLEDGMENT

The first author of this paper would like to thank *Prof. Mitica Craus* for his continued support and valuable insights on analyzing the parallel model for the K-Means Clustering algorithm.

REFERENCES

[1] P. Berkhin. (2002) Survey of clustering data mining techniques. [Online]. Available: http://www.ee.ucr.edu/˜barth/EE242/clustering_survey.pdf

[2] M. K. Jiawei Han, *Data Mining - Concepts and Techniques*. Morgan Kaufman Publications, 2006.

[3] Y. Li and S. M. Chung, "Text document clustering based on frequent word sequences," in *Proc. (ACM) CIKM05*, Bremen, Germany, Oct. 2005, pp. 293 – 294.

[4] O. F. David A. Grossman, *Information Retrieval - Algorithms and Heuristics (Second Edition)*. Springer, 2004.

[5] I. Dhillon and D. Modha, "A data-clustering algorithm on distributed memory multiprocessors," in *In Large-Scale Parallel Data Mining, Lecture Notes in Artificial Intelligence*, 2000, pp. 245 – 260.

[6] M. N. Joshi, "Parallel k - means algorithm on distributed memory multiprocessors," Computer Science Department, University of Minnesota, Twin Cities, Tech. Rep., 2003.

[7] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, L. M. L. Cam and J. Neyman, Eds., vol. 1. University of California Press, 1967, pp. 281 – 297.

[8] K. Stoffle and A. Belkonience, "Parallel k-means clustering for large datasets," *Proceedings of EuroPar*, 1999.

[9] [Online]. Available: http://openmp.org/wp/openmp-specifications/

[10] [Online]. Available: http://www.lam-mpi.org/using/docs/

[11] [Online]. Available: http://www.open-mpi.org/doc/

[12] S. Wang and R. Zhang, "Introduction to clustering," Dalhousie University, Faculty of Computer Science. [Online]. Available: http://flame.cs.dal.ca/˜swang/p1.ppt

[13] [Online]. Available: http://software.intel.com/en-us/intel-mpi-library/

[14] A. Grama, A. Gupta, G. Karypis, and V. Kumar, *Introduction to Parallel Computing*, 2nd ed. One Jacob Way, Reading, MA 01867-3999: Addison-Wesley Publishing; 2nd edition, jan 2003.

[15] "Rdma method for mpi_reduce/mpi_allreduce on large vectors patent description," June 2008. [Online]. Available: http://www.freshpatents.com/Rdma-method-for-mpi-reduce -mpi-allreduce-on-large-vectors-dt20080619ptan20080148013.php

A Case Study on Improving the Performance of Text Classifiers

Mircea Ionut Astratiei and Alexandru Archip

***Abstract*— This paper aims to improve the k-means clustering and k-nn classification algorithms results in order to aid the human expert in choosing the number of clusters and their initial centers for k-means algorithm and the variable *K* for the k-nn algorithm. We present a set of comparative results between classifications that are performed using only the human expert as trainer and an automatic approach that uses clustering results as training sets for the classification of text documents.**

I. Introduction

*Data mining*is the process of extracting patterns from data and establishing relationships. Data mining is used in many areas like mathematics, cybernetics, genetics, marketing, web search engines, etc. The classification of web search results represents only one of the newest applications for data mining: web documents are clustered/classified based on their content and relevant search results with respect to a given set of keywords are better presented to the client. Considering the huge amount of data gathered from the Internet is all-important to aid the human expert in applying the data mining algorithms [1]. The present paper introduces an automated method to classify the collected documents based on text analysis and similarity determination using the Cosine Similarity Measurement. Finally the paper compares the results of generic and improved k-means and k-nn algorithms and highlights the fact that a human expert cannot face the huge amount of data, unlike the computer which might determine the similarity more accurately using formulas.

II. Clustering and Classification Generic Data Mining Algorithms

A. Clustering

Clustering, represents the process of grouping the data into classes called clusters based on some attribute values that are characteristic for all the objects. The elements in one cluster must have similar attributes values and must be different from the elements of the other classes. This similarity is often calculated using distances between the attributes that define the objects.

Clustering methods can be organized in some categories: *partitioning methods, hierarchical methods, density-based methods, grid-based methods, model-based methods, methods for high-dimensional data (such as frequent patternbased methods), and constraint-based clustering* according to [2], [3]. Partitioning methods classify the given data in a known number of groups that represent the clusters. A cluster must satisfy two conditions: it has to contain at least one object and each object must belong to one cluster. Partitioning methods involve determining all clusters during the first iteration and then improving the results by moving objects between clusters at each new iteration. Final results are given when no other efficient change can be done. Through out each iteration, the clusters are defined by a dominant object or by a new object obtained from processing the values of the characteristic attributes.

Hierarchical methods can be either *agglomerative* or *divisive*, also called the *bottom-up* and *top-down* strategies, and consist in building a hierarchy of clusters. Data is not partitioned into a cluster in a single step, but instead a series of partitions takes places. The agglomerative methods proceed by merging each object into groups. On the other hand the divisive approach starts with groups, or a number of groups, that will be successively separated into a larger number of classes.

Density-based methods focus on the notion of compactness and the main idea is to start from a cluster already created. Each new object is added to the cluster that contains the closest neighborhood. One of the partitioning methods used in statistics and machine learning is the kmeans algorithm which aims to create a group of K clusters from an initial set of n objects, each object being assigned to the cluster with the closest mean. Centroid-based technique for k-means use a number n of objects and a constant K that represent the number of the classes in which the human expert desires to split the data [2], [3].

1) Brief description of how the k-means algorithm works: For the given K clusters, a number of K objects are needed in order to represent the centers of the respective clusters. These initial centroids are, in most cases, chosen randomly from the lot of size n. The remaining n-K objects will be assigned to the most similar cluster with respect to the used metric function. Usually the Euclidian metric is employed to compute the distance between the center of the cluster and an object. After all the n objects are assigned, a new center for each cluster is determined. This new center may be the most representative object in the lot or it can be defined as a new object which is obtained by computing the mean values for the significant attributes of all members [2]–[4]. The algorithm reaches a state of convergence when no objects are moved between the clusters or when the centroids of the clusters no longer change their attribute values.

I. Astratiei is a graduate student of The Faculty of Automatic Control and Computer Engineering, The "Gheorghe Asachi" Technical University of Iasi ionut.astratiei@yahoo.com

A. Archip is with The Faculty of Automatic Control and Computer Engineering, The "Gheorghe Asachi" Technical University of Iasi alexandru.archip@cs.tuiasi.ro

B. CLASSIFICATION

Classification is a form of data analysis where a model is used to predict the category to which a new object belongs to. The main difference between clustering and classification is that in classification the number of classes and their labels are predetermined.

This process has two steps. The first step, also known as learning step, consists of analyzing a predetermined training set of data classes or concepts. Unlike the clustering process, this first stage consists of a supervised learning strategy. For the second step, the classifier estimates whether the new data belongs to one of the classes already known, based on the rules obtained at the first step. There are many classification algorithms: *Decision Tree, Bayesian Classification, Rule-Based Classification, Learning from Your Neighbors*.

Decision tree algorithms are based on a tree structure where internal nodes represent test conditions for various attributes. For each such a node, an edge represents the decision taken to reach another node. Finally the leaves indicate the classes determined based on a set of possible decisions [2], [3]. The same idea is used in rule-based classification algorithms where the learning model is represented by a set of IF THEN rules [2], [3]. Bayesian classification consists on statistical data and probabilities computed by Bayes Theorem [2]. Lazy Learners (or Learning from Your Neighbors methods) construct a general model from the training data set.

1) Brief description of how the k-nn algorithm works: A lazy learner algorithm is k-nn (k-nearest-neighbor), its efficiency being demonstrated on large amounts of data. The principle on the k-nn classification consists in comparing the new data with the training set and learning from this analogy. Given a new object, the algorithm searches within the training data set for the K most similar objects (similarity is again considered with respect to a given metric function). The result is given by the simple majority of the similar objects.

K represents the number of training data objects that will be used to compare the new object. K must be a positive non null integer number. The value of K is important and this choice influences the predictions. For $K=1$ the unclassified object is assigned to the class with the most similar object in the training data [2]–[4]

III. COMPUTING TEXT DOCUMENTS SIMILARITY

A. PREPARING THE DOCUMENTS

Considering the fact that Cosine Similarity use term frequency to create the vectors, we need to prepare the text in order to ensure a correct determination of the distance between documents. Normalization of the documents consists in scanning the text and eliminating irrelevant characters like (, : ;)(} { @ # $ % etc) and words like prepositions, pronouns, some adverbs or verbs etc.

B. COSINE SIMILARITY

We compute the distance between documents using the Cosine Similarity Measurement. Cosine Similarity is a measure of similarity between two vectors of$n$dimensions by finding the cosine of the angle between them. For text mining the vectors are the term frequency vectors of the documents [5]. The Cosine Similarity of two vectors v_1 and v_2 is defined as:

$$\cos(v_1, v_2) = \frac{dot(v_1, v_2)}{||v_1|| \cdot ||v_2||}, \tag{1}$$

having:

$$dot(v_1, v_2) = \sum_{i=0}^{n-1} (v_1[i] v_2[i]), \tag{2}$$

$$||v|| = \sqrt{\sum_{i=0}^{n-1} v[i]^2} \tag{3}$$

$n =$ dimension of vectors

The resulting similarity ranges from 0 to 1 where 0 means the vectors are completely different (the documents are dissimilar) and 1 means the vectors are equal (the documents are similar).

Example:

S_1: Data mining is a new technology.

S_2: Data mining is a new process of extracting data patterns.

data, mining, is, a, new, technology, process, of, extracting, patterns

v_1=[1, 1, 1, 1, 1, 1, 0, 0, 0, 0]

v_2= [2, 1, 1, 1, 1, 0, 1, 1, 1, 1]

dot (v_1, v_2)= 6;

$||v_1||$=2.44;

$||v_2||$=3.46;

cos(v_1,v_2)=0.70

IV. IMPROVEMENTS

A. K-MEANS

Our implementation for the k-means algorithm aims to help the human expert in choosing the number of clusters and their initial centroids. Furthermore, after the clusters are obtained, a re-clusterization is made for the documents that belong to none of the clusters already obtained. Similar work is done in [6], [7], but unlike [6] or [7] our approach determines, in a semiautomatic way the variable K and also the initial centroids for each one of the K clusters.

1) The number of clusters and initial centroids: We suppose that there is at least one cluster and we randomly choose the first center for the first cluster. Then the distance between the centroid and the remaining documents is computed and the most similar are eliminated. We consider similar those documents if the distance doesnt exceed a threshold. In our case this threshold is obtained by observing the distances between all the documents we have tested (threshold is estimated in our case 0.1 for the documents in Romanian and 0.13 for the documents in English). The threshold can vary if a more refined or compact classification is needed.

Once we have a centroid, we compute the average distance from the centroid and the documents that are most similar to it. The average is used to determine the next center for the next cluster. The farthermost distance, computed from the last centroid determined and the remaining documents, distance referred to the average already calculated, indicates the next centroids. These steps are repeated until there are no documents remaining. Finally we obtain a number of centroids that represents the input data for the k-means, data obtained automatically.

Having the number of clusters and the initial centroids we can apply the classic k-means. A number of iterations are performed until the clusters dont change classes anymore. After each iteration, a new center is computed. As a new centroid we choose the document that is closer to all the other ones in the same cluster.

The results are then filtered and if there is at least one document in a cluster that is farther from the final centroid that the threshold imputed, there will be a re-clusterization that applies the same steps for the lot of rejected files. After a re-clusterization at least one new cluster must appear.

This condition reduces the probability of including documents in clusters that contain dissimilar documents. The improvement of the algorithm should aid the human expert in order to better determine the fields of interest [8]–[11].

Algorithm 1 Initial centroids

```
1: n := number of files
2: k := 0
3: repeat
4:    for j := 0 to n do
5:       if file[j] =not removed then
6:          dist[j] := Distance(current centroid, file[j])
7:       end if
8:    end for
9:    for j := 0 to n do
10:      if file[j] = not removed AND dist[j] > threshold
         then
11:         remove(file[j])
12:         M :={file[j] | dist[j] > threshold}
13:      else
14:         Determine number of not removed files
15:      end if
16:   end for
17:   Determine the next center as the most distant file from
      M
18:   k++;
19: until (number of remaining file = 0)
```

B. K-NN

The classic k-nn algorithm is independent of other data mining methods and needs a set of input data to learn from. Input data for k-nn algorithm are the variable K and the training set which represents the learning model. K is generally determined experimentally starting with $K=1$ and observing the error rate. This determination is also a human influence; therefore, for an automatic classification algorithm we choose the K value depending on the number of files in the clusters. For a minimum error rate we consider K to be double of the minimum length of the clusters.

If we associate these two data mining methods, clustering and classification, we will obtain an automatic training data set for k-nn. This set represents the results of the k-means clusterization. Therefore the classes obtained will represent the training model for a classifier, the new documents brought being associated with the clusters to which they are most similar. After testing over 90 files in Romanian and 90 files in English using this association of algorithms and the automatic detection of input data for k-means, we managed to improve the precision.

V. RESULTS

The test application has been developed using C ANSI language. Tests were performed on a total of 180 text documents, 90 for Romanian language and 90 for English. Tests for k-nn classification were performed on 52 new text documents for English and 46 new text documents for Romanian. The documents were chosen randomly from free essays web links.

For testing the Romanian documents we chose 12 fields as history, physics, animals, gothic art, heart diseases, psychology, computer science, chemistry, philosophy, economy, astrology and geography. The English documents belonged to the following fields: advertising, AIDS, London, Cold War, cloning, Shakespeare, music, pollution, racism, operating systems, Greek mythology, internet, drugs in sports. In TABLES I, II, III and IV the columns marked with "*" denote the choice made from the human expert perspective.

A. K-MEANS TESTS

For k-means we performed 6 tests: one for our automatic implementation and 5 for the classic method with randomly chosen centers and centers chosen from the human expert perspective. We observed for the total number of classes resulted, the incorrect number of clusters and finally we calculated the percentage of correct clusters reported to the total number.

1) Results interpretation: For the Romanian documents there is a significant difference between the automatic approach, that in our case had correctly classified all the files, and the classic k-means approach where the percentage of correct clusters, in randomly chosen centroids case, is around $50 \div 60\%$. This method gives incorrect results in tests, so it cant be cosidered a reliable solution. We have obtained better results when we have chosen the centroids, but this would imply the existence of a good expert in a real world scenario. During testing we have discovered some cases where the expert tends to extend the meaning of the document to a larger class because of the ideas exposed in the file.

For example, among the test files we had thirteen physics documents, that a human expert tends to classify just as one field (Physics), but using the Cosine Similarity and computing the distance between the files we have obtained

Fig. 1. Number of correct and incorrect clusters for English documents

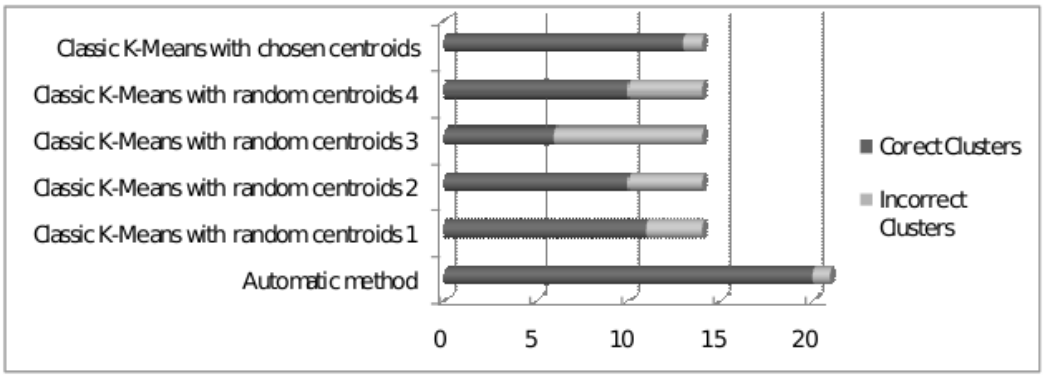


TABLE I

RESULTS OF K-MEANS CLUSTERIZATION FOR ROMANIAN DOCUMENTS

	Auto	**Classic**				
		Centroids				
		Random				**Chosen***
	No 1	No 1	No 2	No 3	No 4	No 5
Total number of clusters	20	15	15	15	15	15
Correct clusters	20	9	8	9	11	11
Incorrect clusters	0	6	7	6	4	4
Percentage of correct clusters	100%	60%	53,33%	60%	73,33%	73,33%

TABLE II

RESULTS OF K-MEANS CLUSTERIZATION FOR ENGLISH DOCUMENTS

	Auto	**Classic**				
		Centroids				
		Random				**Chosen***
	No 1	No 1	No 2	No 3	No 4	No 5
Total number of clusters	21	14	14	14	14	14
Correct clusters	20	11	10	6	10	13
Incorrect clusters	1	3	4	8	4	1
Percentage of correct clusters	95,23%	78,57%	71,42%	57,14%	71,42%	92,85%

three separate fields. Indeed after the automatic clusterization and after we read the files we observed that four of those documents described optics related phenomena, five of them were related to electrical engineering and the last four were mechanics related.

Another example involves the 6 colected documents concerning operating systems. After reading all the documents we have classified them as one cluster labeled Operating Systems and we have chosen one initial centroid. The automatic method had split the documents into two clusters. Indeed the documents contained parallels between Windows and Unix or Mac or they were describing one of the OS, apart from one that explained how to configure an OS. This file was clustered in a separate class, a good thing because even if the same field was reached in all six files, the main idea described in the last file was significantly different from the rest. For a better understanding of why a refined clusterization is needed, imagine the following case: you want to configure Windows and the search results lead you to documents comparing Windows an Linux, instead of offering the desired information.

A notable difference was discovered between the tests with English documents and Romanian documents. It seems that for English the results are approximately $10 \div 15$ percent more accurately in k-means classic methods. On the other hand there has been little research done in the clusterization and classification for the documents in Romanian language which is a bit more difficult to work with because of the multiple derived words and the diacritics.

Test results for k-means are presented in Fig. 1 (English documents) and detailed in TABLE I and TABLE II (for both Romanian and English documents).

B. K-NN TESTS

For the k-nn we performed 4 tests using the automatic k-means results as training set, classic k-means (centroids chosen randomly and from the human expert perspective) results as training set and a training set chosen from the human expert perspective. We observed the total number of documents to classify the number of training set clusters, the number of incorrect classified documents and finally we calculated the percentage of correctly classified documents. We can not rely on the randomly chosen centroids k-means method because a classification using those classes

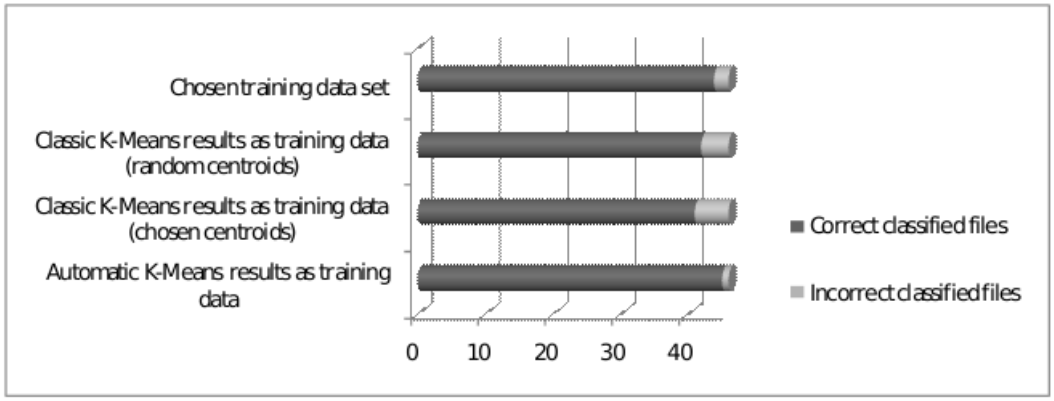


Fig. 2. Number of correct classified files for Romanian documents

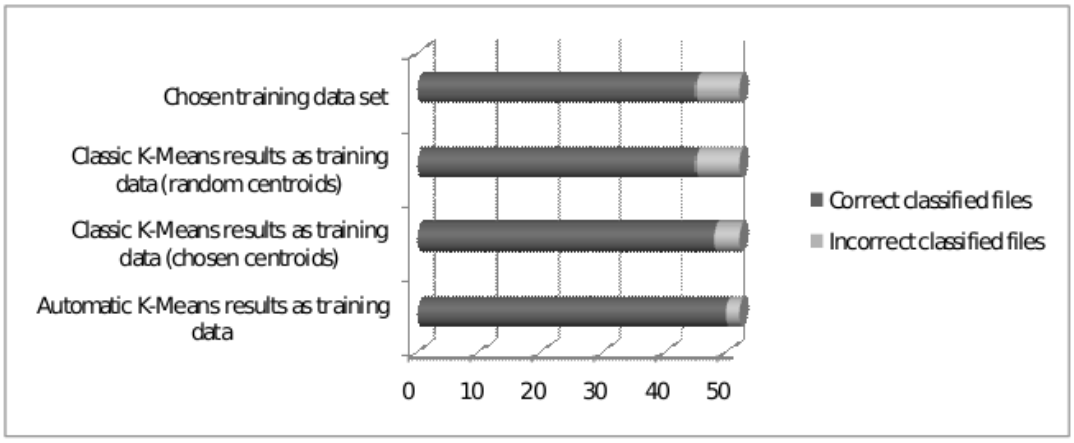


Fig. 3. Number of correct classified files for English documents

TABLE III
RESULTS OF K-NN CLASSIFICATION FOR ROMANIAN DOCUMENTS

	Automatic k-means results	**Classic k-means results**		**Chosen clusters***
		Chosen centroids*	**Random centroids***	
Number of training clusters	20	15	15	15
Number of documents	46	46	46	46
Correct classified documents	45	41	42	44
Incorrect classified documents	1%	5%	4%	2%
Percentage of correct classified documents	97,82%	89,13%	91,3%	95,65%

TABLE IV
RESULTS OF K-NN CLASSIFICATION FOR ENGLISH DOCUMENTS

	Automatic k-means results	**Classic k-means results**		**Chosen clusters***
		Chosen centroids*	**Random centroids***	
Number of training clusters	21	14	14	14
Number of documents	52	52	52	52
Correct classified documents	50	48	45	45
Incorrect classified documents	2%	4%	7%	7%
Percentage of correct classified documents	96,15%	92,30%	86,53%	86,53%

for training data will give ambiguous results. For example the randomly chosen method for English files shuffled the documents about Shakespeares life with the essays about Romeo and Juliet. When we tried to classify new files about Romeo and Juliet or Shakespeares life, these files have not been assigned to their corresponding classes properly.

Out tests revealed that for English files there is a 96% probability that a case like that will not appear and a 98% for the Romanian files if we apply the automatic method for k-means and we use the results as training data for k-nn. For

better results the new files used in testing the k-nn belonged to the same fields like the documents that we have been clustering with k-means. Test results for k-nn are presented in Fig. 2 and Fig. 3 and detailed in TABLE III and TABLE IV.

VI. CONCLUSION AND FUTURE WORK

In this paper we have presented a method of clustering and classification that aids the human expert in choosing the initial number of clusters and the initial centroids for k-means algorithm and training data set for the k-nn algorithm. We have focused on combining two different data mining techniques (a descriptive method as trainer k-means Clustering - and a predictive technique k-nn classifier) in order to improve the accuracy of a classifier.

In order to obtain more precise results we intent to make an analysis that will automatically detect derived and composed words. This analysis should take into consideration other descriptive data mining methods (such as frequent sequence miners) in order to further improve a potential classifier. One such improvement consists in determining dominant phrases for a set of computed clusters. Future development also involves addressing larger datasets.

REFERENCES

[1] O. F. David A. Grossman, *Information Retrieval - Algorithms and Heuristics (Second Edition)*. Springer, 2004.

[2] M. K. Jiawei Han, *Data Mining - Concepts and Techniques*. Morgan Kaufman Publications, 2006.

[3] S. V. Mark Hornick, Erik Marcade, *Java Data Mining Strategy, Standard and Practice*. Morgan Kaufman Publications, 2007.

[4] A. W. Moore, "K-means and hierarchical clustering," 2001. [Online]. Available: http://www.cs.cmu.edu/afs/cs/user/awm/web/tutorials/kmeans11.pdf

[5] E.Garcia, "An information retrieval tutorial on cosine similarity measures, dot products and term weight calculations," 2006. [Online]. Available: http://www.miislita.com/information-retrieval-tutorial/cosine-similarity-tutorial.html

[6] A. M. Dan Pelleg, "X-means: Extending k-means with efficient estimation of the number of clusters," in *Proceedings of the Seventeenth International Conference on Machine Learning*. San Francisco: Morgan Kaufmann, 2000, pp. 727–734.

[7] M. Muhr and M. Granitzer, "Automatic cluster number selection using a split and merge k-means approach," *Database and Expert Systems Applications, International Workshop on*, vol. 0, pp. 363–367, 2009.

[8] P. C. Stan Salvador, "Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms," *Dept. of Computer Sciences Florida Institute of Technology Melbourne*, 2003.

[9] B. Li, S. Yu, and Q. Lu, "An improved k-nearest neighbor algorithm for text categorization," *CoRR*, vol. cs.CL/0306099, 2003.

[10] I. Matveeva, "Document representation and multilevel measures of document similarity," in *Proceedings of the 2006 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*. Morristown, NJ, USA: Association for Computational Linguistics, 2006, pp. 235–238.

[11] D. Arthur and S. Vassilvitskii, "k-means++: the advantages of careful seeding," in *SODA '07: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2007, pp. 1027–1035.

Hybrid Electric Vehicles Control Strategies-A Comparative Study

Babici Cezar, Alexandru Onea Member, IEEE

***Abstract*—Energy consumption and exhaust emissions of hybrid electric vehicles (HEVs) strongly depend on the HEV topology, power ratios of the components and applied control strategy. In order to enhance the design of HEVs, this paper provides a study for evaluating energy conversion phenomena of conventional and HEV topologies. The analysis is based on the different tests results of, fuel efficiency and associated energy losses, and therefore provides insightful information for HEV optimization.**

I. Introduction

ELECTRIC vehicles, hybrid electric vehicles, and fuel cell vehicles have been typically proposed to replace conventional vehicles in the near future. At present, almost all vehicles rely on the combustion of hydrocarbon fuels to derive the energy necessary for their propulsion. This paper provides some experimental results in order to quickly analyze the efficiency of conventional and hybrid cars. (energy consumption, energy management strategy evaluation).

The main tool for calculations is ADVISOR (Advanced Vehicle Simulator). It works interactively with Matlab and Simulink programming environment and contains a database with important types of vehicles, engines and electrical thermal batteries, mechanical transmission, etc. There are two main types of calculations that ADVISOR uses: backward-facing and forward-facing.

In backward-facing calculations, no driver behavior is required. The user must input the driving pattern, a velocity profile, called the speed trace. The force required to accelerate the vehicle is calculated and translated into torque. This procedure is repeated at each stage from the vehicle/road interface through the transmission, drivetrain, etc., until the fuel use or energy use is calculated.

In forward-facing calculations, the user inputs the driver model, then the simulator generates throttle and brake commands that are changed into engine torque, which is passed to the transmission model and passed through the drivetrain until a tractive force is computed.

The majority of calculations are done in the backward mode, although in order to keep the components from exceeding their physical limitations, some forward calculations are necessary. Most data in an ADVISOR simulation is passed backward. This data in the Simulink block diagrams are characterized as requirements. Some data is transferred forward in the Simulink block diagrams. It is characterized as available [1].

A. Description of Vehicle Movement

The tractive effort, *Ft*, in the contact area between tires of the driven wheels and the road surface propels the vehicle forward. It is produced by the power plant torque and is transferred through transmission and final drive to the drive wheels. The resistance usually includes tire rolling resistance, aerodynamic drag, and uphill resistance. According to Newton's second law, vehicle acceleration can be written as:

$$\frac{dV}{dt} = a = \frac{F_t - F_{tr}}{\delta M_v} \tag{1}$$

where V is vehicle speed, F_t is the total tractive effort of the vehicle, F_{tr} is the total resistance, M_v is the total mass of the vehicle, and δ is the mass factor, which is an effect of rotating components in the power train. Equation indicates that speed and acceleration depend on tractive effort, resistance, and vehicle mass [1].

Resistance forces of the runway are:

$$F_{tr} = F_x + F_{air} + F_r \tag{2}$$

The grading resistance can be expressed as:

$$F_x = M_v g \sin(\alpha) \tag{3}$$

where M_v is mass of vehicle, g is gravitational acceleration and the slope α is expressed as a percentage, positive when climbing and negative when descending.

$$\alpha = arctg\frac{slope}{100} \tag{4}$$

Aerodynamic drag is a function of vehicle speed, vehicle frontal area A_f, shape of the vehicle, and air density ρ_{air}:

$$F_{air} = \mathrm{sgn}(V_d)\rho_{air} C_d A_f V_d^{\,2} \tag{5}$$

$$V_d = V - V_{wind} \tag{6}$$

where V_{wind} is the wind speed, A_f is equivalent front surface of the vehicle, and C_d is the aerodynamic drag coefficient. Usual values for C_d are in the range (0.2, 0.4).

The rolling resistance of tires on hard surfaces is

This work was financial supported by the FC Continental Automotive Romania.

Cezar C. Babici is with the Automatic Control and Applied Informatics Departament, "Gheorghe Asachi" Tehnical University of Iasi Faculty of Automatic Control and Computer Engineering, Str. Prof. dr. doc. Dimitrie Mangeron, nr. 27, 700050, Iasi, Romania (+40766434951 e-mail: cbabici@ac.tuiasi.ro.)

Al. Onea is with the Automatic Control and Applied Informatics Departament, "Gheorghe Asachi" Tehnical University of Iasi Faculty of Automatic Control and Computer Engineering, Str. Prof. dr. doc. Dimitrie Mangeron, nr. 27, 700050, Iasi, Romania (e-mail aonea@ac.tuiasi.ro).

primarily caused by hysteresis in the tire materials. This is due to the deflection of the carcass while the tire is rolling. The moment produced by the forward shift of the resultant ground reaction force is called the rolling resistant moment:

$$T_r = F_y \cdot a \tag{7}$$

$$F_r \cdot r_d = F_y \cdot a \tag{8}$$

$$F_r = F_y \frac{a}{r_d} = C_r \cdot F_y \tag{9}$$

$$F_r = (C_{r0} + C_{r1} \cdot v)\, F_y \tag{10}$$

C_r is the rolling resistant coefficient, a is the distance of displacement and r_d is the rolling radius. C_{r0} is usual between (0.004, 0.02) and $C_{r1} << C_{r0}$ [2].The tire slip model relates weight on the tire, longitudinal force, vehicle speed, and slip in an equation or set of tables. Because of the limitations:

$$slip = \frac{\omega_{wr} r_{wh}}{v_r} - 1 \tag{11}$$

$$\omega_{wr} = \frac{1 + slip}{r_w} v_{\lim.req} \tag{12}$$

$$T_{wr} = F_{\lim.r} r_w + T_{w.loss} + J_w \left(\frac{\Delta \omega_{wr}}{\Delta t} \right) \tag{13}$$

where ω_{wr} is the required speed at wheel, r_{wh} is the wheel radius, T_{wr} is the required torque input to the axle, $F_{\lim.r}$ is the necessary average tractive force, $T_{w.loss}$ is the torque required to overcome bearing losses and brake drag and J_w is the rotational inertia.

To predict the maximum tractive effort that the tire–ground contact can support, the normal loads on the front and rear axles have to be determined. By summing the moments of all the forces about point *R* (center of the tire–ground area), the normal load on the front axle F_{yf} can be determined as:

$$F_{yf} = \frac{L_b}{L} M_v g \cos\alpha - \\ - \frac{h_g}{L} (M_v g \sin\alpha + F_{air} + \delta \cdot M_v \frac{dv}{dt}) \tag{14}$$

Similarly, the normal load acting on the rear axle can be expressed as:

$$F_{yr} = \frac{L_a}{L} M_v g \cos\alpha + \\ + \frac{h_g}{L} (M_v g \sin\alpha + F_{aer} + \delta \cdot M_v \frac{dv}{dt}) \tag{15}$$

Normal force acting on each individual wheel is calculated by dividing the normal force produced by the number of wheels on each axle. Speed and traction effort are limited by traction characteristics of the tires.

$$F_{w.r} \le F_{w.\max} \;;\; v_{w.r} \le v_{w.\max} \;. \tag{16}$$

$$F_{w.r} = F_{t.r} - F_{brake.r} + F_{yloss} \tag{17}$$

An automotive power train consists of a power plant (engine or electric motor), a clutch in manual transmission or a torque converter in automatic transmission, a gearbox (transmission), final drive, differential, drive shaft, and driven wheels. The torque and rotating speed of the power plant output shaft are transmitted to the drive wheels through the clutch, gearbox, final drive, differential, and drive shaft. The required speed is limited to the motor's maximum speed. The required torque is limited to the difference between the motor's maximum torque at the limited speed and the torque required to overcome the rotor inertia. The limited torque and speed are then used to interpolate in the motor/controller's input power map. Finally, the interpolated input power is limited by the motor controller's maximum current limit. This behavior is described in the following equations:

$$P_{mot.in.r} = \min(P_{mot.in.map}, I_{con.\max}, V_{bus.prev}) \tag{18}$$

$$P_{mot.in.map} = f(T_{mot.\lim.r}, \omega_{mot.\lim.r}) \tag{19}$$

where f is the functional relationship described by the motor map, $P_{mot.in.r}$ is the required motor's input power, $T_{mot.\lim.r}$ is the limited motor torque required, $\omega_{mot.\lim.r}$ is the limited motor speed required, and, $P_{mot.in.map}$ is the input power required to power the motor at its maximum limited torque and speed.

$$\omega_{mot.\lim.r} = \min(\omega_{mot.r}, \omega_{mot.\max}) \tag{20}$$

$$T_{mot.lim.r} = min(f_1(\omega_{mot.lim.r}), T_{mot.r} + J\left(\frac{\Delta \omega_{mot.lim.r}}{\Delta t} \right)) \tag{21}$$

where f_1 is the functional relationship described by the motor torque envelope. The forward-facing part of the motor/controller model accepts as input the available input power, and produces as outputs the available rotor torque and speed.

To compute the torque that can be produced by the motor/controller given the available input power, the motor/controller efficiency computed during the backward facing calculations is used, modeled as $T_{mot.\lim.r} / P_{mot.\lim.r}$:

$$T_{mot.avail} = T_{mot.\lim.r} \left(\frac{P_{mot.avail}}{P_{mot.\lim.r}} \right) - J_{mot} \left(\frac{\Delta \omega_{mot.\lim.r}}{\Delta t} \right) \tag{22}$$

The engine defined by its moment of inertia is characterized by the developed drive torque T_{act_p}, internal friction torque, T_{frec-p}, and external load on clutch, T_{amb}:

$$J_p \frac{d\omega_p}{dt} = T_{act\_p} - T_{frec-p} - T_{amb} \tag{23}$$

If the clutch is fully engaged and is not alleged any internal friction, then:

$$T_{amb} = T_{cv} \quad (24)$$

$$\omega_p = \omega_{amb} \quad (25)$$

where T_{cv} is gearbox torque, and ω_{amb} is the clutch speed. The gearbox is described by the moment of inertia, J_{cv}, its viscous friction torque is characterized by coefficient of friction D_{cv}. If it is considered i_{cv} the gear ratio then the associated model is:

$$\omega_{amb} = \omega_{cv} i_{cv} \quad (26)$$

$$J_{cv} \frac{d\omega_{cv}}{dt} = T_{cv} i_{cv} - D_{cv} \omega_{cv} - T_{ap} \quad (27)$$

Although propshaft is an elastic element to simplify the model it can be equated with a stiff shaft:

$$\omega_{ap} = \omega_{cv} \quad (28)$$

$$T_{dif} = T_{ap} \quad (29)$$

where ω_{ap} is the propshaft speed, and T_{dif} is the differential torque and T_{ap} is the propshaft torque. In the same way as the gearbox, the differential is shaped by the moment of inertia, J_{dif}:

$$\omega_{ap} = \omega_{dif} i_{dif} \quad (30)$$

where i_{dif} is the differential ratio.

$$J_{dif} \frac{d\omega_{dif}}{dt} = T_{dif} i_{dif} - D_{dif} \omega_{dif} - T_{aa} \quad (31)$$

if the wheels speed is the same. , T_{aa} is in (31) the auxiliary loads torque, D_{dif} is the friction coefficient of the differential

The internal combustion engine has a non-linear model; the inputs are the rotation speed and the angle of the acceleration pedal; the output is the torque of the crank shaft. In the SIMULINK model, a lookup table 2D is simulating the torque characteristic. The output is integrated and divided by the inertia moment (J_p), and the obtained signal is used as a reaction for the 2-D lookup table.

B. Electric Vehicles

Dynamic performance of a vehicle is usually associated with acceleration time, maximum speed and gradeability. The design of the drive train of an electric vehicle means the proper sizing of the electric motor and to choose the proper transmission parameters. The design of these parameters depends mostly on mechanical characteristics (torque, power) of the electric traction motor. The most used in electric vehicles and hybrid electric vehicles are induction motors, permanent magnet synchronous and reluctance motors. In this paper it was used an induction motor model.

The motor/controller model includes the effects of losses in the motor and controller, rotor inertia, and the motor's torque speed-dependent torque capability. Power losses are handled as a 2-D lookup table indexed by rotor speed and output torque. The motor maximum torque is given using a lookup table indexed by rotor speed. Motor control block ensures that the maximum current is not exceeded. Available torque is computed from available power by assuming that the ratio of rotor torque to input (electric) power is the same for the actual/achievable situation as was computed for the request. The motor model is based on the mechanical characteristic. In the stady-state, the motor is described by the equations:

$$U_a = R_a I_a + k\psi_e \omega = R_a I_a + E$$

$$T_e = k\psi_e I_a \quad (32)$$

Maximum torque is obtained when a maximum current is given in the rotor circuit:

$$T_{emax} = k\psi_e I_{amax} \quad (33)$$

In this range of angular speeds, lower than the base angular speed ω_b, maintaining constant the magnetization flux, the EMF of the electric motor increases with the speed of rotation:

$$E = k\psi_e \omega \quad (34)$$

Mechanical power developed by motor has a linear variation dependent on the angular rotation speed:

$$P = T_{emax} \omega \quad (35)$$

$$P_{max} = T_{emax} \omega_b \quad (36)$$

Maximum intensity of the motor current is given by:

$$I_{amax} = \frac{U_{amax} - E_{max}}{R_a} = \frac{U_{amax} - k\psi_e \omega_b}{R_a} \quad (37)$$

To maintain the maximum current even when the angular speed is over ω_b, when the voltage value is limited by the power converter, it is imposed a constant value for EMF, independent of the speed variation. Maximum EMF, with (34) yields:

$$E_{max} = k\psi_e \omega_b = k \frac{\psi_e \omega_b}{\omega} \omega \quad (36)$$

This is the case when the flux is hyperbolic decreasing when the speed is increasing over ω_b. A consequence of the flux decrease over ω_b is the hyperbolic decrease of the torque:

$$T_e = k \frac{\psi_e \omega_b}{\omega} I_{a\max} \quad (37)$$

On the other hand, in this range of speed, the power of the motor is expressed as:

$$P = T_e \omega = k \frac{\psi_e \omega_b}{\omega} I_{a\max} \omega = k\psi_e \omega_b I_{a\max} = P_{\max} \quad (38)$$

which means the power is constant and at the maximum value.

The traction force can be evaluated with (2). For short

periods of time, acceleration can be considered constant:

$$\frac{dv}{dt} = \frac{v_{k+1} - v_k}{t_{k+1} - t_k} \tag{39}$$

where v_{k+1} is the vehicle speed at time k+1 and v_k is vehicle speed at time k. Electricity consumption is calculated by integrating the battery pack supplied power:

$$P_{b-output} = [\frac{M_v g(\sin\alpha + (C_{r0} + C_{r1} \cdot v)\cos\alpha}{\eta_{ME}\eta_t} + \frac{0.5\rho_{aer} C_D A_f v^2 + \delta M_v \frac{dv}{dt}}{\eta_{ME}\eta_t}]v \tag{40}$$

Eq. (39) and (40) allow to compute the tractive force and energy at any moment of a driving cycle. When regenerative braking is used the input power is:

$$P_{b-input} = \gamma[\frac{(M_v g(\sin\alpha + (C_{r0} + C_{r1} \cdot v)\cos\alpha)}{\eta_{ME}\eta_t} + \frac{0.5\rho_{aer} C_D A_f v^2 + \delta M_v \frac{dv}{dt}}{\eta_{ME}\eta_t}]v \tag{41}$$

where $\sin\alpha$ or $\frac{dv}{dt}$ or both terms have negative values and γ (subunit value), is called recuperative factor and indicate the percentage of available energy that an electric car can provide.[3] Net energy consumption from the battery pack is expressed as:

$$E_{iesire} = \int_{tractiune} P_{b-output} dt + \int_{franare} P_{b-input} dt \tag{43}$$

Electrochemical batteries are devices that convert electric energy in potential chemical energy during loading, and convert potential chemical energy in electric energy. Another important parameter of a battery is the state of charge (SOC) defined as the ratio between current capacity and total battery capacity. SOC changing during dt is given by:

$$S_{oc} = \frac{dQ}{Q(i)} = \frac{idt}{Q(i)} \tag{44}$$

where Q(i) is the battery capacity at discharge current i. In the charging mode the current has a negative value and in the discharging mode the current is positive. In this way the battery charge level can be expressed as:

$$S_{oc} = S_{oc0} - \int \frac{idt}{Q(i)} \tag{45}$$

where S_{oc0} is the initial value of charging level [4].

II. Architecture Of Hybrid Electrical Drive Trains

Basically, any vehicle power train is required to develop sufficient power to meet the demands of vehicle performance, carry sufficient energy on-board to support vehicle driving in the given range, demonstrate high efficiency, and emit few environmental pollutants. Broadly, a vehicle may have more than one energy source and energy converter (power source), such as a gasoline (or diesel) heat engine system, hydrogen–fuel cell–electric motor system, chemical battery–electric motor system, etc. A vehicle that has two or more energy sources and energy converters is called a hybrid vehicle. A hybrid vehicle with an electrical power train (energy source energy converters) is called HEV(hybrid electric vehicle). The architecture of a hybrid vehicle is loosely defined as the connection between the components that define the energy flow routes and control ports. HEVs are now classified into four kinds: series hybrid, parallel hybrid, series–parallel hybrid, and complex hybrid.

A series hybrid architecture id presented in Fig.1.

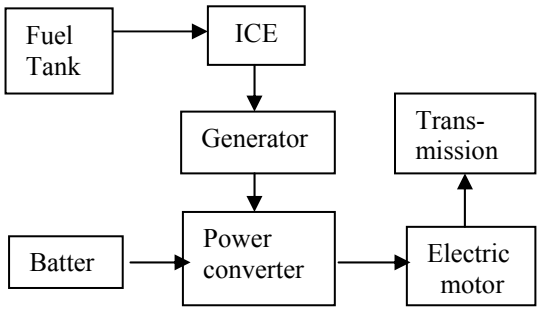


Fig.1 Series hybrid architecture

The main advantages are: engine is fully decoupled from the driven wheels so it can be operated at any point on its speed–torque characteristic map, and can potentially be operated within its maximum efficiency region, and they do not need multigear transmission. The main disadvantages are**:** the energy from the engine is converted twice, the inefficiencies of the generator and traction motor add up and the losses may be significant. The generator adds additional weight and cost.

A parallel hybrid drive train is a drive train in which the engine supplies its power mechanically to the wheels like in a conventional ICE-powered vehicle. The powers of the engine and electric motor are coupled together by mechanical coupling. The mechanical coupling of the engine and electric motor power leaves room for several different configurations.

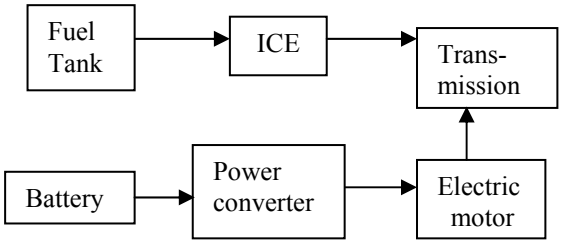


Fig. 2 Parallel hybrid architecture

The advantages of this architecture are: single energy conversion for both electrical and mechanical, considerable design flexibility exists in selecting the size of the M/G. The electric motor can be positioned after the gearbox, motor

shaft directly, making the energy losses through the gearbox to not exist.

A disadvantage of parallel arrangement is that the various added powertrain parts such as added clutches and transmissions. Typically a three-shaft transmission is needed; two input shafts and one output shaft.

III. A Comparative Study Of Hybrid Electrical Vehicles Performances

To minimize fuel consumption and emissions it is equally important to select an appropriate HEV topology as well as to develop the power flow control algorithm. There are few HEV control strategies (CS) mentioned in literature [5]. The simplest strategy is the 'thermostat' or 'on/off' developed initially for series hybrid. The battery state of charge is allowed to fluctuate between the maximum and minimum set points, rather like a thermostat that maintains the temperature within the desired range. The principle of this control strategy is to deplete the battery to a very low SOC and then trigger the internal combustion engine (ICE) to drive the generator to recharge the batteries while powering the electric motor (EM). Once the batteries are fully recharged, the ICE is shut off again until such needs arise again. During deceleration, some brake energy is recovered to help recharge the battery through regenerative braking. The aim of this control strategy is to propel the vehicle entirely under pure electrical mode as often as possible. This gives the advantage of setting the ICE to operate at one point of torque and speed that is most efficient and least polluting. It also prevents the ICE from handling transient loads where the highest level of emissions is usually produced. The EM that propels the vehicle under all driving conditions will handle the transient load. The ICE is set to run on only one fixed gear ratio that is either optimized for fuel economy or low emission when driving the generator. The lower limit on SOC is 0.52.

The Series Power Follower control strategy determines at what torque and speed the ICE should operate. Electrical power is generated, according to the given conditions of EM, batteries, ICE, and the power demanded by the vehicle. This CS is usually designed to maximize fuel economy, or minimize emission, or maximize battery life. The ICE may be turned off if the SOC gets too high, and turned on again if the power required reaches a certain threshold, or if the SOC hits the minimum level. This CS also incorporates regenerative braking to recycle some brake energy back into the batteries during vehicle deceleration. When the ICE is on, its power output tends to follow the power required, accounting for losses in the generator so that the generator output power converges with the power requirement. Therefore, in some instances, the ICE output power may be adjusted by the SOC, which tends to bring the SOC back to the middle of its operating range, or just keep the SOC above some minimum value. At other times, the ICE's output power might be kept near to the power of the ICE at maximum efficiency, or allowed to change no faster than a prescribed rate. Thus, this CS changes according to the preset SOC conditions and the power required for propelling the vehicle. In general, when the SOC is low and the power demand is less than the power of the ICE at its maximum efficiency, the generator is run at a power as close as possible to the ICE's most efficient operation point, without exceeding the system voltage power constraint. The batteries are charged as much as possible to keep the ICE's efficiency as high as possible while maintaining a mid-level SOC. When the power demand is less than the power at maximum efficiency of the ICE, but greater than the battery charge power, the generator is set to run at a power equivalent to the ICE's most efficient operation point. The required power is used to propel the vehicle while excess power generated is used to charge the batteries. During high-power demand, where the requested power is greater than the power at maximum efficiency of the ICE, the generator is set to run at a power equivalent to the ICE's most efficient operation point. Additional peak power is requested from the batteries to satisfy the total power requested. In city driving conditions, where the SOC is high, the ICE is shut off and the vehicle operates under pure electric mode as a zero emission vehicle. The advantage of this control strategy is that the battery packs are relatively small and the SOC is always maintained around a mid-level. In general this allows the overall weight of the propulsion system to be lighter. The disadvantage is that the ICE is forced to operate at multiple points in its efficiency and emission maps to adjust for load changes. This causes the emission to increase as the operation of ICE moves away from its maximum efficiency point. However, changing the throttle slowly may compensate for this negative effect.

The experimental results were obtained by numerical simulation using Matlab and ADVISOR tols. To analyze the performance differences of different types of vehicles it was made a test for every configuration.. Firstly two conventional powertrains were tested . In the first case it was chosen a conventiona powertrain with a 47 kW gasoline engine. The second case is a conventional powertrain with 94 kW gasoline engine. There were used two different gasoline engines with different powers because the first engine it will be compared witj an electrical vehicle with the same power. Performances of the 94 kW conventional engine are compared with an series and an parallel HEV, each of them using a 47 kW induction motor and a 47kW internal combustion engine.

1) The first case analysed is a conventional vehicle with a 47 kW power of internal combustion engine, and the weight 1001 kg.The acceleration time is given by:

$$t_a = \int_{v_1}^{v_2} \frac{\delta \cdot M_v}{F_t - 0.5\rho_{aer} C_D A_f v^2 - (C_{r0} + C_{r1} \cdot v) M_v g} dv \quad (46)$$

For testing fuel efficiency it was considered 5 cycles ECE (equivalent 5km urban traffic), and 5 cycles EUDC (equivalent 38.5 km highway traffic) .

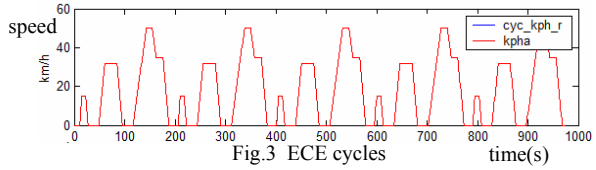


Fig.3 ECE cycles

The obtained results are:
-urban consumption 8.1l/100 km
-highway consumption. 5.3l/100 km

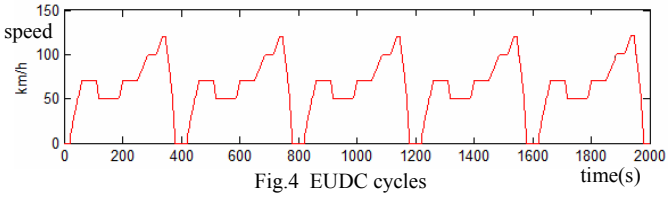


Fig.4 EUDC cycles

The dynamic performances are:
-acceleration time 0-100km/h: 16s;
-maximum acceleration-> 3.1m/ s^2 ;
-maximum speed: 155.6 km/h;

Overall system efficiency is computed as the ratio between the sum of energy used for traction and the energy used to overcome the aerodynamic force, and the difference between the fuel energy and the energy stored in batteries.

In this case the overall system efficiency over 5 ECE cycles is 0.05, and the overall system efficiency over 5 EUDC cycles is 0.174.

2) In the second case the tests were made over a 94 kW conventional vehicle with the weight 1135 kg.
Fuel consumption:
- urban 8.7l/100km
- highway 7.5 l/100km
Overall System Efficiency over ECE cycle is 0.036.
Overall System Efficiency over EUDC cycle is 0.565.
As dynamic performance :
-acceleration time 0-100km/h: 8.7s
-maximum acceleration: 5m/ s^2 ;
-maximum speed: 210.2 km/h;

3)The same tests were performed for an electrical vehicle with an 47 kW induction motor weight 1110 kg and MI-Pba batteries. The extra weight of the electrical vehicle compared with conventional vehicle is given by the energy storage system weight The result is given as the equivalent gasoline fuel consumption.
The equivalent urban consumption is 1.9 l/100 km and the remaining SOC at the end of the 5 ECE cycles is 90%. The urban consumption difference between the conventional car (ICE power 47 kw) and electrical car is a major advantage (6.2l/100 km).
The equivalent highway consumption is 2 l/100 km, and the remaining SOC at the end of the 5 EUDC cycles is 33%. It can be observed a gain of 37.7% (2.3l/100km) in highway consumption compared with first case.

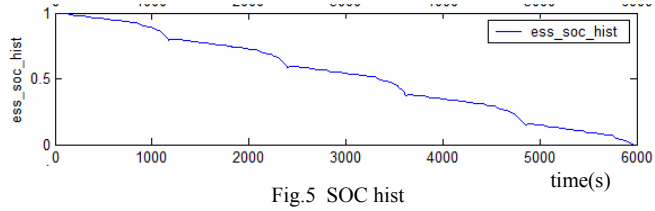


Fig.5 SOC hist

The dynamic performance results:
-acceleration time 0-100km/h 17.7s
-maximum acceleration 5m/ s^2 ;
-maximum speed 130.3 km/h;
Overall System Efficiency over 5 ECE cycle is 0.263.
Overall System Efficiency over 5 EUDC cycle is 0.565.

4) For a better look the same tests were made over another electrical vehicle with MS-LI batteries 47 kW induction motor, and weight 863 kg. The weight difference compared with the previously case is given by the batteries production technology. Equivalent fuel consumption for 5 ECE cycles is 1.6 l/100 km and the remaining SOC at the end of the test is 66%. The result shows a lower consumption than the previously electric car, but a lower SOC at the end of the drive cycles. This has a negative impact on the vehicle operating range. The worse is that the batteries didn't sustained the traction motor until the end of 5 EUDC cycles.

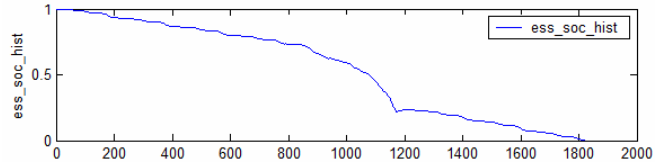


Fig. 6 SOC history

Overall System Efficiency over ECE cycles is 0.244.
The dynamic performances are:
- acceleration time 0-100km/h 14.1s;
- maximum acceleration 5m/ s^2 ;
- maximum speed 124.2 km/h;

5)A hybrid electric vehicle may be the solution for less fuel consumption, less emissions and large range of functioning.
The ECE and EUDC test were performed for series configuration with a 47 kW induction motor, a 47 kW internal combustion engine ,1323 kg, MI-PbA batteries and thermostat strategy (50% hybridization) in order to compare with a 94 kW conventional vehicle. The results over ECE cycles were satisfactory. Fuel consumption starting the test with 50% SOC is 2.5l/100km and the level of SOC at the end of cycles is 42%. Starting the same test with initial SOC 100% the fuel consumption will be lower.
The highway consumption is 6.7l/100km and at the end of the tests the SOC level is 55%. This result is better than the result of 94 kW conventional car test, and worse than the electrial car but the major advantage is a higher level on SOC even that the initial level was 50% not 100% as in the elctrical vehicle case.

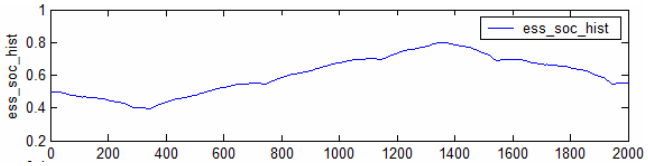


Fig.7 SOC history (EUDC).

Thedynamic performance obtained are:
- acceleration time 0-100km/h 11.5s;
- maximum acceleration 5m/ s^2 ;
- maximum speed 151.7 km/h;
Overall System Efficiency over ECE cycles is 0.132.
Overall System Efficiency over EUDC cycles is 0.154

6)For the same configuration as previously with Load Follower strategy the result of urban consumption test was not satisfactory compared with the other cases. Fuel

consumption starting the test with 50% SOC is 15l/100km and the level of SOC at the end of the drive cycles is better than the previous case:61%. With the same initial SOC of 50% the highway consumption is 7l/100km and the final level of storage system is 55%. The consumption is higher than previously and SOC is the same and the dynamic performances are:

- acceleration time 0-100km/h 11.6s;
- acceleration time 30-60km/h 3s;
- maximum acceleration 5m/s^2 ;
- maximum speed 157.3 km/h;

Overall System Efficiency over ECE cycles is 0.038.
Overall System Efficiency over EUDC cycles is 0.151.

7)For parallel configuration with an 47kW induction motor, an 47kW ICE, 1333 kg, MI-PbA batteries and thermostat strategy (50% hybridization) the results over ECE cycles were good to consider.

Urban fuel consumption starting the test with 50% SOC is 14.5l/100km and 56% final level for SOC. A major difference can be observed in fuel consumption if the starting level of SOC is 70%(6.9l/100 km).

The highway consumption (7.3l/100km) is not high even the starting SOC is 50% and 61% remaining SOC. With starting SOC 70% the fuel consumption is 5.5l/100 km.

As dynamic performance :

- acceleration time 0-100km/h 9.1s;
- maximum acceleration 5m/s^2 ;
- maximum speed 197.6 km/h;

For initial SOC=50%:
Overall System Efficiency over ECE cycles is 0.037.
Overall System Efficiency over EUDC cycles is 0.146.

IV. Conclusion

The conventional vehicle has a large operating range, good performance, but a high fuel consumption especially at start, and high emissions. Electric vehicle with Pb batteries has no emissions but a small operating range.

The electric vehicle with LI batteries weighing less can provide a boost of power when needed but for short time, better performances compare to Pb electrical vehicle, but has a smaller operating range.

For series architecture the thermostat control strategy is better than the Load Follower.

The best configuration is parallel, and that's why many companies has chosen it for their models.(Toyota,Honda).

References

[1] Aaron Brooker, Kristina Haraldsson, Terry Hendricks, Valerie Johnson, Kenneth Kelly, Bill Kramer, Tony Markel, Michael O'Keefe, Sam Sprik, Keith Wipke, Matthew Zolot, "*ADVISOR Documentation*",.NREL, 2002.

[2] Mehrdad Ehsany, Yimin Gao, Sebastien E. Gay, Ali Emadi, *Modern Electric, Hybrid Electric, and Fuel Cell Vehicles*. Boca Raton, CRC Press, 2004..

[3] Allen E. Fuhs, *Hybrid Vehicles and the future of personal transportation*. Boca Raton, CRC Press, 2009.

[4] Iqbal Husain, *Electric and Hybrid Vehicles - Design Fundamentals*, Boca Raton, CRC Press, 2003.

[5] Weng Keong Kevin Lim, Saman Kumara Halgamuge, and Harry Charles Watsson *"Drive and Control System for Hybrid Electric Vehicles"*, in *Handbook of Automotive Power Electronics and Motor Drives* , (Ed. AliEmadi), Taylor & Francis Group, LLC, 2005.

Robust Model Predictive Control For Time-varying Systems

Mitra Bahadorian, Borislav Savković, Ray Eaton, Tim Hesketh

Abstract— **This paper introduces a new approach to robustify model predictive controller (MPC) when applied to time-varying systems, subject to unknown but bounded disturbances. The main idea is to calculate a disturbance invariant set and develop a tube-based control for the time varying system by restricting (i.e. tightening) the constraints of a nominal disturbance-free system and using an error adjustment mechanism simultaneously to achieve constraint satisfaction of actual system and feasibility of finite horizon optimal control problem. Dealing with a time variant system, the advantage of this method is that constraints restrictions are time-invariant which saves the controller from high computational complexity burden. The presented theory is illustrated with a simulation example to show the efficacy and performance of the proposed control method.**

Keywords: Model Predictive Control, Constraint tightening, Time varying system, Robust Control

[1]

I. Introduction

Model Predictive Control (MPC) or Receding Horizon Control (RHC) was introduced in the 1980's with first applications in process control [7]. Since then, it has become a popular approach and has been considerably developed. MPC is a form of iterative optimal control which acts upon the plant model and past and future behavior of the system. The main difference between MPC and other conventional control approaches is that the control action in MPC is obtained online through a finite horizon open-loop control, while the other methods just use a pre-computed control law. The other strength of this approach is its ability to cope with constraints on a system, which makes it widely utilized for control of constrained systems. However, despite all the strengths of MPC, robustness of conventional MPC is still an open problem. The problem arises due to the fact that every plant model is actually an approximation of the real system, and thus there would be a model-plant mismatch and other internal uncertainties in most of the systems. On the other hand, external disturbances and uncertainties acting on a plant are always a matter of concern for control systems.

Due to its importance, robustness of MPC systems has been tackled in several ways due to importance of robustness issue. Comprehensive surveys on robust MPC can be found in [6],[7],[8]. Campo et al [9], introduced the open-loop min max MPC to cope with robustness, however a counterexample by [14] demonstrated that the min-max paradigm by itself is not enough to guarantee robustification. In [10] and [11], Kothare et al, formulated the problem of robustness with the assumption of full state feedback, employing a Linear Matrix Inequality (LMI) based optimization [20]. In [1], Mayne et al proposed a robust MPC scheme using tubes for time-invariant systems. The main idea behind tube-based control is to achieve closed-loop control of the nominal uncertainty-free system with restricted constraints tighter than the ones of the actual system, based on the suitably defined "safety margins", and adjust the control law with an additive correction term to compensate for disturbances and attain robust control of the perturbed systems. In [12] and [13], Richards proposed a robust MPC for time-varying systems, using the same concept of constraint tightening. Mayne et al in [5] also introduced a robust output feedback MPC for linear systems with input and output disturbances in which the initial uncertainty set is high. In [15] Savković employed a time-variant mechanism to adjust the safety margins to improve the performance of MPC in the presence of time-variant uncertainty.

This paper extends the results from [2],[3],[4] to formulate a robust MPC scheme for systems with time-varying dynamics subject to bounded disturbances. The approach of this work is to calculate a robust positively invariant (RPI) set [18] for a given disturbance set as described in [17] and [16] and tighten the constraints on the disturbance-free (nominal) system. The level of constraint tightening is verified by suitably defined "safety margin sets". These sets make enough space for the input to compensate for uncertainties on the input side. The difference between the nominal and perturbed system then will be compensated through an adjusting control law [1]. Since the restricted constraints are adjusted in accordance to the corresponding RPI set, any uncertainty occurring to the system can be accommodated properly by the main control input accompanied by the adjusting control law. The particular formulation of control law applied to the actual system ensures that all the states of the system evolve around the nominal state of the system within the safety margin. It also guarantees that inputs of the system will be hovering around the corresponding nominal inputs.

The contribution of this work is to incorporate a tube MPC controller to time-varying systems subject to unknown bounded disturbances to develop robust time-varying MPC exploiting RPI sets. The idea behind this work is the same as [1], but is extended to be applied to time-varying systems using the RPI set formulation proposed by [16]. The proposed robustification approach is particulary novel in the sense that constraint restriction does not change with the time for the time-varying systems. This considerably saves the

The authors are with The School of Electrical Engineering and Telecommunications, The University of New South Wales, Sydney NSW 2052, Australia. email :{mitrab,b.savkovic,r.eaton,t.hesketh} @unsw.edu.au

system from reconfiguration and expensive computations due to Minkowki sums and Minkowki differences. (See [22] for a good overview of computational issues involved with Minkowski sums and differences.)

The rest of this paper if organized as follows: Section II, provides the reader with necessary definitions and preliminary results. In section III, a time-varying system is briefly presented. Section IV, presents a new robust MPC paradigm for time-varying systems. In section V, stability and performance of aforementioned robust MPC have been analyzed. Section VI, demonstrates the results of proposed robust MPC controller in simulation and finally in section VII, concluding remarks are presented.

Notation: An empty set is denoted by $\emptyset$. With $\mathbb{S}_1, \mathbb{S}_2 \subseteq \mathbb{R}^n$, $\mathbb{S}_1 \oplus \mathbb{S}_2 = \{a + b : a \in \mathbb{S}_1, b \in \mathbb{S}_2\}$ denotes the Minkowski sum and $\mathbb{S}_1 \ominus \mathbb{S}_2 = \{z : \forall b \in \mathbb{S}_2, (z + b) \in \mathbb{S}_1\}$ denotes the Minkowski difference between $\mathbb{S}_1$ and $\mathbb{S}_2$. If $\{\mathbb{S}_i\}_{i=1}^m$ is a collection of m sets, with $\mathbb{S}_i \subseteq \mathbb{R}^n$, then $\sum_{i=1}^m \mathbb{S}_i$ denotes their Minkowski sum defined by $\sum_{i=1}^m \mathbb{S}_i = \mathbb{S}_1 \oplus \mathbb{S}_2 \oplus \ldots \oplus \mathbb{S}_m$.

II. Preliminary definitions and results

The following definitions will be required in this paper:

*Definition 1 (**Stable Matrix**):* If all eigenvalues of matrix $M \in \mathbb{R}^{n \times n}$ lie strictly inside the unit circle in the complex plane, matrix M is stable.

*Definition 2 (**Polytopic matrix set**):* A polytopic matrix set $\mathbb{N}$ is defined as $\mathbb{N} \triangleq \{N | N = \sum_{i=1}^q \lambda_i N_i; \sum_{i=1}^q \lambda_i = 1\}$ where $N_i \in \mathbb{R}^{n \times n}$, $\lambda_i \in \mathbb{R}$ and $\lambda_i \geq 0$

*Definition 3 (**Linear Difference Inclusion**):* A linear difference inclusion is a relation of the following form:

$$\begin{aligned} & x(t+1) = Ax(t) + w(t) \\ & A \in \mathbb{A}_c \\ & w \in \mathbb{W} \end{aligned} \tag{1}$$

with $x \in \mathbb{R}^n$ denoting the current state of a system, $x(t)$ denoting the state of the system and with $w \in \mathbb{W} \subseteq \mathbb{R}^n$ representing an unknown disturbance, where the set $\mathbb{W}$ represents the set of all possible disturbances, and $\mathbb{A}_c$ denoting a polytopic matrix set.

*Definition 4 (**Robust Positively Invariant set**):* A set R_∞ is a robust positively invariant (RPI) set for a linear differential inclusion as given by expression (1) if R_∞ satisfies $AR_\infty \oplus \mathbb{W} \subseteq R_\infty$ for any $A \in \mathbb{A}$. See [16] for more details.

*Definition 5 (**Minimal Robust Positively Invariant set**):* A set is a minimum RPI set for the linear difference inclusion as given by expression (1) if it is the smallest such RPI set. See [16] for more details.

Remark 1: In predictive controllers, invariant sets are used to guarantee constraint satisfaction, stability and convergence properties and that actual states evolve around nominal and uncertainty free state.

Remark 2: Usually we find the approximation to the mRPI set using the method from [16], however the computational complexity grows exponentially as the quality of approximation improves.

III. The time-varying system

The aim of this work is to control a discrete-time linear system with time-varying dynamics given by:

$$x(t+1) = A(t)x(t) + B(t)u(t) + w(t) \tag{2}$$

where $x(t) \in \mathbb{R}^n$ is the system state, $u(t) \in \mathbb{R}^m$ is the controlled input and $w(t) \in \mathbb{R}^n$ represents the disturbance vector. We also assume that state matrices $A(t)$ and $B(t)$ are time varying matrices belonging to polytopic matrix sets $\mathbb{A}$ and $\mathbb{B}$ respectively, where vertices of polytopic set $\mathbb{A}$ are A_1, A_2, A_3 etc. and vertices of polytopic set $\mathbb{B}$ are B_1, B_2, B_3 etc.

The system (2) is subject to state and and input constraints

$$u(t) \in \mathbb{U},\ x(t) \in \mathbb{X},\ t \geq 1 \tag{3}$$

where $\mathbb{U} \subseteq \mathbb{R}^m$ and $\mathbb{X} \subseteq \mathbb{R}^n$ are both convex polytopic sets and contain the origin. For each time step, the corresponding disturbance $w(t)$ is assumed to be unknown but bounded. The disturbance set $\mathbb{W}$ contains the origin and is defined as follow:

$$w(t) \in \mathbb{W} \subseteq \mathbb{R}^n,\ t \geq 1 \tag{4}$$

To control the system, we are looking for a control policy which would cope with time-varying system to account for the disturbances, while satisfying all of the constraints of the system and lowering the computational complexity. This will be discussed in the next section.

IV. Robust Time-variant MPC

The objective is to robustify the uncertain time-variant system (2). Let us define the nominal(reference) system as follows:

$$\overline{x}(t+1) = A(t)\overline{x}(t) + B(t)\overline{u}(t) \tag{5}$$

where $\overline{x}(t) \in \overline{\mathbb{X}}$ denotes the nominal state and $\overline{u}(t) \in \overline{\mathbb{U}}$ denotes the open-loop control input for the nominal system (5). Later on in this section, $\overline{\mathbb{X}}$ and $\overline{\mathbb{U}}$ sets will be formulated to form tighter constraints than the constraints of actual system.

*Definition 6 (**Time-varying Robust Control Policy**):* With $x(t)$ denoting the actual state of the system, $\overline{x}(t)$ denoting the nominal state of the system and $\overline{u}$ denoting the control input to the nominal disturbance-free system, the robust MPC control policy for the actual system (2) is defined as $u(t)$, i.e:

$$u(t) = \overline{u}(t) + K(x(t) - \overline{x}(t)) \tag{6}$$

$$\overline{u}(t) = \overline{u}^*(0|t) \tag{7}$$

$$\{\overline{u}^*(k|t)\}_{k=0}^{N-1} = \min_{\{\overline{u}(k|t)\}_{k=0}^{N-1}} \overline{J}(\overline{x}_t, \overline{u}_t) \tag{8}$$

where $K \in \mathbb{R}^{m \times n}$ denotes a corrective feedback term which enforces $x(t)$ to evolve as close as possible to nominal state $\overline{x}(t)$ and the cost function $\overline{J}(\overline{x}_t, \overline{u}_t)$, as defined in expression (9) below, denotes the objective cost function of the finite horizon optimal control.

To develop the nominal MPC control policy for system (5), let N be a positive integer , denoting the optimization horizon of finite horizon MPC and $Q \in \mathbb{R}^{n \times n}$ and $R \in \mathbb{R}^{m \times m}$ be

symmetric positive definite weighting matrices. The relevant MPC optimization problem is then given by:

Finite Horizon Optimization Problem $P_t(\overline{x}_0)$

$$\overline{J}(\overline{x}_t, \overline{u}_t) = \sum_{k=0}^{N-1} ||\overline{x}(k|t)||_Q^2 + ||\overline{u}(k|t)||_R^2 + (1/2)||\overline{x}(N|t)||_{Q_f}^2 \quad (9)$$

subject to constraints:

$$\overline{x}(k|t) \in \overline{\mathbb{X}} = \mathbb{X} \ominus Z \quad (10)$$

$$\overline{u}(k|t) \in \overline{\mathbb{U}} = \mathbb{U} \ominus KZ \quad (11)$$

$$\overline{x}(N|t) \in \overline{\mathbb{X}}_f \quad (12)$$

where:

$$\overline{x}(0|t) = \overline{x}(t) \quad (13)$$

$$\overline{x}(k+1|t) = A(k+t)\overline{x}(k|t) + B(k+t)\overline{u}(k|t) \quad (14)$$

$$0 \leq k \leq N-1 \quad (15)$$

- N denots a fixed prediction horizon for the finite horizon optimization problem of time-varying MPC.
- The notation $(k|t)$ denotes the prediction $k \geq 0$ steps ahead from current time.
- $P_t(\overline{x}(t))$ denotes a finite horizon optimization problem which at every time step t performs a predictive optimization of the future evolution of the nominal and uncertainty-free system (5), based on the current nominal state "$\overline{x}(t)$".
- The set $\overline{\mathbb{X}}_f \subseteq \mathbb{R}^n$ denotes the terminal set of the finite horizon optimization problem which in this work has been set to 0 for the sake of simplicity.
- With Z denoting a RPI set for the linear difference inclusion given by

$$\begin{aligned} &\delta(t+1) = (A(t) + B(t)K)\delta + w \\ &A(t) + KB(t) \in \mathbb{A}_c \\ &w \in \mathbb{W} \end{aligned} \quad (16)$$

  where $\mathbb{A}_c$ is a matrix polytopic set since $A(t) \in \mathbb{A}$ and $B(t) \in \mathbb{B}$ and $\mathbb{A}$ and $\mathbb{B}$ are matrix polytopic sets, i.e. Z satisfies $(A(t) + KB(t))Z \oplus \mathbb{W} \subseteq Z$, the purpose of Z and linear difference inclusion given by expression (1) will become apparent through lemma (1) below.

With $\delta(t)$ denoting the difference between the actual and nominal state defined by:

$$\delta(t) = x(t) - \overline{x}(t) \quad (17)$$

The following lemma holds:

Lemma 1: $\delta(t)$ evolves according to a linear difference inclusion given by :

$$\begin{aligned} &\delta(t+1) = (A(t) + B(t)K)\delta(t) + w(t) \\ &A(t) + B(t)K \in \mathbb{A}_c \\ &w \in \mathbb{W} \end{aligned} \quad (18)$$

Proof: Employing expressions (2), (5) and (6), this claim follows immediately from the results in [1] applied to the current case involving matrix polytopic sets, which results in the linear difference inclusion given by expression (18). ■

Remark 3: Lemma 1 ensures that $\delta(t)$ which is the difference between the actual and nominal system, evolves according to linear difference inclusion. Note that if $\delta(1) \in Z$ then $\delta(t) \in Z$ for any $t > 1$. This follows as Z is a RPI set for the linear difference inclusion in expression (18).

The following algorithm can be used to formulate the robust MPC control law:

Algorithm 1 (Robust Time-variant MPC)

1) *Offline;* Find the stabilizing feedback control K
2) *Offline;* Calculate the RPI set Z
3) *Online;* For each time iteration t, solve the problem $P_t(\overline{x}(t))$ to find control input of the nominal system
4) *Online;* Use expression (6) to find control input of the actual system
5) *Online;* Increment t, go to step 3

The first step in the algorithm is to find the feedback control K to stabilize the system. Different methods such as pole placement or an LQR approach can be utilized to calculate the feedback control K. The next step is to calculate the corresponding RPI robust invariant set Z. Kouramas et al [16] proposed an efficient method to calculate the robust invariant set for time-variant systems. The RPI set can be formulated offline if all the state matrices are available, hence it is sufficient to calculate the RPI set just once not for each time step. This is considerably important since calculation of the RPI set can be computationally expensive for higher dimension systems, hence it is beneficial not to form this set for each iteration. It should be noted that formulation of the RPI set Z proposed by [16] is conservative and may not be always possible, particulary for higher dimension systems. This paper is based on the assumption that the RPI set can be calculated. Once both stabilizing feedback control and the RPI set are formulated, the next step is to iteratively solve the optimal control problem $P_t(\overline{x})$. This leads to a control law $\overline{u}(t)$, and followed by that the control policy $u(t)$ can be calculated through (6).

Since constraints of the nominal system are tightened as (10) and (11), the sets Z and KZ can be interpreted as safety margins employed to calculate the nominal constraint sets $\overline{\mathbb{X}}$ and $\overline{\mathbb{U}}$. This ensures evolution of nominal system within tighter constraints than the actual system.

V. Stability and performance analysis of time-variant robust MPC

In this section, the definition of controller and its stability and feasibility properties are analyzed. Let $\overline{\Omega}_1$ be the region where $P_1(\overline{x}(1))$ is feasible, i.e. if $\overline{x}(1) \in \overline{\Omega}_1$, then there exists a solution $\{\overline{u}(k|1)\}_{k=0}^{N-1}$ such that constraints given by expressions (10) to (14) are satisfied for problem $P_1(\overline{x}(1))$. We also say $P_t(\overline{x}(t))$ is feasible if there exists a sequence $\{\overline{u}(k|t)\}_{k=0}^{N-1}$ such that constraints given by expressions (10) to (14) are satisfied. Assuming that $\overline{\mathbb{X}} \neq \emptyset$, $\overline{\mathbb{U}} \neq \emptyset$ and $\Omega = \overline{\Omega} \oplus Z$,

the following theorem will establish the feasibility and stability properties of the proposed control law as well as satisfying constraints for the aforementioned system .

Theorem 1: Let $x(1) \in \Omega$, $\overline{x}(1) \in \overline{\Omega}(1)$ and $x(t) - \overline{x}(t) \in Z$. Then the following hold for $t \geq 1$

1) $P_t(\overline{x}(t))$ is feasible.
2) $\overline{x}(t) \in \overline{\mathbb{X}}(t)$ and $\overline{u}(t) \in \overline{\mathbb{U}}(t)$
3) $lim_{t\to\infty}\overline{x}(t) = 0$ and $lim_{t\to\infty}\overline{u}(t) = 0$
4) $x(t) \in \overline{x}(t) \oplus Z$ and $u(t) \in \overline{u}(t) \oplus KZ$
5) $x(t) \in \mathbb{X}$ and $u(t) \in \mathbb{U}$

Proof: **Claim 1**: By assumption, $\overline{x}(1) \in \overline{\Omega}_1$ i.e. $P_1(\overline{x}_1(1))$ is feasible. By an induction argument analogous to the standard feasibility proof for time-invariant MPC (see e.g. [7] or page 98 of [19]), it follows that $P_t(\overline{x}_t(t))$ is feasible for all $t > 1$. The argument essentially involves showing that the optimal solution at time t given by $\{\overline{u}^*(0|t), \overline{u}^*(1|t), ...\overline{u}^*(N-1|t)\}$, can be used to propose a feasible solution at time $t+1$ given by $\{\overline{u}^*(1|t), \overline{u}^*(2|t), ...\overline{u}^*(N-1|t), 0_m\}$ where $0_m \in \mathbb{R}^m$ denotes the vector with all entries equal to zero.

Claim 2: Note that $\overline{x}(0|t) \in \overline{\mathbb{X}}$ and $\overline{u}(0|t) \in \overline{\mathbb{U}}$, from expressions (10) and (11) respectively. Also note that $\overline{\mathbb{X}}(0|t) = \overline{x}(t)$ and $\overline{\mathbb{U}}(0|t) = \overline{u}(t)$ from expressions (13) and (7) respectively. Hence, the claim follows immediately.

Claim 3: Let the minimum value of the cost function (9) at time t be defined as $\overline{V}(\overline{x}(t))$, i.e. $\overline{V}(\overline{x}(t)) = \min_{\overline{u}(t)} \overline{J}(\overline{x}_t, \overline{u}_t)$. The proof is analogous to the Lyapunov stability proof for MPC, as given on page 91 of [19]. First, by an identical argument to the proof for standard MPC (see page 99 of [19]), an upper bound for the expression $V(\overline{x}(t+1)) - V(\overline{x}(t))$ is obtained, to give $V(\overline{x}(t+1)) - V(\overline{x}(t)) \leq -\overline{x}(t)'Q\overline{x}(t) - \overline{u}(t)'R\overline{u}(t)$, i.e. $V(\overline{x}(t+1)) \leq V(\overline{x}(t)) - \overline{x}(t)'Q\overline{x}(t) - \overline{u}(t)'R\overline{u}(t)$, which implies $V(\overline{x}(t)) \leq V(\overline{x}(1)) - \sum_{k=1}^{t-1}(\overline{x}(t)'Q\overline{x}(t) + \overline{u}(t)'R\overline{u}(t))$. Letting $t \to \infty$, then, since the function V is non-negative (i.e. $V(\overline{x}(t)) \geq 0$ for all $t \geq 1$), it follows that $0 \leq V(\overline{x}(1)) - \sum_{k=1}^{\infty}(\overline{x}(k)'Q\overline{x}(k) + \overline{u}(k)'R\overline{u}(k))$. From this latter expression, and the fact that R and Q are positive definite, it follows that if either $\lim_{t\to 0}\overline{x}(t) = 0$ or $\lim_{t\to 0}\overline{u}(t) = 0$ is false, then a contradiction is obtained as $\sum_{k=1}^{\infty}(\overline{x}(k)'Q\overline{x}(k) + \overline{u}(k)'R\overline{u}(k)) = \infty$, i.e. the sequence diverges.

Claim 4: Since $\delta(t) \in Z$, it follows that $x(t) - \overline{x}(t) \in Z$, and by Claim 2, it is given that $\overline{x}(t) \in \overline{\mathbb{X}}$ which leads to $x(t) \in \overline{x}(t) \oplus Z$. It also follows from (6) that $u(t) = \overline{u}(t) + K(\delta(t))$ and $\delta(t) \in Z$, therefore $u(t) \in \overline{u}(t) \oplus KZ$.

Claim 5: Since $x(t) \in \overline{x}(t) \oplus Z$ by Claim 4 of this Theorem, since $\overline{x}(t) \in \overline{\mathbb{X}}$ by Claim 2 of this Theorem, and since $\overline{\mathbb{X}} = \mathbb{X} \ominus Z$ from (10), it follows that $x(t) \in [\mathbb{X} \ominus Z] \oplus Z$. It thus follows by Claim (ii) of Theorem 2.1 from [18], that $[\mathbb{X} \ominus Z] \oplus Z \subseteq \mathbb{X}$. Thus $x(t) \in \mathbb{X}$. By an analogous proof, it may also be shown that $u(t) \in \mathbb{U}$. ■

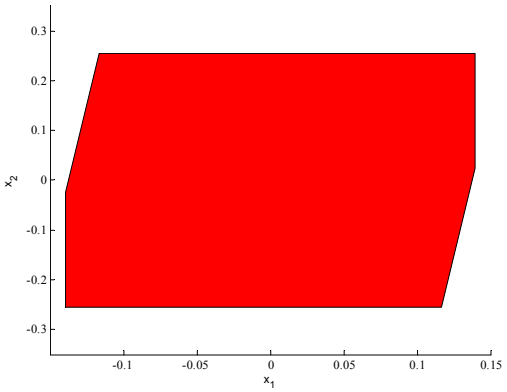


Fig. 1. *Approximation of mRPI set with* $\epsilon = 0.01$ *for system* S (see [16])

VI. Illustrative example

In this section, the proposed method is illustrated by considering a time variant discrete time system S with unknown but bounded disturbance i.e.:

$$S : x(t+1) = A(t)x(t) + B(t)u(t) + w(t)$$

with $A(t) \in \mathbb{A}$ where the vertices of matrix polytope $\mathbb{A}$ are given by $\{A_1, A_2, A_3, A_4\}$ where:

$$A_1 = \begin{pmatrix} 1.1 & 1 \\ 0 & 1 \end{pmatrix}, A_2 = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$$

$$A_3 = \begin{pmatrix} 0.8 & 1 \\ 0 & 1 \end{pmatrix}, A_4 = \begin{pmatrix} 0.6 & 1 \\ 0 & 1 \end{pmatrix}$$

Also $B(t) = B$ where:

$$B = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

The constraint sets are defined as:

$$\mathbb{X} = \{\begin{bmatrix} x_1 & x_2 \end{bmatrix}' : -3 \leq x_1 \leq 3\,,\ -3 \leq x_2 \leq 3\}$$
$$\mathbb{U} = \{u : -1 \leq u \leq 1\}$$
$$\mathbb{W} = \{\begin{bmatrix} w_1 & w_2 \end{bmatrix}' : -0.1 \leq w_1 \leq 0.1\,,\ -0.1 \leq w_2 \leq 0.1\}$$

The initial state of the system is chosen as: $x(1) = \begin{pmatrix} -2.9 & 2.75 \end{pmatrix}'$. The weighting matrices are chosen as $Q = 10I_2$, $R = 1$ where I_2 denotes a 2 by 2 identity matrix. The prediction horizon is chosen as $N = 10$.
Hereby, the feedback stabilizing matrix K is calculated as $K = [-1.2\ -1]$. We used pole placement to calculate this matrix with desired poles placed very close to the origin. Stability of the system is then checked by LMI formulation.

The simulation is performed by employing the Multi Parametric Toolbox (MPT)[21]. A random but bounded disturbance is imposed to the system beginning at $t = 0$ and stoping at $t = 20$. In each time step, state matrix $A(t)$ is chosen randomly as one of the vertices A_1, A_2, A_3 or A_4. In the presence of the disturbance, the actual states of the system

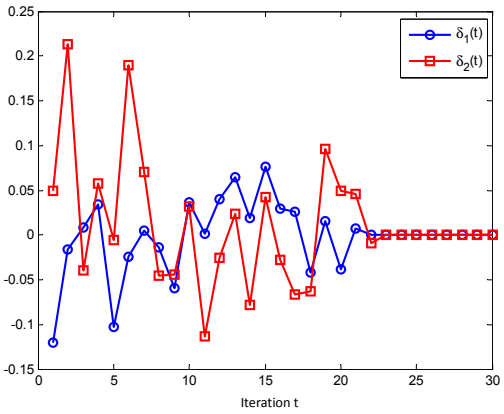


Fig. 2. *Plot of* $\delta_1(t)(blue), \delta_2(t)(red)$

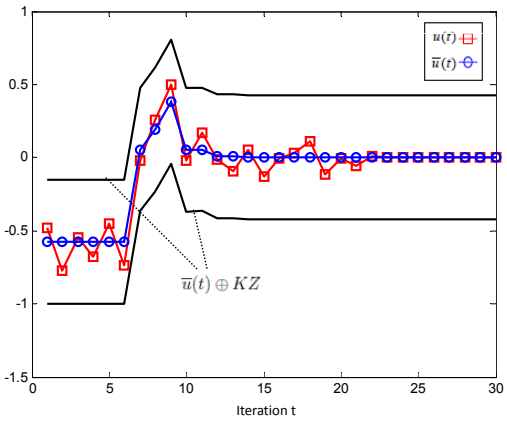


Fig. 3. *Plot of* $u(t)$, $\overline{u}(t)$ and $\overline{u}(t) \oplus KZ$

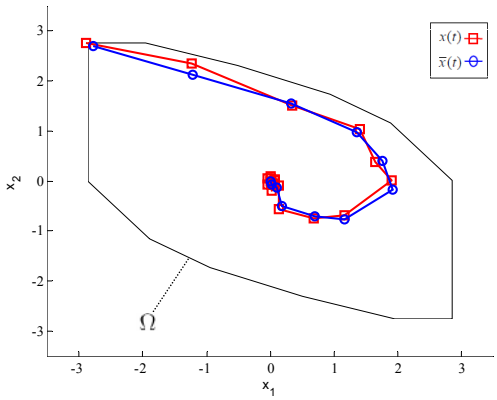


Fig. 4. *Plot of region of attraction* Ω, $x(t)$ *and* $\overline{x}(t)$ *trajectories*

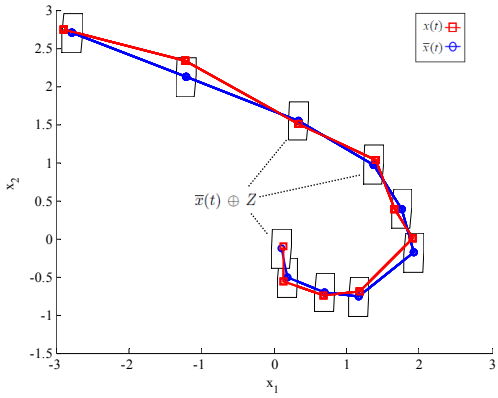


Fig. 5. *Plot of* $x(t)$, $\overline{x}(t)$ *and* $\overline{x}(t) \oplus Z$ *for* $t = 1:10$

evolves around the nominal states without violating the constraints. Once the uncertainty stops, the actual states converge to the nominal states quickly. Figure (1) shows the smallest RPI set Z for system S. Let $\delta(t) = \begin{pmatrix} \delta_1(t) & \delta_2(t) \end{pmatrix}'$. Figure (2) shows the difference between nominal and actual systems, $\delta_1(t)$ and $\delta_2(t)$. It can be seen that for the first 20 iterations where disturbance exists, there exists some difference between nominal and actual system which corresponds to disturbance, however it is a controlled difference, which means that it is bounded and does not explode. In figure (3) it can be seen that the control inputs stay within the input constraints, evolving around the nominal inputs of the system within the safety margin set KZ. Figure (4) illustrates that states of the system remain within the set Ω for all the time. Figure (5) shows an enlarged plot of state space over iteration $1 \leq t \leq 10$. It can be seen that seen that the actual states $x(t)$ evolve around the nominal states $\overline{x}(t)$ within the safety margin set Z.

From the simulation results, it can be easily seen that the proposed controller performs as expected. All constraints are satisfied and the system is stable. There are no constraints violated in the presence of uncertainty.

VII. Conclusion

In this paper as an extension of existing research, a robustified time varying MPC control is presented. The proposed approach employs a time variant adjusting mechanism to cope with disturbances which improves the performance of the system. Under the assumption that disturbances are bounded, feasibility of optimization and satisfying the constraints are all guaranteed. The proposed controller has been applied to a time varying system to illustrate the performance and efficiency of the controller. Future work will focus on efficient computational methods, especially to incorporate with robust positive invariant set formulation.

References

[1] D.Q. Mayne and W. Langson, *Robustyfying model predictive control of constrained linear systems*, Electronics Letters, Vol 37, No 23, 8 Nov 2001, Pages 1422 – 1423.

[2] D.Q. Mayne,M.M. Seron,S.V. Rakovic, *Robust model predictive control of constrained linear systems with bounded disturbances*, Automatica, Vol 41, Issue 2, Feb 2005, Pages 219 – 224.

[3] D.Q. Mayne,S.V. Rakovic,R. Findeisen,F. Allgoewer, *Robust output feedback model predictive control of constrained linear systems*, Automatica, Vol 42, Issue 7, July 2006, Pages 1217 – 1222.

[4] W. Langson,I. Chryssochoos,S.V. Rakovic,D.Q. Mayne, *Robust model predictive control using tubes*, Automatica, Vol 40, Issue 1, Jan 2004, Pages 125 – 133..

[5] D.Q. Mayne,S.V. Rakovic,R. Findeisen,F. Allgoewer, *Robust output feedback model predictive control of constrained linear systems: Time-varying case*, Automatica (2009), doi:10.1016/j.automatica.2009.05.009.

[6] C.E. Garca, D. M. Prettb, M. Moraria, *Model predictive control: Theory and practiceA survey*, Automatica, Vol25, Issue 3, May 1989, Pages $335 - 348$

[7] D. Q. Mayne, J. B. Rawlings, C. V. Rao, P. O. M. Scokaert, *Constrained model predictive control: Stability and optimality* , Automatica 2000, Vol36, Pages $789 - 814$

[8] A. Bemporad, M. Morari, *Robust model predictive control, a survey*, Robustness in identification and control , Vol 245, doi:10.1007/BFb0109854. 1999

[9] P. Campo, M. Morari, *Robust model predictive control* , In *Proceedings of the 6th American Control Conference*, Minneapolis, United States, 1987.

[10] M. Kothare, V. Balakrishnan, M. Morari, *Robust model predictive control using linear matrix inequalities*, Automatica, Vol 23, No 10, Pages $1361 - 1379$

[11] Z. Wan, M. V. Kothare, *An efficient off-line formulation of robust model predictive control using linear matrix inequalities*, Automatica 2003, Vol 39, Pages 837846

[12] A. Richards, *Robust Model Predictive Control for Time-Varying Systems*, *Proceedings of the 44th International Conference on Decision and Control*, Seville, Spain, 2005

[13] A. Richards, *Robust Constrained Model Predictive Control*,Ph.D. thesis. Massachusetts Institute of Technology, 2005

[14] P.Q. Zheng, *Robust control of systems subject to constraints*, Ph.D. dissertation, California Institute of Technology, Pasadana, CA, USA,2005

[15] B. Savković, *Time-variant robust model predictive control under communication constraints*, *Proceedings of the American Control Conference 2010 (ACC 2010)*,Baltimore, Maryland, USA, 2010

[16] K. I. Kouramas, S. V. Raković, E. C. Kerrigan, J. C. Allwright. D. Q. Mayne, *On the Minimal Robust Positively Invariant Set for Linear Difference Inclusions*, *Proceedings of the 44th International Conference on Decision and Control*, Seville, Spain, 2005

[17] S.V. Raković,C. Kerrigan,K.I. Kouramas,D.Q. Mayne, *Invariant Approximations of the Minimal Robust Postively Invariant Set*, IEEE Transactions on Automatic Control, Vol 50, No. 3, March 2005.

[18] I. Kolmanovsky,E.C. Gilbert, *Theory and computation of disturbance invariant sets for discrete-time linear systems*, Mathematical Problems in Engineering, Vol 4 (1998), Issue 4, Pages $317 - 367$.

[19] G.C. Goodwin, M.M. Seron, J.A De Dona *Constrained Control and Estimation, An Optimization Approach*, Springer-Verlag, 2005.

[20] S. Boyd, E. E. Ghaoui, E. Feron, V. Balakrishnan, *Linear matrix inequalities in system and control theory*, Studies in applied mathematics, 1994

[21] M. Kvasnica,P. Grieder,M. Baotic,F.J. Christophersen, *Multi-Parametric Toolbox (MPT)*. Available online under: http://control.ee.ethz.ch/$\sim$mpt/

[22] M. Vasak, (2000). *Time Optimal Control of Piecewise Affine Systems.* Ph.D. dissertation, University of Zagreb, 2000, Available online at http://act.rasip.fer.hr/pub-opis.php?id=375. 187, 190, 191, 197, 198

Functional analysis of a communication framework used in a modular simulator

Lucian-Florentin M. Barbulescu

Abstract—**Building a modular simulator implies using some kind of communication environment to exchange data between the components. This environment must be reliable, and most important, very fast in order to keep the global execution time to a minimum. This paper presents a functional analysis for a communication framework that can be used to interconnect a set of modules in order to build a simulator for an industrial installation.**

I. Introduction

WHEN simulating an industrial installation one approach is to break it into components, create an independent elementary simulator for each one, and then interconnect those simulators with some kind of communication channels. Using this approach any complex installation can be reduced to a finite set of basic component which reduces the development time. Also, the individual components can be placed on different machine and thus can use the processing power of a network in order to complete the task. Another advantage is that, if an installation contains several components of the same type, only one simulator for that component can be created and several instances of it will be used to achieve the final goal. The "write once, use many" aspect must also not be ignored. Creating an elementary simulator for one installation does not necessarily means that it cannot be used to simulate another system. By contrary, industrial installations are made of several common components (pumps, vanes, engines etc.) and creating a library of elementary simulators can be a good approach.

But the modular simulators cannot run without some good communication channels. Those channels must be reliable, fast and easy to use in order to obtain the best possible results. Also, another function that those channels can provide is that of creating a distributed system by allowing the elementary simulators to reside on different physical machine and offering to them the possibility of interconnection via a computer network. For example, consider that we have an installation which is simulated with the aid of seven elementary simulators interconnected via four communication channels.

Lucian-Florentin M. Barbulescu is with the University of Craiova, Faculty of Automation, Computers and Electronics, Computer Science Department, Bvd. Decebal, Nr. 107, 200440, Craiova, Dolj, ROMANIA (phone: +40745343516; fax: +40251438198; e-mail: lucian.barbulescu@cs.ucv.ro).

.

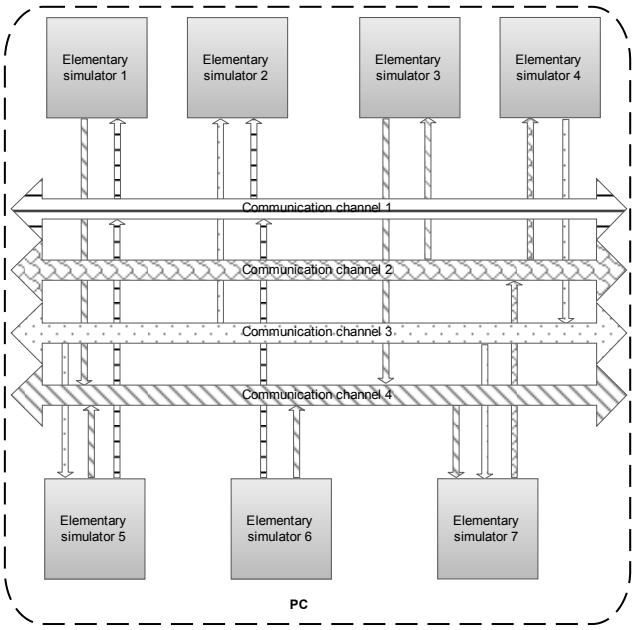


Fig. 1. Block diagram of a modular simulator. The system contains seven elementary simulators interconnected via four communication channels. All elements reside on the same physical machine which offers a limited processing power.

This installation is placed on a single machine and its functionality is limited by the processing power, the memory size, the operating system etc. It may be desired that this system to be implemented using several physical machines interconnected via a network and thus using the resources of several computers. The disadvantage in this case can be the speed of the network, but in our days, when 100Mbit/s is the minimum available, this can be easily ignored.

To create the distributed system one has to define some physical boundaries. The elementary simulators are designed as monolithic blocks and thus each one must reside on only one machine. The communication channel also is a monolithic block and it must reside on only one machine and thus, the only possible separation is at the link between the elementary simulator and the communication channel. The link between an elementary simulator that sends data and the communication channel is done via an "output connector", while the link between the channel and the elementary simulator that receives data will be done via an "input connector".

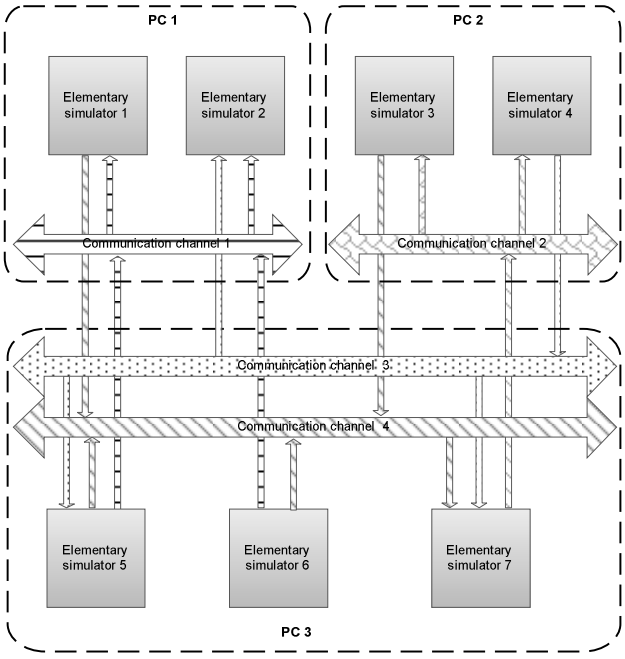


Fig. 2. Block diagram of a distributed simulator. By simply changing the communication channels the seven elementary simulator were placed on different machines. The links are performed using input and output connectors.

By introducing this abstract notion of connector it is easy to implement a system which can be modified only by configuration. In this way a simulator can be implemented to work either on a single machine or on several ones and the switch between implementations can be performed at any time without changes in the source code.

II. ARCHITECTURE OF THE COMMUNICATION CHANNELS

A communication channel is in essence a common medium which can be used by the elementary simulators, which are linked to it via the connectors, to exchange data. From the channel point of view it receives data form one type of connectors and is sends data to a different type of connectors. Those being said it is obvious that the communication channel and its connectors follow the producer/consumer paradigm. They guarantee that the received data is sent to all interested components without error and in exactly the same order as it was sent.

Each communication channel comes with a specific set of connectors which can be linked to the elementary simulators. Because it was a requirement to be able to change the channel without changing the source code of the elementary simulators, it was necessary to impose some rules to those connectors. All input connectors and output connectors implement a common interface which is exposed to the elementary simulators. The simulators will interact with this interface and they are not aware of the actual implementation of the corresponding connector. In this way the changing of the channel can be performed quickly and without intervention at the source code level.

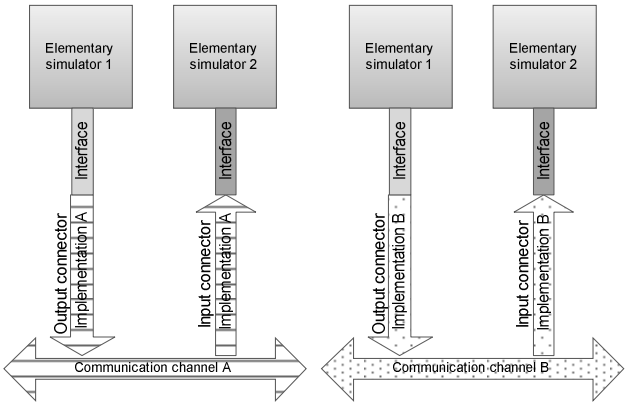


Fig. 3. Two elementary simulators connected via one communication channel A (left) and via a second communication channel B (right). In both situations the input and output connectors respectively use a common interface.

To create a new communication channel one has to extend the class *AbstractCommunicationChanel* or to implement the interface *CommunicationChanel*, and also to create an input and an output connector for it by extending the class *AbstractConnector* and implementing the interface *InputConnector* and *OutputConnector* respectively. The connectors will accept the data to send presented in the form of a byte array. The maximum size of the array is not specified and is left to the developer to decide if it is needed.

The current framework provides four implementations for the communication channels:

- Shared memory communication channel; This communication channel can be used to transfer data between entities located on the same machine and is based on a common memory area which can be accessed by the input and output connectors. When a connector wants to write data in the common memory it first checks if the area is empty and it waits if not. After a successful write it marks the memory as full and it will inform the interested components about the new data. All consumers, sequentially, will read the data, and the last one will mark the area as empty. The disadvantage of this channel is that all operations are performed sequentially and this can be costly if many producers and consumers are connected.

- Queue communication channel; This communication channel can be used to transfer data between entities located on the same machine and is based on the Java class BlockingQueue [2] introduced in Java 1.5. For each consumer connected to this channel, a queue is created. When data is sent to the channel via a connector it will be added to all the queues of the consumers. After this each one can read the data at any time. Compared to the previous implementation the advantage here is the possibility of performing read and write operations asynchronously.

- Socket communication channel; This communication

channel can be used to transfer data between entities located on the same or on different machines and is based on Java stream sockets [3]. The communication channel is a stream server while the connectors are stream clients. Whenever a connector is linked to the channel a new socket is created and saved internally. The channel holds a list of sockets connected to the consumers and it creates a thread for each socket connected to a producer. Whenever a message arrives on one of this threads it is sent to all the consumers via the sockets. This channel is very simple and fast for its purpose and it offers the reliability of the socket stream implementation.

- JMS communication channel; This communication channel can be used to transfer data between entities located on the same or on different machines and is based on the ActiveMQ [1] implementation of Java Messaging System (JMS). The communication channel is a JMS broker and it contains only one topic. All connectors that want to receive data are linked to the topic as consumers. A connector that sends data will post a message to the topic. Via the JMS engine the data is sent to all the consumers. This channel offers the advantage of JMS and the possibility to easily connect the simulator with other applications regardless of the location or even of the language in which they are implemented provided that they have JMS compatibility.

III. TESTING PROCEDURE AND RESULTS

A. *Test environment*

To test the communication channels described earlier a test environment with the channel, one producer and one consumer was created. For all channels the data integrity, message order and the transmission time were tested. The data integrity and message order were verified by comparing the information sent with the information received. The transmission time was tested differently for the first two channels and for the last two.

In the first case all entities were placed on the same physical machine and the transmission time was measured by reading the timestamp T_{start} exactly before the producer generated the message and the timestamp T_{stop} exactly after the consumer had received it. The difference $T = T_{stop} - T_{start}$ represents the time needed to transport the message between the producer and the consumer and because both timestamps are relative to the same machine clock, it represents an absolute value. The test machine in this case was a normal workstation with an Intel E8400 Core2Duo CPU, 6MB Cache L2 3.0GHz per core, 4GB of RAM, 7200RpM HDD, Windows XP SP3 and JVM version 1.6.0.20.

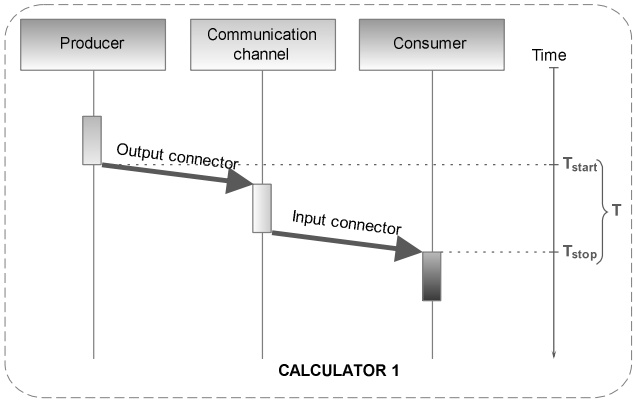


Fig. 4. Test environment for the channels which can transfer data between entities located on the same physical machine.

In the second case the producer was placed on one machine and the consumer on another. In this case the above method does not work because each one of the two timestamps is relative to the clock of the machine where the component resides and if the two clocks could not be perfectly synchronized, an error can be introduced. Because the measurements are in nanoseconds this error can be quite big and the results will be unusable. The solution to this problem was to perform two successive measurements, one with the producer on the first machine and the consumer on the second and the other with the consumer on the first machine and the producer on the second.

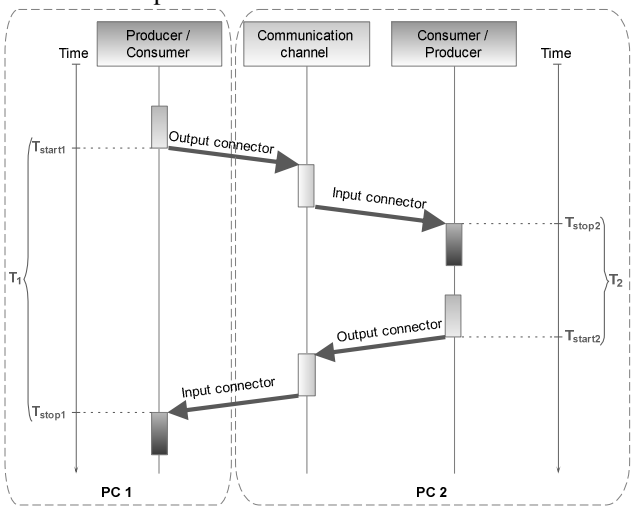


Fig. 5. Test environment for the channels which can transfer data between entities located on different physical machines.

In the first run the timestamp T_{start1} was read on machine 1 just before sending the message and the timestamp T_{stop2} on machine 2 just after receiving it. In the second run the timestamp T_{start2} was read on machine 2 just before sending the message and the timestamp T_{stop1} on machine 1 just after receiving it. The differences $T_1 = T_{stop1} - T_{start1}$ and $T_2 = T_{start2} - T_{stop2}$ represents absolute times because the corresponding timestamps were taken on the same machine. By applying the formula $T = (T_1 - T_2) / 2$ a good

approximation of the time needed to transfer the information via the channel is obtained. The physical test environment in this case was composed of two normal workstations, with the same configuration as in the previous case, interconnected via a cross-over network cable.

For each channel, six tests were performed, the difference between them being the size, in bytes, of the sent data. The six tests correspond to data size of 1, 2, 4, 8, 16 and 32 bytes.

B. Test results and interpretation

The first goal of the tests was to validate the reliability of the communication channels, in other words to check if all packets that were sent were also received in the same order. In the following table the validation results are presented:

As expected the number of data packets sent was equal to the number of data packets received and no differences could be detected between them. As a conclusion, all communication channels are reliable.

TABLE I
DATA VALIDATION FOR THE COMMUNICATION CHANNELS

Channel	Data size[1]	Packets sent	Packets received	Equal packets
Shared memory	1	10000	10000	10000
	2	10000	10000	10000
	4	10000	10000	10000
	8	10000	10000	10000
	16	10000	10000	10000
	32	10000	10000	10000
Queue	1	10000	10000	10000
	2	10000	10000	10000
	4	10000	10000	10000
	8	10000	10000	10000
	16	10000	10000	10000
	32	10000	10000	10000
Socket stream	1	10000	10000	10000
	2	10000	10000	10000
	4	10000	10000	10000
	8	10000	10000	10000
	16	10000	10000	10000
	32	10000	10000	10000
JMS	1	10000	10000	10000
	2	10000	10000	10000
	4	10000	10000	10000
	8	10000	10000	10000
	16	10000	10000	10000
	32	10000	10000	10000

[1]In bytes

The next goal was to compute the time needed to transfer different amounts of data from a producer to a consumer. The maximum value is important because this has an important implication in the functionality of the communication channel. One quick conclusion which can be obtained is that the size of the data has no major influence in the transmission time. This is an unexpected conclusion, but one explication is that most of the time is actually spent on code execution and a small amount is spent on the actual transfer. The code execution time is influenced by the Operating System, the Java Virtual Machine, other applications running in the same time etc. and it can be improved by using a dedicated machine for the system.

TABLE II
TRANSFER TIME STATISTICS

Channel	Data size[1]	Average time[2]	Maximum time[2]	Minimum time[2]
Shared memory	1	10395	114088	3499
	2	11061	130887	3500
	4	11265	247425	3499
	8	10708	203679	3499
	16	11135	254775	3500
	32	11233	215228	3500
Queue	1	12678	226077	3849
	2	12306	301670	3849
	4	12666	203330	3849
	8	12315	226427	3849
	16	12314	294321	3849
	32	12829	230277	3849
Socket stream	1	100150	319357	62994
	2	98993	414708	55294
	4	99323	402459	57744
	8	98678	515848	56344
	16	99974	362564	61944
	32	99831	573943	59144
JMS	1	208481	1144734	96940
	2	219863	1195129	131236
	4	217737	1163983	99390
	8	216352	1142635	99040
	16	228437	976051	124587
	32	216820	1166432	97640

[1]In bytes
[2]In nanoseconds

Another conclusion which can be drawn is that, since the maximum execution time varies from about 115 microseconds (Shared Memory Channel) to 1195 microseconds (JMS Channel), those channels can be used to implement a wide range of simulators for even complex industrial installations because the time delay they introduce is quite low.

Another important observation is that the average values are very low, varying from about 10 microseconds (Shared Memory Channel) to about 228 microseconds (JMS Channel) and, most important, over 88% of the transfer times for each channel and each test are less than the computed average value plus 50%. This leads to the conclusion that the bigger values may have occurred because of the fact that the test machines were not dedicated, but normal workstations.

The graphical representation of the transfer times is not very relevant as the big number of results (10000) make the graphic unreadable. However, by computing the average value of each group of 100 readings and representing the results, conducts to an interesting conclusion: for all channels and all tests the average transfer time of packets 1401 to 1500 is slightly bigger than the previous average (for packets 1301 to 1400) and is followed by a drop at the

next computed average (1501 to 1600). The total average of the packets from this point forward is smaller than the average of the first 1400 packets.

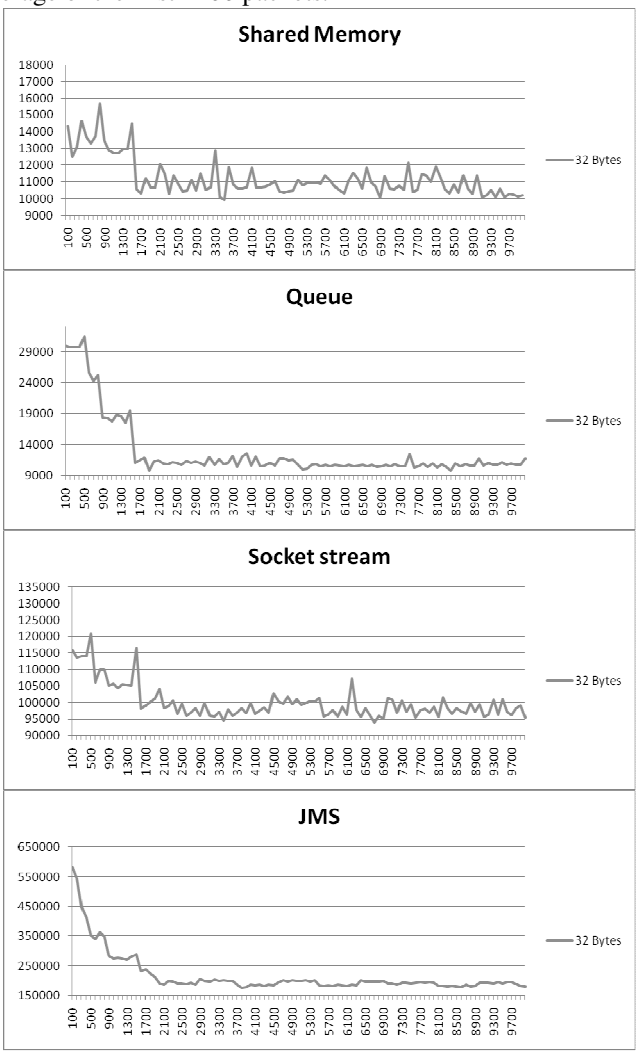


Fig. 6. Graphical representation of the 100 time averages for the Communication channels. To simplify the drawings only the time averages for the 32 Bytes messages were represented. Notice the graph variation in the 1300-1700 intervals.

The single explication here is the JVM Hotspot [2] feature, that is, optimization of the sections of code that are executed repetitively. This means that the longer they run, the better they perform and in this way, if it is necessary, the communication channel can be "wormed" before actually adding it to the system.

IV. FUTURE WORK

The future work for this project is to create a framework that can be used to easily implement the elementary simulators. The final goal is to provide a set of classes which a user can expand and, with a minimum implementation, based only on the necessary mathematical formulas, to create a reusable and reliable elementary simulator for a physical component. The final installation that has to be simulated with this system is a water lock, but the plan is not to limit at this and continue the work in order to obtain a framework capable of simulating almost any industrial installation.

V. CONCLUSION

As a general conclusion this communication framework is fast and reliable and can be used without any problems to interconnect a set of elementary simulators in order to obtain a modular simulator for an industrial installation. All channels performed very well and their execution time decreased over time due to the Java Virtual Machine Hotspot capability.

REFERENCES

[1] B. Snyder, R. Davies and D. Bosanac "ActiveMQ in Action (Unedited draft),", Ed. Manning, ch. 1, submitted for publication.

[2] *J2SE 5.0 Performane White Paper*.[Online] Available: http://www.slideshare.net

[3] Q. H. Mahmoud. (1996, December 11). *Sockets programming in Java: A tutorial* [Online]. Available: http://www.javaworld.com

Robust Control Law Design for a Synchronous Motor Using Feedback Linearization Method

Eugen Bobaşu, *Member, IEEE*, Dan Popescu, *Member, IEEE* and Sergiu Ivanov

***Abstract*— In this paper a combination of the feedback linearizing control technique and Glover-McFarlane control method is applied for the control of a synchronous motor. The use of feedback linearization requires the complete knowledge of the nonlinear system. In practice, there are many processes whose dynamics is very complex, highly nonlinear and usually incompletely known. To improve robustness, it may be necessary to modify the exact linearization controller. First, we apply the method of nonlinear control and state feedback linearization to synchronous motor model and we obtain a nonlinear control law. This law, aggregated with our nonlinear system, achieves input-output linearization and in the case of multivariable approach, the nonlinear control law achieves also decoupling. Then, Glover-McFarlane H_∞ design is used with the goal of increasing robustness of the existing controller. Finally, some simulation results are included to demonstrate the performance of these controllers and the results are compared with the classical dq vector control.**

I. Introduction

A method largely used for the control of nonlinear systems is to calculate a linear controller for the linear approximation of the nonlinear system around an operating point. This kind of control works in a small neighbourhood of the operating point, and when the system is far from this point, the linear controller will not have the desired behaviour. Thus, the feedback linearization is a good technique because the nonlinear system is transformed into a linear system and only then the linear controller is applied. Therefore, a controller associating feedback linearization and linear control is working in any point, not only in a small neighbourhood of the operating point.

The robust feedback linearization is a new form of feedback linearization which gives a linearizing control law that transforms the nonlinear system into its linear approximation around an operating point, causing only a small transformation in the natural behaviour of the system which is desired in order to obtain robustness.

The control of synchronous motors has been widely investigated in a number of works under various points of view (Cerruto et al., 1995; Caravani et al., 1995; Di Gennaro et al., 1994). One of the most frequently used mathematical descriptions is expressed in the (d, q) frame and the coupling between the angular velocity and the electrical quantities results in a bilinear model with the angular velocity as a natural output to be controlled. If model parameters are perfectly known, non-linearities can be canceled by proper selection of state feedback controls via exact feedback linearization.

Several techniques from linear and nonlinear control theory have been applied to the problem of robust feedback linearization: Lyapunov redesign method, sliding modes, the H_∞ approach, Glover-McFarlane H_∞ design, etc.

Our goal is to combine the exact feedback linearization control technique and Glover-McFarlane design method in order to control a synchronous motor with nonlinearities and parameter uncertainties. The method of nonlinear control and state feedback linearization is applied to synchronous motor model and it is obtained a nonlinear control law. This law, together with our nonlinear system, achieves input-output linearization and in the case of multivariable approach, the nonlinear control law achieves also decoupling. Then the Glover-McFarlane H_∞ design is applied, with the goal of increasing robustness of existing controllers without significantly compromising performance.

The paper is organized as follows: in Section II, the nonlinear mathematical model of the synchronous motor is presented together with two linearized models obtained by classical feedback linearization method and, respectively, robust feedback linearization method. In Section III, first, the Glover-McFarlane H_∞ design method is applied in order to robustify the controller obtained by feedback linearization and pole placement method. Then, some simulation results are presented in order to compare the robustness and performances of the designed controllers with those obtained with the classical *dq* vector control. Some concluding remarks are presented in Section IV.

II. Mathematical Models of Synchronous Motor

A. Nonlinear Mathematical Model

The mathematical model of a permanent magnet synchronous motor, which is expressed in the so-called (d,q)-frame, and deduced from the application of the Park transformation, can be written as follow (Caravani et al., 1998):

E. Bobaşu is with the Automatic Control Department, University of Craiova, A.I. Cuza, 200585, Romania (e-mail: ebobasu@automation.ucv.ro).

D. Popescu is with the Automatic Control Department, University of Craiova, A.I. Cuza, 200585, Romania (e-mail: dpopescu@automation.ucv.ro).

S. Ivanov is with the Electrical Drives and Industrial Informatics Department, University of Craiova, A.I. Cuza, 200585, Romania (e-mail: sivanov@em.ucv.ro).

$$\dot{x} = f(x) + \sum_{i=1}^{2} g_i(x)u_i,$$
$$y_j = h_j(x);\ j = 1,2, \tag{1}$$

in which $f(x), g_1(x), g_2(x)$ are smooth vector fields

$$x^T = [i_d, i_q, \omega],\quad u^T = [u_d, u_q]$$
$$f(x) = \begin{bmatrix} -\frac{R}{L} i_d + p\omega i_q \\ -\frac{R}{L} i_q - p\omega i_d - \frac{p\phi\omega}{L} \\ \frac{p\phi}{j} i_q - \frac{f}{j}\omega - \frac{m}{j} \end{bmatrix} \tag{2}$$
$$g_1^T = \left[\frac{1}{L}\,0\,0\right];\ g_2^T = \left[0\,\frac{1}{L}\,0\right]$$
$$y_1 = h_1 = i_d$$
$$y_2 = h_2 = \omega$$

where R is the stator windings resistance, L the inductance, ϕ is the flux of the permanent magnets, i_d, i_q are the currents and u_d, u_q are the applied voltages, and p is the number of pole pairs; ω denotes rotor angular velocity, j is the rotor moment of inertia, f is the viscous friction coefficient and m is the load torque.

In angular velocity control problems typical outputs of interest are the current i_d and the angular velocity ω. In fact, the electromagnetic torque is generated by the i_q component of the current. Therefore, forcing the i_d component to zero tends to align the current vector along the q direction.

This optimizes the use of all the available current for torque producing purposes.

B. Feedback Linearizing Methods

The multivariable nonlinear system we consider is described in state space by equations of the following kind:

$$\dot{x} = f(x) + \sum_{i=1}^{m} g_i(x)u_i$$
$$y_j = h_j(x) \qquad j = 1 \ldots m \tag{3}$$

in which $f(x)$, $g_1(x)$, $g_2(x)$,...., $g_m(x)$ are smooth vector fields.

The problem of exact linearization via feedback and diffeomorphism consists in transforming a nonlinear system (3) into a linear one using a state feedback and a coordinate transformation of the system's state.

We introduce now the Lie derivative of a function $h(x): R^n \to R$ along a vector field $f(x) = [f_1(x), \ldots f_n(x)]^T$

$$L_f h(x) = \sum_{i=1}^{n} \frac{\partial h(x)}{\partial x_i} f_i(x) \tag{4}$$

Definition. A multivariable nonlinear system of the form (3) has a relative degree $\{r_1, \ldots, r_m\}$ at a point x^0 if:

(i) $L_{g_j} L_f^k h_i(x) = 0$ (5)

for all $1 \le j \le m$, for all $1 \le i \le m$ for all $k < r_i - 1$ and for x in a neighbourhood of x^0,

(ii) the $m \times m$ matrix

$$A(x) = \begin{bmatrix} L_{g_1} L_f^{r_1-1} h_2(x) & . & . & L_{g_2} L_f^{r_1-1} h_1(x) \\ L_{g_1} L_f^{r_2-1} h_2(x) & . & . & L_{g_2} L_f^{r_2-1} h_2(x) \\ . & . & . & . \\ L_{g_1} L_f^{r_m-1} h_m(x) & . & . & L_{g_m} L_f^{r_m-1} h_m(x) \end{bmatrix} \tag{6}$$

is nonsingular at $x = x^0$.

Theorem. Let be the nonlinear system of the form (3). Suppose the matrix $g(x^0)$ has rank *m*. Then the State Space Exact Linearization Problem is solvable if and only if

(i) for each $0 \le i \le n-1$, the distribution G_i has constant dimension near x^0;

(ii) the distribution G_{n-1} has dimension n;

(iii) for each $0 \le i \le n-2$, the distribution G_i is involutive.

1) Classical Feedback Linearization: The classical feedback linearization is accomplished by using a linearizing control law of the form $u_c(x, w) = \alpha_c(x) + \beta_c(x)w$, where w is a linear control, and a diffeomorphism $x_c = \phi_c(x)$, with

$$\alpha_c(x) = -A^{-1}(x)\left[L_f^{r_1} h_1(x) \cdots L_f^{r_m} h_m(x)\right]^T$$
$$\beta_c(x) = A^{-1}(x)$$
$$\phi_c^T(x) = \left[\phi_{c_1}^T(x) \cdots \phi_{c_m}^T(x)\right]$$
$$\phi_{c_i}^T(x) = \left[h_i(x)\ L_f h_i(x) \cdots L_f^{r_i-1} h_i(x)\right]$$

The linearized system is:

$$\dot{x}_c = A_c x_c + B_c w \tag{7}$$

where A_c and B_c are the matrices of the Brunovski canonical form.

2) Robust Feedback Linearization: The main difference between the robust feedback linearization and the classical one is that the linearized system has the form

$$\dot{x}_r = A_r x_r + B_r v \tag{8}$$

with $A_r = \partial_x f(0)$ and $B_r = g(0)$, which corresponds to the linear approximation of the nonlinear system (3).

The robust feedback linearization is accomplished by using a linearizing control law of the form $u(x, w) = \alpha(x) + \beta(x)v$, where v is a linear control, and a diffeomorphism $x_r = \phi(x)$, with

$$\alpha(x) = \alpha_c(x) + \beta_c(x)LT^{-1}\phi_c(x)$$
$$\beta(x) = \beta_c(x)R^{-1}$$
$$\phi(x) = T^{-1}\phi_c(x) \quad (9)$$
$$L = -A(0)\partial_c\alpha_c(0)$$
$$T = \partial_x\phi_c(0),\ R = A^{-1}(0)$$

The functions $\alpha(x)$, $\beta(x)$, and $\phi(x)$ satisfy

$$\partial_x\alpha(0) = 0,\ \beta(0) = I, \text{ and } \partial_x\phi(0) = I \quad (10)$$

C. Linearized Models of the Synchronous Motor

1) Classical Feedback Linearized Model: We consider as output variables

$$y_1 = i_d$$
$$y_2 = \omega$$

Easy calculus show that the matrix for mathematical model of the synchronous motor is nonsingular and the relative degree is $\{r_1, r_2\} = \{1, 2\}$.

For the system given by (1), the decoupling matrix is

$$A(x) = \begin{bmatrix} L_{g_1}L_f^0 h_1(x) & L_{g_2}L_f^0 h_1(x) \\ L_{g_1}L_f^1 h_2(x) & L_{g_2}L_f^1 h_2(x) \end{bmatrix} = \begin{bmatrix} \frac{1}{L} & 0 \\ 0 & \frac{p\phi}{Lj} \end{bmatrix} \quad (11)$$

Now, the input-output system can be rewritten in the form:

$$\begin{bmatrix} \dot{y}_1 \\ \ddot{y}_2 \end{bmatrix} = \begin{bmatrix} L_f h_1(x) \\ L_f^2 h_2(x) \end{bmatrix} + A(x)\begin{bmatrix} u_d \\ u_q \end{bmatrix} \quad (12)$$

Because the decoupling matrix (11) is not singular, it is possible to design a nonlinear input

$$\begin{bmatrix} u_d \\ u_q \end{bmatrix} = A^{-1}(x)\begin{bmatrix} -L_f h_1(x) + w_1 \\ -L^2{}_f h_2(x) + w_2 \end{bmatrix} \quad (13)$$

The functions for the classical linearizing feedback control law are

$$\alpha_c(x) = \begin{bmatrix} Ri_d - Lpi_q\omega \\ Ri_q + Lpi_d\omega + p\phi\omega + \frac{Lfi_q}{j} - \frac{Lf^2}{jp\phi} \end{bmatrix},$$

$$\beta_c(x) = \begin{bmatrix} L & 0 \\ 0 & \frac{jL}{p\phi} \end{bmatrix},\ \phi_c(x) = \begin{bmatrix} i_d \\ \omega \\ \frac{p\phi i_q - f\omega}{j} \end{bmatrix}$$

In the new coordinate we have

$$\dot{y}_1 = w_1$$
$$\ddot{y}_2 = w_2 \quad (14)$$

The state feedback (13) transforms this system into a system whose input-output behavior is identical to that of a linear system having transfer function matrix of the form

$$H(s) = \begin{bmatrix} \frac{1}{s} & 0 \\ 0 & \frac{1}{s^2} \end{bmatrix} \quad (15)$$

2) Robust Feedback Linearized Model: The functions for the robust linearizing control law are $\alpha(x)$, $\beta(x)$, and $\phi(x)$ calculated using the functions $\alpha_c(x)$, $\beta_c(x)$, and $\phi_c(x)$ given before and the matrices

$$L = \begin{bmatrix} -\frac{R}{L} & 0 & 0 \\ 0 & -\frac{p\phi R}{jL} - \frac{p\phi f}{j^2} & \frac{f^2}{j^2} - \frac{p^2\phi^2}{jL} \end{bmatrix},$$

$$T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & \frac{p\phi}{j} & -\frac{f}{j} \end{bmatrix},\ R = \begin{bmatrix} L & 0 \\ 0 & \frac{jL}{p\phi} \end{bmatrix}.$$

III. Robust Control Design

Imposing on the system (15) an additional feedback of the form

$$v_1 = -c_{10}(y_1 - y_{1ref}) \quad (16)$$
$$v_2 = -c_{20}(y_2 - y_{2ref}) - c_{21}\dot{y}_2$$

then, the obtained system has a linear input-output behavior, described by the following diagonal transfer function matrix

$$H(s) = \begin{bmatrix} \frac{c_{10}}{s + c_{10}} & 0 \\ 0 & \frac{c_{20}}{s^2 + c_{21}s + c_{20}} \end{bmatrix} \quad (17)$$

A. Glover McFarlane Control Design

We consider the structure of the control system shown in Fig. 1, where are implemented the control laws (16) and K_{r1}, K_{r2} are the robustifying controllers (G_{s1}, G_{s2} are the nominal shaped plants).

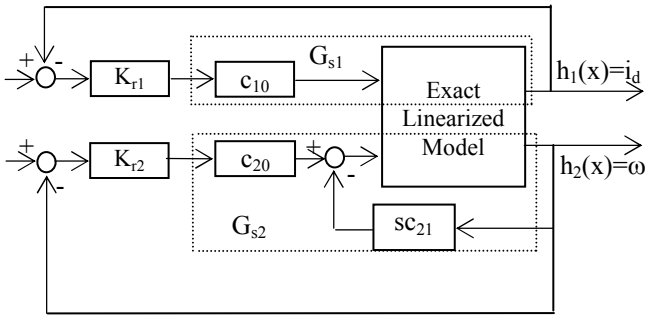


Fig. 1. The control loop.

In this design, the model uncertainties are included as perturbations to the nominal model, and robustness is guaranteed by ensuring that the stability specifications are

satisfied for the *worst-case* uncertainty. Here, since the system is decoupled, we design separately the two robustifying controllers. The method described next [McFarlane and Glover 1992] is applied to the nominal shaped plants G_{s1}, G_{s2}, but the indices are neglected.

Let $G_s = N/M$ be the normalized coprime factorization of the nominal shaped plant.

The normalized coprime factor uncertainty characterization is given by

$$\left\{ \frac{N+\Delta_N}{M+\Delta_M} : \left\| [\Delta_N \Delta_M] \right\| \le \varepsilon \right. \tag{18}$$

The following steps yield the optimal controller that assumes a state-space (A,B,C) available for the transfer function G_s:

1) Obtain Z by solving the algebraic Riccati equation (ARE)

$$AZ + ZA - ZC^T CZ + BB^T = 0 \tag{19}$$

2) Obtain X by solving the ARE

$$AX + XA - XBB^T X + C^T C = 0 \tag{20}$$

3) Compute the maximum possible ε for the given nominal shaped plant

$$\varepsilon_{\max} = (1 + \rho(XZ))^{-1/2} \tag{21}$$

where ρ denotes the spectral radius. Hence, in this design scheme there is no need for an explicit characterization of uncertainty. The method detects and solves for the worst-case scenario.

4) The robustness margin ε is chosen to be slightly less than $\varepsilon_{\max}$. Let $\gamma = 1/\varepsilon$.

5) The state-space realization of the robustifying controller K_r is given by

$$\begin{bmatrix} A + BF + \gamma^2 (L^T)^{-1} ZC^T C & \gamma^2 (L^T)^{-1} ZC^T \\ B^T X & 0 \end{bmatrix} \tag{22}$$

where $F = -B^T X$ and $L = (1-\gamma^2) I + XZ$.

An important feature of this algorithm is that the loop transfer functions, before and after robustification, are not significantly different.

B. Simulation Results

The simulation was performed for the following parameter values of the synchronous motor:

$R = 0{,}6\ \Omega$; $L = 1{,}2 \times 10^{-3}\ H$; $f = 1{,}4 \times 10^{-3}\ Nms$;

$j = 2{,}5 \times 10^{-3}\ Kgm^2$; $\phi = 0{,}12\ Wb$; $p = 4$

We are testing the control performance for step changes in the reference. The simulation was done for the equation model (1) and the nonlinear control law (9), (12). The design parameters are computed using a pole-placement design technique.

For load torque m = 0, y_{1ref} = 0 and a series of step references for y_{2ref} (30 rad/sec at start, 70 rad/sec at 0.5 sec and 90 rad/sec at 1.5 sec), the evolution of variables u_d, u_q, i_d, i_q, ω and ω_{ref} is presented in Fig. 2. It can be seen that the i_d current tends to zero and the angular velocity ω achieves each time the reference after less than 0.1 sec.

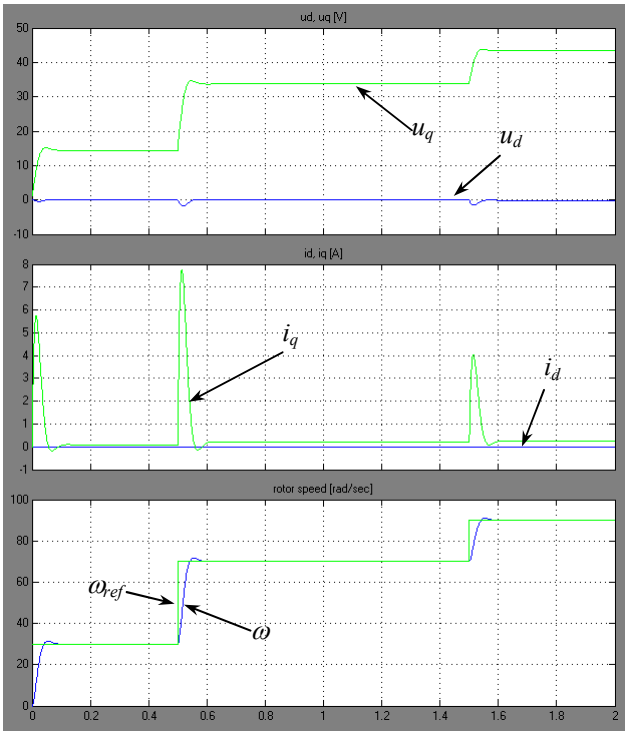


Fig. 2. The results of the simulation for the classical feedback linearized model.

The results of the simulation for the robust feedback linearized model are plotted in Fig. 3.

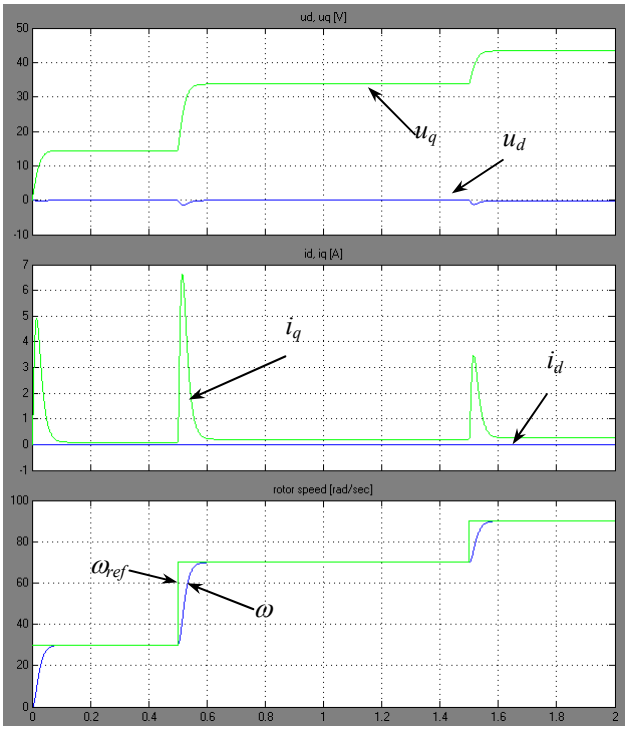


Fig. 3. The results of the simulation for the robust feedback linearized model.

We notice the very good behavior of the control system. The two controlled variables are following the corresponding reference values with high accuracy. The speed has no override thanks to the proper regulation of the active current i_q. At its turn, this current is smoothly controlled by the corresponding voltage component, u_q. In the same delicate manner, the reactive current id is maintained rigorously at its null reference value.

The proposed control strategy is to be compared with the classical *dq* vector control of the PMSM. The most basic such control imposes the *d*-axis component of the current to zero [10-12]. If a current source inverter is used for supplying the motor, the reference values of the two current components are transformed from the Park reference to the fix one, the resulted currents being the reference values for the inverter. When a voltage source inverter is used for supplying the PMSM, the reference values of the voltages result as functions by the necessary currents. If steady state operation is considered $\left(\frac{di_q}{dt}=0\right)$, the two components result:

$$u_{dref} = -pL_q\Omega_m i_{qref},$$

$$u_{qref} = R_a i_{qref} + K_T\Omega_m$$

By using the models previously developed by the authors [8, 9] and imposing the same profile for the reference speed, for the same motor, the resulted *dq* voltages, currents and rotor speed are the one plotted in Fig. 4.

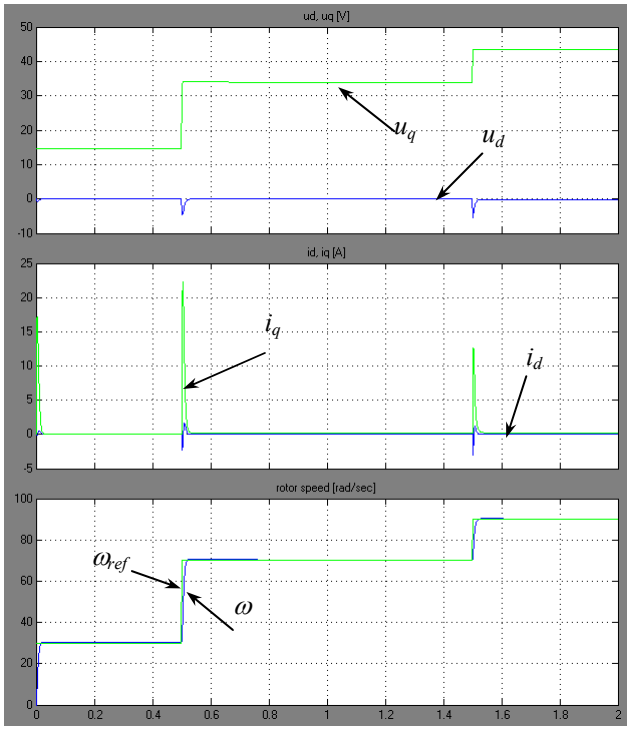


Fig. 4. The results of the simulation for the dq vector control.

In Fig. 5 are plotted the corresponding stator currents.

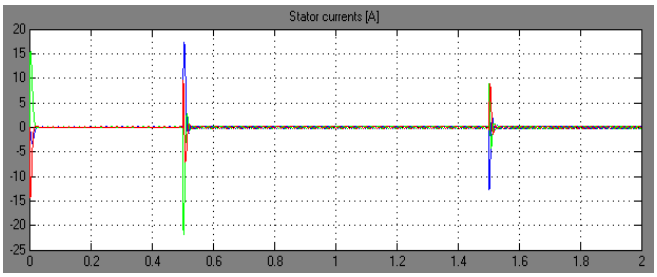


Fig. 5. The stator currents for dq vector control of the PMSM.

From Fig. 4 and 5 one can note the very good dynamical behavior and the quite reduced stator currents during the steady state operation, but the currents i_d and i_q are greater than in the Fig. 2 and 4.

IV. Conclusion

It can be seen that, qualitatively, the results obtained with the proposed control strategy are close by the ones obtained with the classical *dq* control, which confirm the correctness of the proposed strategy. The results can be improved quantitatively by a more proper pole allocation. Further work will be made for testing the robustness properties with respect to variations of some values of the nonlinear model parameters.

References

[1] A. Isidori, *Nonlinear control systems*, 3rd Edition, Springer-Verlag, Berlin, 1995.

[2] A. J. Fossard, D. Normand-Cyrot, *Systemes nonlineaires*, Masson, Paris, 1993.

[3] E. Bobasu, E. Petre, D. Popescu, "On nonlinear control for electric induction motors", In *Process Control'98*, Pardubice, Czech Republic, June Vol. 1, pp. 44-47.

[4] D. McFarlane, K. Glover, "A Loop Shaping Design Procedure Using Hinf Synthesis", *IEEE Trans. Aut. Control*, vol. 37, no. 6, pp.759-769, June 1992.

[5] P. Caravani, S. Di Gennaro, "Robust Control of Synchronous Motors with Non-linearities and Parameter Uncertainties". *Automatica*, vol. 34, no. 4, pp. 445-450, 1998.

[6] E. Cerruto, A. Consoli, A. Raciti, A. Testa, "A robust adaptive controller for PM motor drives in robotic applications". *IEEE Trans. Power Electron*, no. 10, pp. 62-71, 1995.

[7] S. Di Genaro, M. Tursini, "Control techniques for synchronous motor with flexible shaft". *Proc. IEEE Conf. on Control Appl.*, Glasgow, pp. 471-476, 1994.

[8] S. Ivanov, V. Defosse, F. Labrique, P. Sente, "Simulink Library for Simulation of the Permanent Magnets Synchronous Machine Drive", in Proc *22nd European Conference on Modelling and Simulation*, Nicosia, Cyprus, 2008 pp. 363-369.

[9] S. Ivanov, F. Labrique, V. Defosse, P. Sente, "Models of the Permanent Magnets Synchronous Machine and Controllers for Simulation of the Faults", *Electromotion Jurnal*, ISSN 1223 - 057X, Nr. 3, Vol. 15, July 2008, pp. 113-120.

[10] Kaswierkowski and al., "Novel space vector based current controller for PWM inverters", *IEEE Transactions on Power Electronics*, vol. 6, no. 1, January 1991.

[11] P. Vas, *Sensorless Vector and Direct Torque Control*, Clarendon Press, Oxford, 1998.

[12] P. Vas, *Vector control of AC Machines*, Clarendon Press, Oxford, 1990.

Position Control of a Non Conventional Hyper Redundant Arm

G. Boccolato, I. Dinulescu, A. Predescu and D. Cojocaru

Abstract — **This paper deals with the subject of a hyper-redundant robot control simulation. This type of arm changes its configuration by bending a continuous backbone formed of sections connected in a serial configuration. Theoretically the arm can achieve any position and orientation in the working space. Few prototypes were implemented in the last 25 years. The last arm prototype is like a cone one. It was designed and the practical realization is now running. The goal of this paper it's to establish the constructive way to build the visual algorithm to control the system. One limitation introduced about the visual serving system is that it does not answer in real time. The delay introduced by the image-processing task is taken into consideration and simulations are performed to design Closed-Loop control that stabilizes the system.**

The research focused on hyper redundant robots to the Mechatronics Dept., Faculty of Automation, Computers and Electronics, University of Craiova, Romania, is supported by the Romanian National University Research Council CNCSIS under ID_93 IDEI grant.
Alice Predescu is Ph.D. student, predescualice@gmail.com
Giuseppe Boccolato is an Early Stage Researcher supported by the European Commission under Marie Curie Program Network, FreeSubnet project, code MRTN-CT-2006-036186 gboccolato@tin.it
Ionut Dinulescu is Ph.D. student, ionutdinulescu@gmail.com
Dorian Cojocaru is full professor cojocaru@robotics.ucv.ro

I. PREFACE

The hyper-redundant robotic structures, also identified as tentacular robots or continuum robots, represent a robotic class of great interest and also of great complexity. The mathematical modeling of such a robot takes under consideration the kinematics approach. The direct kinematics models are based on the concept of curved segment and the concatenation of these segments to form a continuum multi-segment [1], [2], [8]. The literature on this argument is wide. We found interesting: [3], [6], [7] and [11]. Various solutions have been tried in order to obtain a fast and accurate control algorithm.

Since 2008, the research group from the University of Craiova, Romania designed a new experimental platform for tentacle manipulators. First, a cylindrical structured was designed and experimented. A new prototype based on truncated cone segments was designed and implemented.

The version built now, uses stepper motors that actuate the cables which, by correlated screwing and unscrewing of their ends, determines their shortening or prolonging, changing the backbone's curvature. The backbone of the tentacle is an elastic rod made of steel, which sustains the entire structure.

Depending on which cable shortens or prolongs, the tentacle bends in different planes, each one making different angles (rotations) respective to the initial coordinates frame attached to the manipulator segment – i.e. allowing the movement in 3D.

The control of hyper-redundant robots is a complex problem, and known solutions are based on visual servoing control. A specialized module was developed in order to fulfill this task. To simulate the closed loop controlled system, the visual acquisition system it is rapresented as a delay.

A clustering algorithm was developed and adapted for the particular case of hyperredundant robots. This algorithm identifies the bases of each robot segment as a cluster and computes their centers.

The visual servoing control system is based on binocular vision. The continuous measurement of the arm's parameters derived from the real time computation of the binocular optic flow of the two images has been compared to the desired arm position [9], [10].

The error control function has been built in the 3D Cartesian space using the video information obtained from the two cameras after a fixed time. The 2D errors obtained in the planes of the two images have been determined by the differences between the existing values and the ones desired for the angles that define the arms projections.

Having non real-time system, the measures will not be taken in the time we want. To correct the possible times errors the algorithm uses two different threads, one for the acquisition and one for the picture analysis. Using a circular buffer we are able to measure the position and the speed in the operation space reconstructing the frames spaced in the time equally.

The error function is computed in the virtual space of the images and calibration is not necessary thus providing the opportunity to synthesize a more robust control law.

Figure 1. The structure reproducing a tentacle shape.

II. TENTACLE ARM KINEMATICS

For the 3D case, a virtual wire is considered, which gives the direction of the curvature (α).

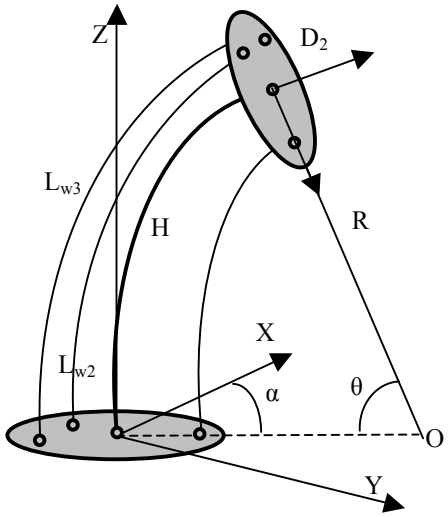


Figure 2. The geometry of one segment.

The direction of the virtual wire must coincide with the direction of the desired curvature. The lengths in (1) are computed from Fig 2.:

$$\begin{aligned} L_{11} &= R + \frac{D_1}{2}\cdot\cos(\alpha_1) & L_{12} &= R + \frac{D_2}{2}\cdot\cos(\alpha_1) \\ L_{21} &= R + \frac{D_1}{2}\cdot\cos(\alpha_2) & L_{22} &= R + \frac{D_2}{2}\cdot\cos(\alpha_2) \\ L_{31} &= R + \frac{D_1}{2}\cdot\cos(\alpha_3) & L_{32} &= R + \frac{D_2}{2}\cdot\cos(\alpha_3) \end{aligned} \quad (1)$$

where $\alpha_1 = -\alpha$, $\alpha_2 = 120° - \alpha$, $\alpha_3 = 240° - \alpha$, D_1 and D_2 are the diameters of the discs.

Based on the relation (1) the curvature radia R_1, R_2 and R_3 of the three control wires are then obtained according to the relation (2).

$$\begin{aligned} R_1 &= \frac{\sqrt{L_{11}^2 + L_{12}^2 - 2\cdot L_{11}^2\cdot L_{12}^2\cdot\cos\theta}}{2\cdot\sin\frac{\theta}{2}} \\ R_2 &= \frac{\sqrt{L_{21}^2 + L_{22}^2 - 2\cdot L_{21}^2\cdot L_{22}^2\cdot\cos\theta}}{2\cdot\sin\frac{\theta}{2}} \\ R_3 &= \frac{\sqrt{L_{31}^2 + L_{32}^2 - 2\cdot L_{31}^2\cdot L_{32}^2\cdot\cos\theta}}{2\cdot\sin\frac{\theta}{2}} \end{aligned} \quad (2)$$

In depth information on how relations (2) are obtained can be found in [2].

Finally, the lengths of the control wires are computed with the relation (3):

$$\begin{aligned} L_{w1} &= R_1\cdot\theta \\ L_{w2} &= R_2\cdot\theta \\ L_{w3} &= R_3\cdot\theta \end{aligned} \quad (3)$$

The desired shape of the tentacle arm is obtained by controlling the θ and α angles of the three segments. For each of the segments we impose the interval time to go from the current shape to the desired shape of the individual segments. This approach allows the control wires to move with constant velocities, easily calculated with relation (4):

$$v_i = \frac{L_{wi}^f - L_{wi}^i}{T_f}, \quad (4)$$

where L_{wi}^f and L_{wi}^i are the final and initial lengths of the control wires for the i-th segment and T_f is the imposed interval time. The position of the segment end-point is calculated based on the values of the θ and α angles.

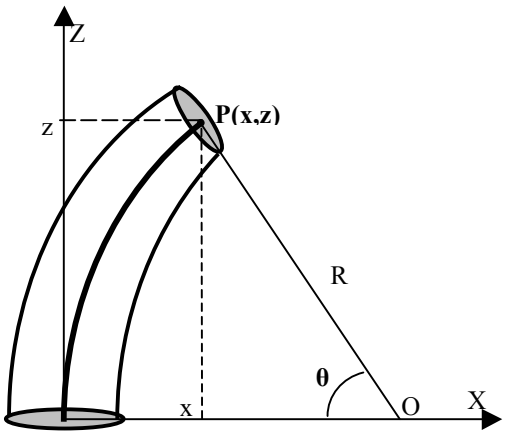


Figure 3. The position of the segment's end-point.

For the 2D case, the coordinates of the terminal point are calculated with the following relations, obtained using the geometrical representation in Fig. 3:

$$\begin{aligned} x &= R - R\cos(\theta) \\ z &= R\sin(\theta) \end{aligned} \quad (5)$$

where R=H/θ and H is the height of the segment and θ is the curvature angle.

The 3D position of the terminal point takes in consideration also the direction angle α. The position is calculated from the relation (5) by rotating the end point with angle α against the z axis. Relation (6) shows how the 3D position is obtained:

$$\begin{aligned} x &= [R - R\cos(\theta)]\cdot\cos(\alpha) \\ y &= [R - R\cos(\theta)]\cdot\sin(\alpha) \\ z &= R\sin(\theta) \end{aligned} \quad (6)$$

III. DIFFERENTIAL KINEMATICS

The position control system is used for controlling the (x,z) position of the arm's endpoint. For the 2D case, the relations (5) show that the x and z relations are dependent, therefore it is enough to specify only the coordinate x for the reference position. The z coordinate of the reference position will be calculated using the relation (7).

$$z = R\sin(\arccos\ (1-\frac{x}{R})) \qquad (7)$$

From relation (7), the x coordinate is expressed as a function of the curvature angle θ, as in (8).

$$x = f(\theta) = R(1-\cos\ \theta) \qquad (8)$$

The velocity of the x-cordinate is easily computed by the derivation of (8) in respect to the curvature angle θ. The differential model for one segment is then obtained with the relation (9).

$$\partial x = \frac{\partial f(\theta)}{\partial \theta} = J(\theta)\cdot\partial\theta \qquad (9)$$

where x is the x-axis coordinate, θ is the curvature angle and J is the Jacobian, calculated as in (10) using relations (7) and (9):

$$J(\theta) = \frac{\partial f(\theta)}{\partial \theta} = -\frac{H\cdot(1-\cos(\theta))}{\theta} + \frac{H\cdot\sin(\theta)}{\theta} \qquad (10)$$

In the 3D case, the position of the segment's end point depends both on the curvature angle θ and direction angle α. The relation between the position velocity and the internal parameters velocities is expressed as in the relation (11):

$$\partial\begin{bmatrix} x \\ y \\ z \end{bmatrix} = J\cdot\partial\begin{bmatrix} \theta \\ \alpha \end{bmatrix} \qquad (11)$$

The Jacobian J is the matrix calculated in relation (12) below.

$$J = \begin{bmatrix} \frac{\partial x}{\partial\theta} & \frac{\partial x}{\partial\alpha} \\ \frac{\partial y}{\partial\theta} & \frac{\partial y}{\partial\alpha} \\ \frac{\partial z}{\partial\theta} & \frac{\partial z}{\partial\alpha} \end{bmatrix} \qquad (12)$$

The elements of the 3D Jacobian are obtained by derivation of the relations (6) in respect to the appropiate angle.

IV. IMAGE BASED SYSTEM CONTROL

Two video cameras provide two images of the whole robot workspace. The two image planes are parallel with XOY and ZOY planes from robot coordinate frame, respectively [5]. The cameras provide the images of the scene that are stored in the frame grabber's video memory. Respective to the image planes are defined two dimensional coordinate frames, called screen coordinate frames or image coordinate systems. Denote $X_{S_1}Y_{S_1}$ and $Z_{S2}Y_{S_2}$, respectively, the axes of the two screen coordinate frames provided by the two cameras. The spatial centers for each camera are located at distances D1 and D2, with respect to the XOY and ZOY planes, respectively. The orientation of the cameras around the optical axes with respect to the robot coordinate frame, are noted by Ψ and Φ, respectively (Fig. 4).

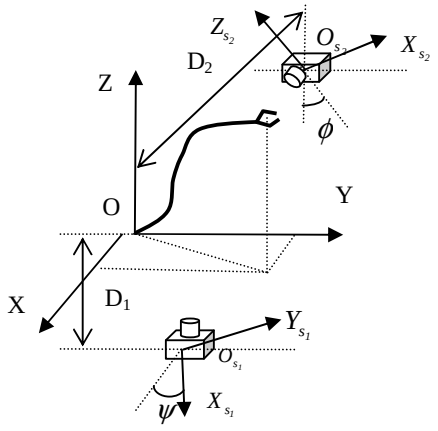


Figure 4. The robot coordinate system.

The control system is an image – based visual servo control where the error control signal is defined directly in terms of image feature parameters. The desired position of the arm in the robot space is defined by the curve C_d, or, in the two image coordinate frames $Z_{S_1}O_{S_1}Y_{S_1}$ and $Z_{S_2}O_{S_2}Y_{S_2}$, by the projection of the curve C in the image coordinates frame (Fig. 5).

The control problem of this system is a direct visual servo-control, but the classical concept of the position control, in which the error between the robot end-effector and target is minimized, is not used. In this application the control of the shape of the curve in each point of the mechanical structure is used. The method is based on the particular structure of the system defined as a "backbone with two continuous angles $\theta(s)$ and q(s)". The control of the system is based on the control of the two angles $\theta(s)$ and q(s) (Fig. 6).

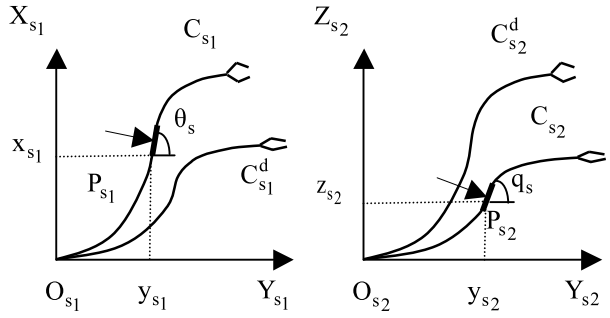


Figure 5. The camera coordinate system.

These angles are measured directly or indirectly. The angle $\theta(s)$ is measured directly by the projection on the image plane $Z_{S_1}O_{S_1}Y_{S_1}$ and q(s) is computed from the projection on the image plane $Z_{S_2}O_{S_2}Y_{S_2}$.

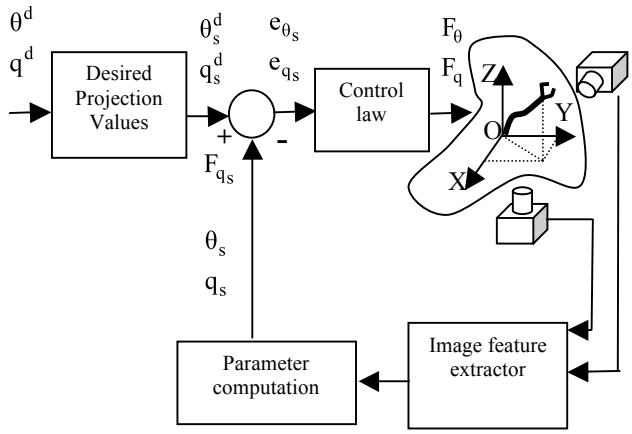


Figure 6. The global control system.

In order to implement the visual-servoing system, a benchmark was organized based on two color camera with 0.05 lux low light sensitivity and the DT3160 family frame grabber from Data Translation.

The cameras have motorized Pant/Tilt/Zoom and are mounted in perpendicular planes offering the input for the frame grabber. The Pant/Tilt/Zoom precision is sufficient for this step of the application development. Two white screens are placed in front of the cameras in order to increase the images contrast. The tentacle arm is placed between each camera and its background screen.

The image processing tasks are performed using Global LAB Image2 from Data Translation. The robot control algorithms are implemented in a C++ program running on a Pentium IV PC. In order to facilitate the image feature extraction, a set of markers are placed on joints along the backbone structure.

A very important task in developing this application is to control the camera position and orientation. From this point of view, the calibration operation assures that the two cameras' axes are orthogonal. The calibration for a pan/tilt/zoom camera orientation is achieved by means of an engineered environment and a simulation graphic module [4]. In the beginning, the tentacle manipulator receives the needed commands in order to stand in a test pose (imposed position and orientation). The term "camera calibration" in the context of this paper refers only to positioning and orienting the two cameras at imposed values.

V. Closed Loop Control

The control wires are actuated by stepper motors which allow open-loop control of the robot's shape. However, the stepper motors may loose steps, which results in large positioning errors. In order to compensate the errors, the control loop has been closed with an artificial vision system which measures the current curvature angles of the segments.

The differential kinematics obtained in the relation (10) is used by the control law in order to get the reference position. The closed-loop control system is shown in Fig.7.

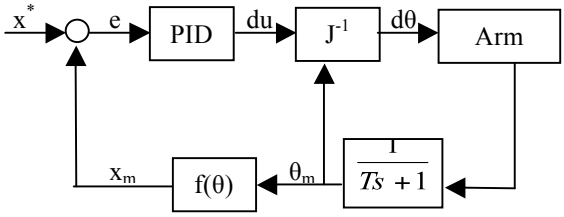


Figure 7. The differential control system.

The video camera is modelled as a first-order system, as in relation (13), where T is the time constant of the camera system:

$$VC(s) = \frac{1}{Ts + 1} \tag{13}$$

The video acquisition system measures the curvature angle θ_m, based on which the measured value of the end point's x coordinate is then obtained, using relation (8). The error calculated as the difference between the reference position x^* and the measured value x_m is the input of the PID controller, which computes the command du.

Using relation (13), the $d\theta$ command value is computed and transmitted to the arm's control system.

VI. Simulation and Experiments

A MatLab simulation software has been created in order to facilitate experiments (Fig.8). The simulator allows full configuration of the mechanical parameters of the robotic arm, which include the diameters of the discs, and the distances between them.

The initial and final shapes of the tentacle arm are specified by setting the corresponding values for the α and θ angles for each of the three segments. Then, the software simulates the shape transformation of the arm using the control approach proposed in the previous sections of this paper.

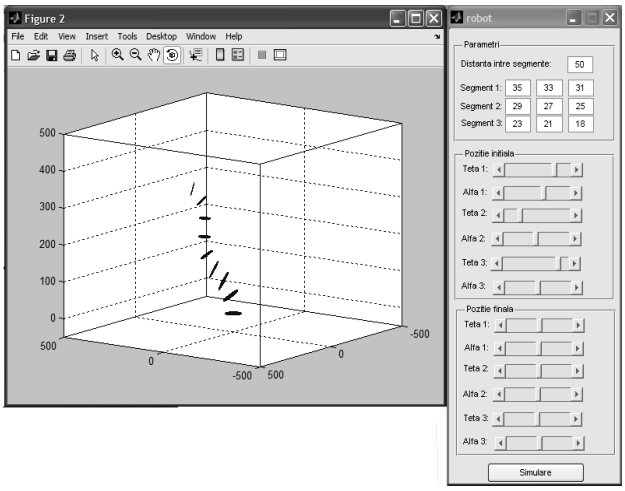


Figure 8. The MatLab simulator of the tentacle arm.

Fig. 9 shows the evolution simulation of one arm segment when its corresponding curvature angle changes in time. The end point trajectory is also drawn by the simulator.

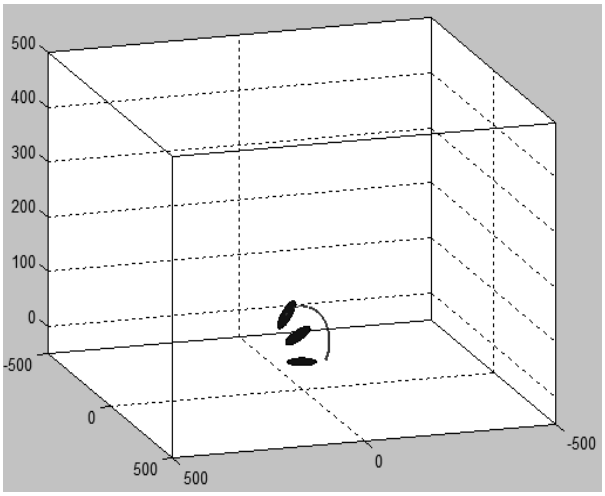


Figure 9. Trajectory simulation of the tentacle arm.

Fig. 10 shows the length variation of the control segments for a 2D motion ($\alpha = 0$), where θ goes from $-\pi/4$ to $\pi/4$. It can be remarked that the plots intersect when $\theta=0$, which corresponds to the position where the lengths of the three control segments are equal.

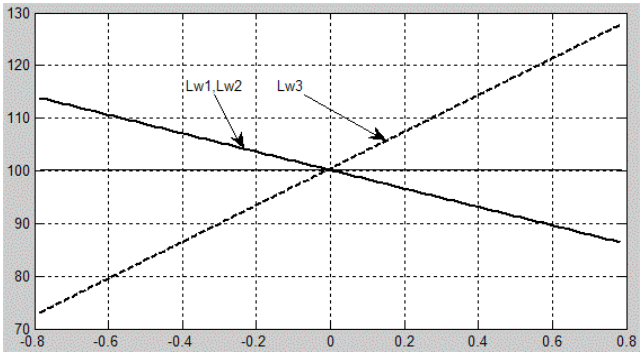


Figure 10. The evolution of the control wires lenghts for the 2D case.

The closed-loop control system has been implemented using the Simulink environment. The Simulink model is presented in Fig. 11.

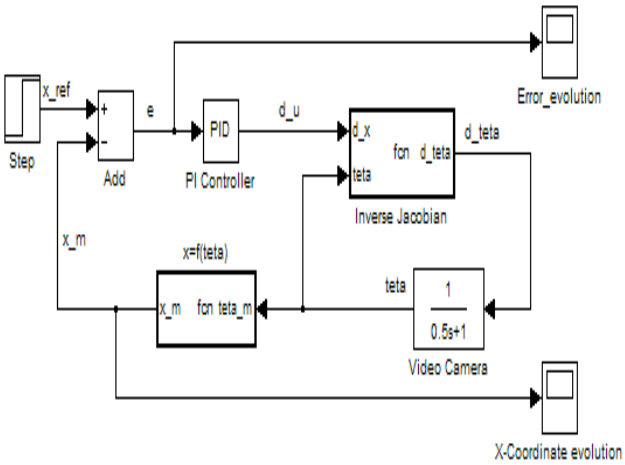


Figure 11. The Simulink model of the control system.

The Simulink model allowed the tunning of the PID controller and evaluation of the control system performance. Fig. 12 shows the evolution of the curvature angle and of the system's error as a response to a one-unit step input.

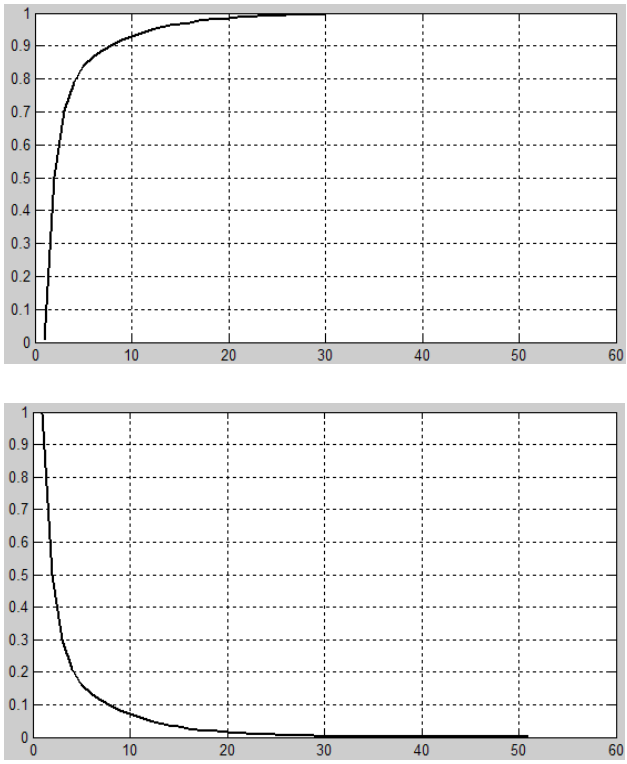


Figure 12. Evolution of the a) system output, b) system error.

Fig. 13 shows the variations of the x and z coordinates when θ goes from $-\pi$ to π, while α is constant. From this plot results the circular shape of the segment's work envelope that gives its maximum reachability in the operation space.

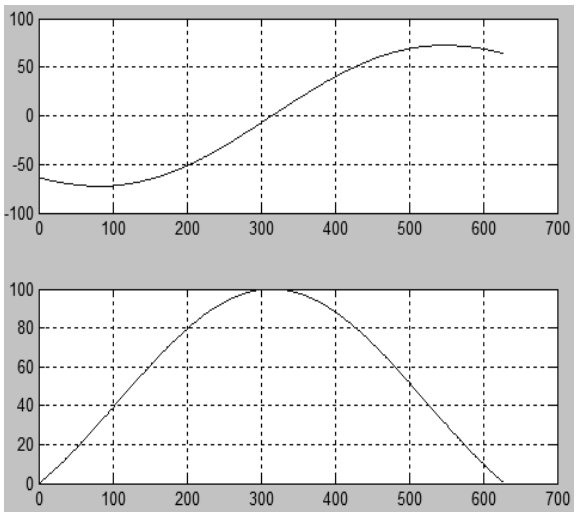


Figure 13. Variation of the x and z coordinates for the end-point.

VII. CONCLUSIONS AND FUTURE WORK

The kinematics model of the hyper-redundant arm, along with simulation results have been presented in this paper. Also a vision based control system has been proposed. The control system uses two external cameras to capture the current shape of the robotic arm. The controller uses a traditional PID control law and the differential kinematics of the arm. A MatLab simulator has been designed in order to test the kinematics of the robot and closed-loop control system.

The next step in our research involves the implementation of the controller on the real robot.

TABLE 1
REFERENCE DISCS CENTERS COORDINATES COMPUTED IN IMAGE COORDINATE SYSTEM

	Disc 1			Disc 2		
	X	Y	Z	X	Y	Z
Initial position	212	212	41	212	212	164
Intermediate	212	212	41	210	213	166
Final position	212	212	41	207	226	170
	Disc 3			Disc 4		
	X	Y	Z	X	Y	Z
Initial position	212	212	334	212	212	502
Intermediate p	190	217	330	186	233	491
Final position	177	243	331	134	268	478

TABLE 2.
REFERENCE DISCS CENTERS COORDINATES ARE ALSO COMPUTED IN VIRTUAL CONTROL SPACE (COMPATIBLE WITH THE GRAPHIC ENVIRONMENT COORDINATES SYSTEM AND TRANSFORMED USING THE DISTANCE TO THE CAMERAS.

	Disc 1			Disc 2		
	X	Y	Z	X	Y	Z
Initial position	0	0	0	0	0	100
Intermediate	0	0	0	-1.63	0.81	101.63
Final position	0	0	0	-4.07	11.38	104.88
	Disc 3			Disc 4		

	X	Y	Z	X	Y	Z
Initial position	0	0	200	0	0	300
Intermediate	-15.02	3.41	197.27	-16.92	13.67	292.84
Final position	-23.89	21.16	197.95	-50.76	36.44	284.38

TABLE 3.
DISTANCE TO THE TARGET EVOLUTION IN TIME FOR EACH REFERENCE DISC.

	Disc 2	Disc 3	Disc 4
Initial position	12,98	31,58	64,60
Intermediate position	11,32	19,54	41,86
Final position	0,40	0,42	0,27

Acknowledgement: The research presented in this paper was supported by the Romanian National University Research Council CNCSIS through the IDEI Research Grant ID93 and by FP6 MARTN through FREESUBNET Project no. 36186.

REFERENCES

[1] Blessing, M. and Walker, I.D. (2004). Novel Continuum Robots with Variable Length Sections, Proc. 3rd IFAC Symp. on Mechatronic Syst. Sydney, Australia, pp. 55-60.

[2] Boccolato, G., Dinulescu,I., Predescu, A., Manta, F., Dumitru, S., Cojocaru, D. (2010). 3D Control for a Tronconic Tentacle, IEEE 12th Conference on Computer Modelling and Simulation, pp.380-385.

[3] Chirikjian, G. S. and Burdick, J.W. (1995). Kinematically Optimal Hyper-redundant Manipulator Configurations, IEEE Transaction on Rob. and Autom., vol. 11, no. 6, pp. 794-798.

[4] Cieslak, R. and Morecki, A. (1999). Elephant Trunk Type Elastic Manipultor, Robotica, Vol. 17, pp. 11-16.

[5] Cojocaru, D., Tanasie, T.R., (2008). Calibration Application for Real-Time Hyperredundant Robot's Vision Control, Proceedings of the RAAD 2008 17th International Workshop on Robotics in Alpe-Adria-Danube Region, ISBN 978-88-9037-0-8, paper no. 39, September 15-17, 2008, Ancona, Italy.

[6] Cojocaru, D., Ivanescu, M., Tanasie, R.T., Dumitru, S., Manta, F., (2010). Vision Control for Hyperredundant Robots, International Journal Automation Austria (IJAA), ISSN 1562-2703, IFAC-Beirat Österreich, Vol. 1, 18 (2010), p52-66.

[7] Gravagne, I.A. and Walker, I.D.(2000). On the Kinematics of Remotely – Actuated Continuum Robots, Proc. 2000 IEEE International Conference on Robotics and Automation, San Francisco, pp. 2544-2550.

[8] Hirose, S. (1993). Biologically Inspired Robots, Oxford University Press.

[9] Immega, G. and Antonelli, K. (1995). The KSI Tentacle Manipulator, Proc. 1995 IEEE Conference on Robotics and Automation, pp. 3149-3154.

[10] Ivanescu, M., Cojocaru, D., a.o., 2007, Visual Servoing by Artificial Potential Methods of a Hyperredundant Robot, Proceedings of 16th International Workshop on Robotics in Alpe-Adria-Danube Region - RAAD 2007, ISBN 978-961-243-066-5, p254-261, June 7-9, Ljubljana, Slovenia.

[11] Tanasie, R., Ivănescu, M., Cojocaru, D., 2009, Visual Servoing Camera Orienting and Positioning Algorithm for Tentacular Robot Control, RAAD 2009, 18th International Workshop on Robotics in Alpe-Adria-Danube Region, ISBN 978-606-521-315-9, paper ID 027, May 25-27, Brasov, Romania.

[12] Walker, I.D. and Carreras, C. (2006). Extension versus Bending for Continuum Robots, International Journal of Advanced Robotic Systems, Vol. 3, No.2, 2006, ISSN 1729-8806, pp. 171-178.

Price-based, distributed control in power systems

P.P.J. van den Bosch, A. Jokic, R.M. Hermans, M. Lazar [1]

Abstract. Present arrangements and regulation for ancillary services for power balance in power systems cannot cope with future developments in power systems as the participants do not receive proper incentives for required behaviour. This paper analyzes the consequences of current arrangements for system operation and stability when all participants make their own trade-off between risks and economic operation. Two-sided markets for ancillary services are proposed to replace the single-sided market of present secondary control arrangements and a real-time, price-based, distributed control strategy for power imbalances is described which can replace primary control. With only local information of a time-varying (nodal) price for energy each participant makes its own economical decisions. The distributed control scheme enforces that in steady state the global economic solution will be achieved satisfying all network and participant constraints and global objectives. In contrast with present arrangements for primary control, all participants receive proper incentives such that economically rational behavior supports the global requirements on stability, reliability and minimal costs. The new proposal guarantees a reliable and efficient operation of power systems in a market environment with responsive, reliable and accountable but also competing prosumers, a large penetration of renewables and continent spanning transmission networks.

KEY WORDS: Distributed control, price-based control, ancillary services; nodal prices, power balance; primary control; secondary control

1. INTRODUCTION

We assume that there are transparent and open markets for day-ahead trading of energy based on predictions of available power sources and demand. These markets are based on Balance Responsible Parties (BRP) which are the only entities that are allowed and capable to trade on these markets. Based on their bids on these (future) energy markets, they decide about the amount of energy they will sell/buy on these markets to create an energy balance among their own production, demand and net energy bought/sold from the markets, in all time periods of some future time interval (e.g. the next day). The System Operator (TSO) provides them with incentives which ensure that BRPs will maximize their profits by reducing their risks for having an energy imbalance. The market will decide about how much net energy has to be delivered/received from other BRPs and for which price. These considerations are based on predicted amounts of energy and prices. Uncertainty and disturbances are explicitly not taken into account. Bilateral contracts, although maybe less optimal than selling or buying on the Power eXchange (PX) market, are still attractive to reduce their risks due to volatile and not predictable prices at the PX.

P.P.J. van den Bosch, A. Jokic, R.M. Hermans and M. Lazar are with Electrical Engineering Faculty, Eindhoven University of Technology, The Netherlands, *P.P.J.v.d.Bosch@tue.nl*

Paper [1] discusses what has to be done, from a systems point of view, to guarantee a reliable and an economic operation of the power system. It focuses on arrangements, markets and required incentives to deal with Ancillary Services (AS) for power imbalance which are intended for and can cope with uncertainties and unexpected disturbances [2-6]. It was shown that the present way of dealing with uncertainty and disturbances is neither consistent, nor optimal and not well suited for the challenges of the future [2,8]. In [1] a proposal has been made about market-based solutions to achieve that goal, namely a two-sided ahead market and price-based control for ancillary services. This paper continues that discussion with a focus on the trade-offs of the participants (BRP's and TSO) in such a market environment between risks and economic operation. It proposes consistent and incentives-driven alternatives for the presently used primary and secondary control arrangements. Moreover, it shows that solutions exist for real-time, distributed, price-based control of power imbalances, even in the presence of network constraints and nodal pricing.

Notation: We assume that *power [MW]* and *energy [MWh]* can be both positive (production) and negative (consumption), *prosumer*: end-user who can produce (producer) or consume (consumer) electric energy, *prosumption*: production or consumption.

2. PRESENT ARRANGEMENTS

A BRP is a reliable, accountable partner in the daily operation of power markets. It has to and is able to represent its own production capacities and demands but also the prosumption of its prosumers (producers/consumers) which are represented by their BRP on the markets. In the Netherlands there is an open market for energy with a market share > 20%. At the APX (Amsterdam Power eXchange) all BRP's can trade and take care of their own energy balance (production + demand + net import = 0). Together with long-lasting and short bilateral contracts and traded energy at the APX (and associated prices) they shape the E-program for the next day. All BRPs have to satisfy their *energy commitments in each program time unit (PTU)*, else the unknown real-time price of imbalance power has to be paid. As stated earlier, this trade is based on the amount of energy

in a PTU of length T, e.g. 0.25 or 1 [h]. The power is measured each 4 seconds and integrated over T [h] to calculate the amount of exchanged energy. A BRP has a responsibility to realize the agreed energy in each PTU, but NO responsibility to keep a *power balance* (Fig. 1).

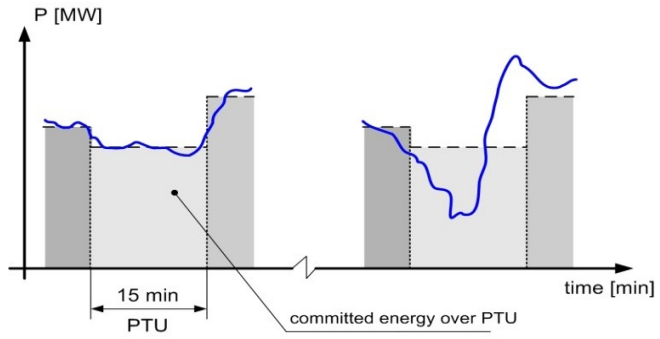


Figure 1 Power profiles in a PTU. Both satisfy the same energy requirement.

In real-time the predicted values will deviate from their real values. In a grid without control at system level any load imbalance ΔP [MW] will introduce a constant frequency deviation $\Delta P/c_{nw}$ [Hz] from the nominal frequency f_o (50 [Hz]), with c_{nw} [MW/Hz] the network constant owing to frequency-dependent prosumption. The larger the equivalent inertia J [kgm^2] and the network constant c_{nw} in the network, the better the disturbance is counteracted. Control is the ultimate tool to cope with unpredictability. Control requires signals of which both a reference and a measured value are known. Inside a synchronous (AC) power system, the global available frequency *f* and the local power flows are relevant signals to track the power balance. Primary (PC) and Secondary control (SC) are based on these signals. Each BRP and most likely several of its controllable power sources (+/-) will locally measure the frequency *f* [Hz], detect any deviation Δf [Hz] from the nominal frequency f_o [Hz] and adjusts its power accordingly with a proportional control law:

$\Delta P = c_{pci} \cdot \Delta f$ [MW].

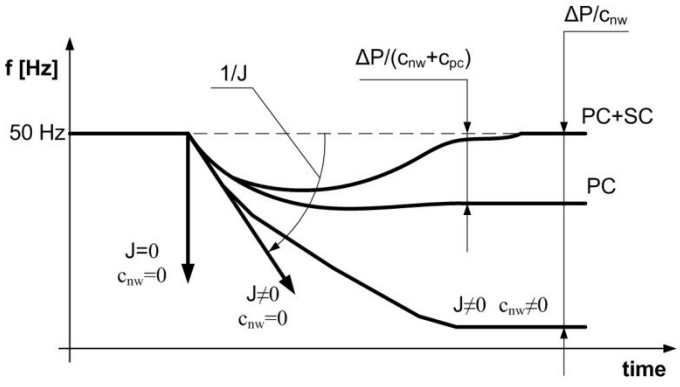


Figure 2, Consequence of load disturbance ΔP [MW] on frequency f depending on values J, c_{nw} and without/with PC and/or SC control active

The power grid is divided into several control areas, coinciding with individual countries. The TSO in a control area measures the cross-border power and energy exchange. Based on this error in the power exchange (ΔP) and possible frequency deviation (Δf) the area control error (ACE) $\Delta P + c_{sc} \Delta f$ [MW] is calculated, where c_{sc} is a system constant of the area. SC has to reduce this error ACE to zero. The TSO utilizes an I-controller, which output P_{sc} [MW] indicates how the power set points in the control area have to be changed. Fig.2 elucidates these effects.

In [1] drawbacks of these present arrangements are discussed, with the following conclusions:

- BRPs have to satisfy their negotiated energy within a PTU. There is no requirement for a power balance.
- Although PC is a necessary service for guaranteeing a proper power balance of the power network, there are only negative incentives and unreasonable costs for the BRP.
- SC influences the energy imbalance, not immediately the power balance.
- At the transition between PTUs there are too many, often conflicting, control signals active which will influence the power balance: the necessity to control the demanded energy in a PTU, and the actions from the PC en SC. The net effect is that up to 70% of the PC reserve capacity is used to counteract the control actions for achieving the demanded energy at the end of each PTU, so reducing the precious PC capacity for emergency situations to 30% of its intended value [10]. The amount of really available PC will reduce even more in the near future.
- The effects of these drawbacks on the present power balancing can introduce unwanted oscillations and even instability.

In the future more destabilizing trends are to be expected. For example:

- The dynamics of technical devices, control loops and markets are starting to overlap, introducing unexpected and unintended "stability" problems, as elucidated at the end of a PTU with large frequency deviations of up to 150 mHz within a time frame of 10 minutes [10].
- The grid will be used ever more for economic operation making some (cross-border) tie-lines to be loaded up to their maximum, which restricts the use of AS from far away. By sacrificing some economic profits, sufficient spare capacity must be allocated on the relevant (cross-border) connections. The spatial dimension of the grid really matters for AS [3-5].
- Many units become connected by power-electronic converters to the grid. These units become purposely insensitive for the actual frequency and voltages of the network, reducing the network constant c_{nw} and the equivalent inertia J of the network, resulting in less passive stability.

3. NEW ARRANGEMENTS FOR ANCILLARY SERVICES

Incentives and rules are required to guarantee both a reliable and a stable grid in spite of technological, economic and societal changes, competing BRPs, and cross-border trade. They have to guarantee low prices, high reliability, low sensitivity to the large uncertainties of renewables, low sensitivity to large, unexpected disturbances and sufficient incentives for upgrading the grid and the production capacity for future operations. This generic goal is not the natural aim of prosumers, neither of BRPs. It is important to note that, although the TSO could have better estimates of uncertainties in the system (and therefore for the AS needs) as it benefits more from the aggregation effects, the BRPs have more knowledge and more incentives for this estimation. These incentives include their desire for improving their time-varying uncertainty estimates as well as finding the optimal trade-offs between risks and economic benefits. The TSO, as the only 'consumer' of AS, has a monopoly and its only incentive is to be on 'the secure side', even if this security implies utilization of over conservative and less optimal solutions.

Therefore, in [1] a new ancillary services market is being proposed which is open, transparent with sufficient liquidity and with proper regulation and sufficient incentives for a reliable and economic power system. The ancillary services market is an *ahead* market to cope with expected uncertainties before operation. The quantities traded are options in energy [MWh] to receive or deliver within a PTU when needed. They can, but in general will not, be called into operation. BRPs assess their own uncertainties and liabilities. They define their own reserve needs for SC for the expected uncertainties in their production or demand. Any excess or deficit can be traded on an AS market. If the AS markets yields a cheaper solution compared with its own solution (e.g. switchable or adjustable loads), the BRP can select the market. *This ahead AS market-based approach* is being proposed as an attractive alternative of present SC.

Adequate modeling and thorough mathematical analysis presents firm theoretical justification for the policy to install "smart meters" and so price-based control, which helps consumers control their demand for power in response to evolving prices [12]. Price-based control has been proposed earlier, e.g. in [14]. Past years we have generalized these approaches to distributed and real-time implementations which can cope with only local information and hard transmission constraints and so yield local or nodal prices [4]. *This real-time, imbalance market approach* is being proposed as an attractive alternative of PC.

4. NEW ARRANGEMENTS FOR SECONDARY CONTROL

With AS markets [1,2,6,8], each BRP has to define its own expected production P_k [MWh] and consumption D_k [MWh] of energy for each considered PTU k. The expected difference E_k [MWh] ($P_k+D_k+E_k=0$) has to be assured by trading on the energy market (PX). However, both quantities P_k and D_k are associated with uncertainties. This uncertainty can, for example, be expressed by using so-called probability density functions (PDF) of both P_k and D_k, which express the probability that P_k/D_k has a certain value. The mean values will be partly a function of the price λ [€/MWh]: the higher the price, the higher the estimated production and the lower the expected demand. By combining the PDF's of both P_k and D_k, the PDF of E_k can be constructed or estimated. In Fig. 3 an example of such a PDF is elucidated. Given such a PDF the BRP has to decide which deterministic bid curve for $E_k(\lambda)$ he has to offer to the power exchange and to which risks the BRP will be exposed as, in general, the agreed value $E^{PX}{}_k$ at the PX market will not coincide with the value of E_k. Not satisfying the agreed energy $E^{PX}{}_k$ will result in costs incurred by the TSO. We distinguish between costs as a consequence of an agreed maximum size of the imbalance on the AS market, billed with the AS price, and the non predicted imbalance, billed with the imbalance price. Options to avoid these additional costs are better predictions of $E_k(\lambda^{PX}{}_k)$, depending on the expected price $\lambda^{PX}{}_k$ at the power exchange

- actively controlling its own power P_k and/or demand D_k to keep $E^{PX}{}_k+P_k+D_k=0$ or
- buying rights on the Ancillary Service market (AS) for a maximum energy imbalance in a PTU at lower prices than the imbalance price.

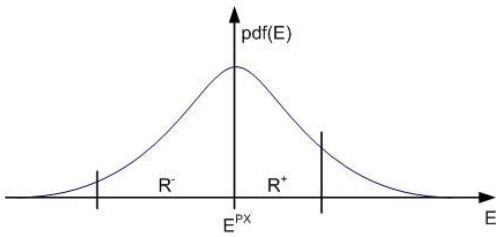


Figure 3, PDF of E_k, and selection $R^+{}_k$ and $R^-{}_k$.

As an open and transparent market will offer the required amount of energy at at least the same, but in general a better price, participating in the AS market is beneficial, compared with own arrangements for ancillary services. We propose two AS markets (AS^+, AS^-). In each market a BRP is requesting (R) ancillary services, is supplying (S) them or is passive. A request R is expressed as a maximum amount of energy [MWh]: $R^+{}_k$ [MWh] is the maximum amount of surplus energy and $R^-{}_k$ [MWh] the maximum amount of shortage energy that a BRP will try to compensate by trading on the ahead AS markets. The decision about these values $R^+{}_k$ and $R^-{}_k$ can be taken based on the PDF of E_k and the expected prices at the AS market $\lambda^{AS+-}{}_k$ and the expected imbalance price $\lambda^{imb}{}_k$, as elucidated in Fig. 3. In selecting $R^+{}_k = R^-{}_k = 0$, so being passive at the AS markets, all deviations $\Delta E_k = E_k - E^{PX}{}_k$ from the agreed $E^{PX}{}_k$ have to be paid based on the price at the imbalance market. With finite values of both $R^+{}_k$ and $R^-{}_k$, deviations $-R^-{}_k < \Delta E_k < R^+{}_k$ have to be paid based

on the AS market prices $\lambda^{AS+/-}_k$ and outside this interval based on the imbalance market price λ^{imb}_k. The price λ^{AS+}_k is used when there is a request to absorb too much energy, and λ^{AS-}_k when there is a request to deliver energy. The price λ^{imb}_k is used when the TSO detect a surplus of power in its control area, and λ^{imb}_k when the TSO detects a shortage of power. Using the PDF of ΔE_k, the expected costs can be calculated. A proper choice of both R^+_k and/or R^-_k reduces or even minimizes these expected costs. Based on these insights the BRP can make a proper selection for his bid curves $\lambda^{AS+}(R)$ and $\lambda^{AS-}(R)$ and the amounts R^+_k and R^-_k. Fig. 3 illustrates that the selection of an appropriate value for R^+_k or R^-_k is a trade-off between probabilities. By asking a fee from BRP's requesting AS and giving (part of) that amount to BRP's prepared to supply AS when asked by the TSO, transparent behavior is being supported. Just requesting large amounts of AS to avoid high cost when imbalance energy is needed, is therefore financially not a recommended strategy. The costs for requesting AS can be formulated, for example, as $c_0+c_1\lambda^{AS+}_k R^+_k$ [€] with c_0 [€] and c_1 [-] > 0. These bid curves are decreasing function $\lambda^{AS+}(R)$ and $\lambda^{AS-}(R)$. Each BRP can have for each PTU k two bid curves $\lambda^{AS+}_k(R)$ and $\lambda^{AS-}_k(R)$. The prices reflect the maximum affordable price for buying AS when needed. If the market price $\lambda^{AS+/-}(R)$ is higher, own alternatives have to be found, as the market is not willing to supply the required services for the stated maximum price. If the market price $\lambda^{AS+/-}(R)$ is lower, the market offers a cheaper solution than own alternatives.

A market not only needs demand (request) for AS, but also BRPs offering AS (supply). BRPs which have easily controllable or price-sensitive power and/or loads, can offer their excess capacity at the AS markets. They can make a profit from their ability to quickly supply (S) energy when needed by unexpected requests (R) from the TSO when an imbalance occurs in a control area. The AS supplying BRP's can offer in each PTU their bid curves $\lambda^{AS+}(S)$ and $\lambda^{AS-}(S)$ [€/MWh] and the maximum amounts S^+ and S^- [MWh]. The bid curve will be increasing functions of S. The prices reflect the minimum price $\lambda^{AS+-}(S)$ [€/MWh] for which the required option S [MWh] will be made available when demanded. When the market price $\lambda^{AS+-}(S)$ is lower, the BRP is not willing to supply the desired quantity of ancillary service. At the AS market the aggregated bid curves are added, both for the AS^+-market (request for absorbing energy: R too much energy, S: offers to absorb this energy when needed) and for the AS^--market (request for additional energy: R shortage of energy, S: offers to deliver this energy when needed). For each PTU k, separately for the AS^+ and AS^--market, prices λ^{AS+}_k and λ^{AS-}_k are determined and maxima for each BRP i ($R^+_{i,k}$, $R^-_{i,k}$, $S^+_{i,k}$, $S^-_{i,k}$) such that there is a balance between the requested ($R_{i,k}$) and supplied ($S_{i,k}$) ancillary services of all BRP's i:

$$\sum_i\{R^+_{i,k}(\lambda^{AS+}_k) + S^-_{i,k}(\lambda^{AS+}_k)\} = 0$$

$$\sum_i\{R^-_{i,k}(\lambda^{AS-}_k) + S^+_{i,k}(\lambda^{AS-}_k)\} = 0$$

Ultimately the buyer/seller of the ancillary service has to pay/will receive the agreed price (λ^{AS+}_k or λ^{AS-}_k) when AS is requested by the TSO. A unique market solution necessitates that the aggregated monotonously non-increasing curve $\lambda^{AS+}_k(R/S)$ crosses the monotonously non-decreasing aggregated curve $\lambda^{AS-}_k(R/S)$. With the market clearing prices there are unique combinations of BRPs which agree to prosume their offered bid when needed. When the deviations of R and/or S are outside the agreed values of the AS-markets, the TSO will ask for imbalance power with price λ^{imb}_k [€/MWh]. A necessary requirement for the expected prices will be: $\lambda^{PX}_k < \lambda^{AS+-}_k < \lambda^{imb}_k$ with, within a BRP, the marginal production costs λ^p_k[€/MWh] < λ^{PX}_k and the marginal consumption costs λ^c_k[€/MWh] > λ^{PX}_k. These price dependencies are illustrated in Fig. 4.

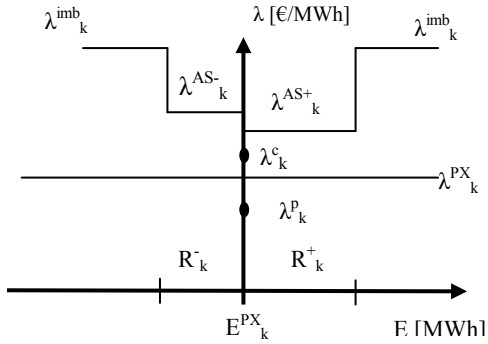


Figure 4, Dependencies among expected prices: $\lambda^p_k < \lambda^{PX}_k < \lambda^c_k < \lambda^{AS+/-}_k < \lambda^{imb}_k$

Now, the costs and profits of a BRP can be calculated. If a BRP consumes too much ($E_k < E^{PX}_k - R^-_k$) in a time period k, the following costs can be distinguished:

- fixed costs at the power exchange: $E^{PX}_k.\lambda^{PX}_k$
- fixed costs owing to the ancillary service market, reserving energy: $(c_0+c_1R^-_k\lambda^{AS-}_k) + (c_0+c_1\lambda^{AS+}_k R^+_k)$
- costs owing to the ancillary service market, using the maximum reserved energy: $R^-_k.\lambda^{AS-}_k$ and
- costs owing to having imbalance: $(E^{PX}_k - R^-_k - E_k).\lambda^{imb}_k$

The first amount is being paid at the PX, the second part to the TSO for reserving AS energy, the third part to the TSO for utilizing contracted AS energy in PTU k outside the agreed amount E^{PX}_k to a maximum R^-_k. The fourth contribution is owing to utilizing non-negotiated imbalance energy. As the BRP also earns money by selling the contracted power D_k with price λ^c_k to its internal consumers, and by paying for the energy P_k with price λ^p_k bought from its internal producers, its profit f^{profit}_k [€] becomes

$$f^{profit}_k = D_k\lambda^c_k - P_k\lambda^p_k - E^{PX}_k.\lambda^{PX}_k - (c_0+c_1\lambda^{AS+}_k R^+_k) - (c_0+c_1\lambda^{AS-}_k R^-_k) - R^-_k.\lambda^{AS-}_k - (E_k - E^{PX}_k - R^-_k).\lambda^{imb}_k$$

The maximum profit is achieved when $E_k = E^{PX}_k$, some less profit when the deviations are agreed on in the AS-markets ($-R^-_k \le E_k - E^{PX}_k \le R^+_k$) and considerable less when the deviations are exceeding the estimated and agreed values of R^+_k and R^-_k, as illustrated in Fig. 5, for a net-producing BRP with too

much energy (Request for AS). In Fig. 6 a net-consuming BRP which can supply (S) AS is illustrated. Without request from the TSO, its maximum profit is achieved when $E_k=E^{PX}{}_k$. When the TSO asks this BRP to supply AS and/or imbalance energy, its profits will increase.

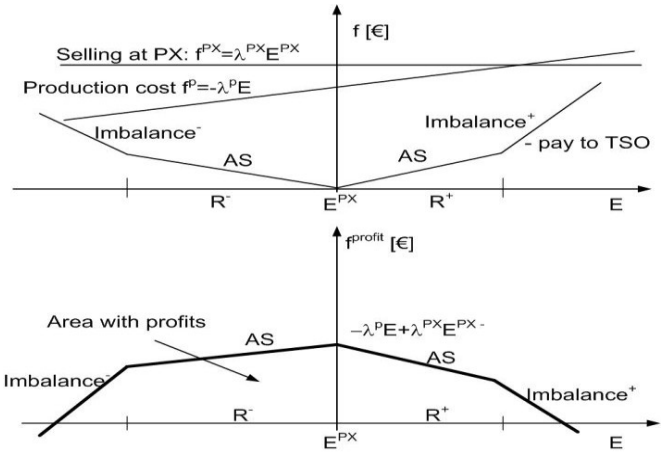


Figure 5, BRP net-producer, requests AS. Max. profit if $E_k= E^{PX}{}_k$.

Both Fig. 5 and Fig. 6 elucidate that the proposed market arrangements yield true financial incentives for maintaining the agreed prosumption of both the PX- and AS-markets. Yet, there are also incentives to request and supply AS when prices are appropriate

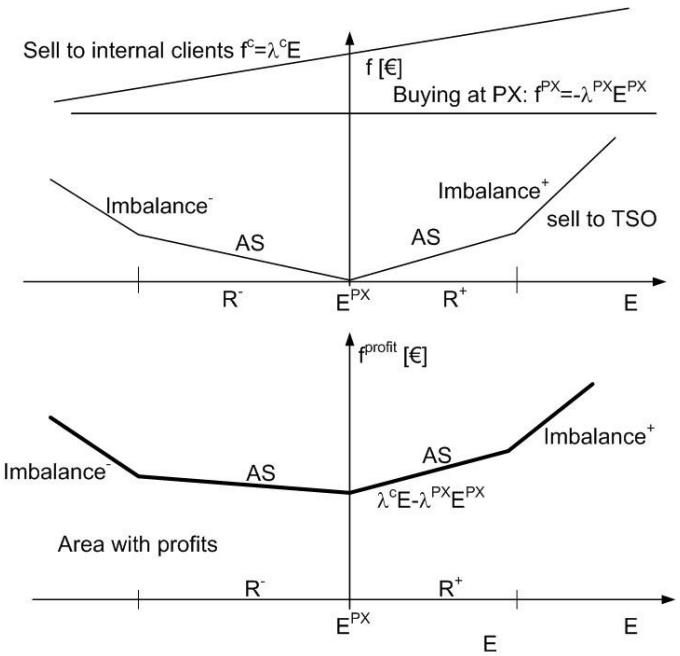


Figure 6, BRP net consumer, supplying AS.

Summary secondary control: With the proposed AS market and sufficient BRPs participating, this market mechanism can replace the present arrangements for SC. Each BRP can assess its own needs and options for AS. The TSO still has to operate at the AS markets for guaranteeing the control areas requirements on frequency, cross-border power deviations and emergency situations, but the majority of AS is traded among the BRP's. Network constraints introduce one-sided restrictions for AS. So, the AS are not homogenously distributed among the network, but discretely different. Also nodal pricing is needed when network restrictions occur [4-6]. The theory presented in [9,11] has the capability for devising novel distributed control schemes for optimal secondary control of the future European power network, even among countries.

Comparison: The BRPs will become responsible for their own estimation of AS. Needed AS can be obtained at the AS markets. BRPs can reduce their risks by buying AS at the AS market. There are consistent incentives for correctly estimating and trading the needs for AS. Both too high and too low estimates introduce additional costs. Owing to the two-sided market lower costs are to be expected, yet there are sufficient incentives to guarantee a required energy and power balance.

5. NEW ARRANGEMENTS FOR PRIMARY CONTROL

As already discussed, the primary control action is of crucial importance for the power balance and for the proper operation of a power system. Provision of PC is now enforced by regulations and as such it is not in line with an economically driven behavior of a BRP. As a consequence, inconsistencies arise in the system wide control actions, which could not only result in suboptimal use of energy sources, but also raise questions concerning feasibility and stability of the future, dynamically more stressed power systems. A solution to these problems can technically be obtained by utilizing a real-time price-based control scheme [8], as briefly described next and as illustrated in the example below.

Virtually all global network constraints, such as global power balance and transmission network power flow limits, can be adequately translated in real-time varying prices which, loosely speaking, quantitatively correctly reflect the economical value for satisfying each constraint at each time instant. Mathematically, this can be obtained through a suitable *dynamic* extension [9] of the so-called Karuch-Khun-Tucker (KKT) optimality conditions [15] related to the power flow equations and constraints in the transmission network. At any location in the network, an *imbalance price* is defined as a suitable combination of the Lagrange multipliers [15] from the KKT optimality conditions. These Lagrange multipliers are in turn updated in *real-time* based on the network frequency deviation measurements and on power flow measurements in the critical lines of a transmission system. The imbalance prices therefore reflect the current state of the physical system and give real-time incentives for prosumers to adopt their prosumption in such a way that integrity and stability of the system are protected. It is proven that such KKT price-based controllers yield an economically optimal power balancing solution, while the stability of the system can be assessed using suitable techniques from robust control theory [4]. An indication of the real-time price $\lambda^i(t)$ [€/MWh] is shown in Table 1. The costs f_k [€] incurred by the TSO depend on the sign of the

area control error (ACE) and the relative value of the actual power P(t) of a BRP exchanged with the grid with respect to the "average power levels" defined by $(E^{PX}_k+R^+_k)/T$, E^{PX}_k/T and $E^{PX}_k-R^-_k/T$, with T [h] the length of a PTU. The costs incurred by the TSO are calculated, for example, as follows. First the region {Imb, AS^+, AS^-, Imb} is being calculated depending on the value of P(t) with $P_{min} \le P(t) \le P_{max}$. Then, the time-continuous integral of the product of the actual price $\lambda^i(t)$ and the difference between actual power P(t) and the average power in a PTU, is calculated for each PTU k. With the power P(t) of a BRP in Imb, the costs are:

$$f_k = \int_0^T \{P(t) - \frac{E_k^{PX} + R_k^+}{T}\}\lambda^i(t)dt$$

There are many reasonable alternatives, e.g. by stating that all BRPs which contribute in decreasing the power imbalance get a reward, even if they are in an AS ($-R^-_k < \Delta E_k < R^+_k$) or in an imbalance region ($\Delta E_k < -R^-_k$ or $\Delta E_k > R^+_k$).

Table 1. Price $\lambda^i(t)$ [€/MWh] for energy incurred on a BRP by the TSO depending on the area control error (ACE) and the value of P(t) defined by $P_{min} \le P(t) \le P_{max}$. Length PTU is T [h].

ACE<0	ACE≈0	ACE>0	P_{min}	P_{max}
f<50 Hz	f≈50 Hz	f>50 Hz		
λ^{imb}_t	0	$-\lambda^{imb}_t$	$(E^{PX}_k+R^+_k)/T$	$>(E^{PX}_k+R^+_k)/T$
λ^{AS+}_k	0	$-\lambda^{AS+}_k$	E^{PX}_k/T	$E^{PX}_k+R^+_k)/T$
$-\lambda^{AS-}_k$	0	λ^{AS-}_k	$E^{PX}_k-R^-_k/T$	E^{PX}_k/T
$-\lambda^{imb}_t$	0	λ^{imb}_t	$<(E^{PX}_k-R^-_k)/T$	$E^{PX}_k-R^-_k)/T$

Notice, that de price λ^{AS+}_k for AS is fixed within a PTU and that the imbalance price λ^{imb}_t is time varying within a PTU.

6. REAL-TME, DISTRIBUTED, PRICE-BASED POWER BALANCE CONTROL

In this section we present, in some more details, the real-time price-based control strategy as a suitable approach to solve some of the challenging problems facing future, market-based power systems. More precisely, on the example of economically optimal power balance and transmission network congestion control, we present how global objectives and constraints can in real-time be translated into time-varying prices which adequately reflect the current state of the physical system. Assuming rational, profit maximization behavior of prosumers, we prove that the prices can be efficiently used as real-time control signals. In mathematical terms, the presented solution is obtained as a suitable dynamic extension of the so-called Karush-Kuhn-Tucker (KKT) optimality conditions related to the economically optimal power flow problem, and can be seen as a specific application of the general KKT control strategy which we developed in [1].

6.1 Problem definition

Suppose that in some node i in the network there is connected a prosumer which injects power p_i into that node. Without loss of any generality let us assume that there is only one prosumer connected to each node in the network. Let $J_i(p_i)$ represent the corresponding variable cost of production in a case of producer, and let it represent the negated benefit function in a case of consumer. Now suppose that at any time instant, a price λ_i [€/MWh] is announced at the node i. Assuming rational behavior, prosumer i will set its power injection level equal to the following value:

$$p_i(\lambda_i) = \arg\_max_{\tilde{p}_i}\{\lambda_i \tilde{p}_i - J_i(\tilde{p}_i)\} \quad (1)$$

For a given price λ_i, the expression $\lambda_i \tilde{\ }$ $\tilde{\ }$ in (1) is the total economical benefit the prosumer i receives when consuming/producing power $\tilde{\ }$. To complete the problem definition considered in this section, we still need to introduce the following definitions and notions.

In addition to the prosumers which react to the announced electricity price according to (1), in each node there are, in general, also prosumers which are price insensitive. For example, in case of a wind turbine, power production level at any time instant depends only on the wind speed and not on the price as well. We will use $\hat{p}_i$ to denote the aggregated power injection from such price-insensitive prosumers connected to the node i.

To describe the power flows in the network tie-lines for a given values of nodal power injections p_i, $\hat{p}_i$, $i = 1,...,n$, we will use a ``DC power flow" model, as follows. With $\delta_i, i = 1,...,n$ denoting a voltage phase angle at the node i, the power flow in a line connecting nodes i and j is given by $p_{ij} = b_{ij}(\delta_i - \delta_j) = -p_{ji}$. Here b_{ij} denotes the susceptance of the corresponding tie-line. If $p_{ij} > 0$, power in the line i-j flows from node i to node j, and the power balance in a node i yields $p_i + \hat{p}_i = \sum_{j \in I(N_i)} p_{ij}$, where $I(N_i)$ denotes the set of all nodes which are directly connected to the node i by a tie-line. Now, it is straightforward to derive the overall network balance condition

$$p + \hat{p} = B\delta, \quad (2)$$

where the $B \in R^{nxn}$ matrix is a symmetric matrix defined as follows. Its diagonal elements are given by $[B]_{ii} = \sum_{j \in I(N_i)} b_{ij}$, while for the off-diagonal elements $(i \neq j)$ we have $[B]_{ij} = \begin{cases} -b_{ij} & \text{if} \quad j \in I(N_i) \\ 0 & \text{if} \quad j \notin I(N_i) \end{cases}$.

Due to thermal and stability limits, power flow in each line in the network is limited to some maximal value, i.e. we have the following set of inequalities

$$b_{ij}(\delta_i - \delta_j) \le \bar{p}_{ij}, \quad \text{for all pairs } (i, j \in I(N_i)), \quad (3)$$

where $\bar{p}_{ij}$ denotes the corresponding maximal power flow. It will be convenient to rewrite the set of inequalities (3) in a compact matrix form: $L\delta \le \bar{p}_L$, where, as before we have $\delta = [\delta_1, \ldots, \delta_n]^T$, while L and $\bar{p}_L$ denote respectively the

matrix containing power lines susceptances b_{ij}, and the suitably defined vector containing the maximal power flow line limits.

Finally, we are ready to formally define the optimal nodal prices problem.

Optimal nodal prices (ONP) problem. For a constant value of price-inelastic power injections $\hat{p}$, find a vector of nodal prices $\lambda = [\lambda_1, \dots]^T$ which solves the following optimization problem

$$\min_{\lambda,\delta} \quad \sum_{i=1}^{n} J_i(p_i(\lambda_i)) \qquad (4a)$$

$$\text{subject to} \quad p(\lambda) - B\delta + \hat{p} = 0, \qquad (4b)$$

$$L\delta \le \overline{p}_L, \qquad (4c)$$

where $p(\lambda) = [p_1(\lambda_1), \dots]^T$ and $p_i(\lambda_i)$ is given by (1).

According to the above ONP problem, the vector of optimal nodal prices is defined as follows. Suppose that each prosumer is behaving rationally, i.e. according to (1). Then the optimal nodal prices are the prices for which the total economical welfare of the system is maximized (i.e. minimum in (4a) is attained) while the system is in balance (constraints (4b)) and no line in the network is congested (constraints (4c)).

In this section we are further concerned with the following, optimal price-based *control problem*. Suppose that power injection adjustment of a consumer *i* to the price λ_i is a dynamical process, i.e. that instead of the algebraic mapping (1), the mapping from price to power injection is described with a set of differential equations. Then the problem of finding on-line (i.e. in a real-time, during the system operation) the vector of optimal nodal prices becomes a complex control problem. Here, the goal is to quickly, in real-time, issue a vector of optimal nodal prices so that the overall, closed-loop system is stable and that, as the vector of price inelastic power injections $\hat{p}$ changes to some new value, the system always necessarily settles in the economically optimal working point. For more details on suitable modes describing the dynamics of a power system, the interested reader is referred to [2].

6.2 A solution: distributed KKT controllers

In this subsection we present a solution to the above described optimal price-based control problem. Sacrificing many details in the exposition, our goal here is to present the core ideas of the proposed approach and some of its desirable features, while for more details and for all the proofs, the interested reader is refereed to [2, 3].

In a dynamical setting, and having in mind the control problem described in the previous subsection, it is important to note that the violation of all the constraints appearing in (4) can be efficiently measured on-line. More precisely, violation of the power imbalance constraints (4a) can be detected by measuring network frequency deviations. Let $\Delta f = [\Delta f_1, \dots]^T$ denote the vector of network frequency deviations in the nodes (note that each node *i* can locally measure the value of Δf_i). Then $\Delta f = 0$ implies satisfaction of the constraints (4b). Furthermore, the line power flows, and therefore possible violations to the constraints (4c), i.e. the value Δp_L , can easily be directly measured. Schematically, the price-based control configuration is presented in Figure 7.

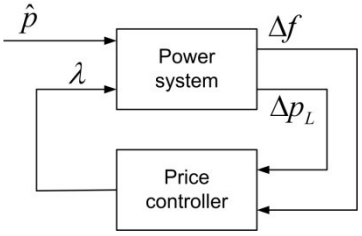


Figure 7. Price-based control scheme

A solution to the price-based control problem can be obtained by a suitable dynamic extension of the so-called Karush-Kuhn-Tucker (KKT) optimality conditions [4] related to the ONP problem. The KKT conditions, for our purpose, provide us with an explicit, algebraic characterization of the optimal nodal prices, as opposed to (4), which define them only implicitly as a solution to the optimization problem. In [2, 3] we have proposed the following price-based controller

$$\begin{pmatrix} \dot{x}_\lambda \\ \dot{x}_\mu \end{pmatrix} = \begin{pmatrix} -K_\lambda B & -K_\lambda L^\bullet \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_\lambda \\ x_\mu \end{pmatrix} + \begin{pmatrix} -K_f & 0 \\ 0 & K_p \end{pmatrix} \begin{pmatrix} \Delta f \\ \Delta p_L + w \end{pmatrix}, \qquad (5a)$$

$$w \ge 0, \quad K_o x_\mu + \Delta p_L + w \ge 0, \quad w^T (K_o x_\mu + \Delta p_L + w) = 0 \qquad (5b)$$

$$\lambda = \begin{pmatrix} I_n & 0 \end{pmatrix} \begin{pmatrix} x_\lambda \\ x_\mu \end{pmatrix}, \qquad (5c)$$

where x_λ and x_μ denote the controller states, Δf and Δp_L are the inputs to the controller, and the variable *w* is an ancillary variable used to enforce the so-called complementarity slackness conditions (see e.g. for more details [1,4,5]) appearing in the KKT conditions. The output λ is the vector of nodal prices, while the positive definite diagonal matrices K_λ, K_f, K_p and K_o are the controller gains. Due to the non-negativity and orthogonality conditions in (5b), the controller (5) belongs to the class of linear complementarity systems [6,5], which has recently received a significant attention in the hybrid systems community. For all the details of implementation of the controller (5), e.g. in the simulation software tools like MATLAB, as well as for the results concerning well-posedness and stability of the closed-loop system, the interested reader is refereed to [2,3].

The main idea behind the controller (5) is to use the integral control action to enforce, in steady state, the following conditions:

$$\Delta f = 0, \qquad (6a)$$

$$-\Delta p_L \geq 0, \quad x_\mu \geq 0, \; x_\mu^{\mathsf{T}} \Delta p_L = 0, \qquad (6b)$$

$$B\lambda + L^{\bullet} x_\mu = 0. \qquad (6c)$$

Note that the condition (6a) implies the power balance constraint (4a), while the conditions (6b) imply satisfaction of line flow limits (4c). The condition (6c) originates directly from the KKT optimality conditions, and is crucial as it guarantees that the vector of nodal prices is indeed the optimal one, in a sense of the ONP problem.

To summarize, the control law (5) defines the dynamics of the nodal price updates based on the network frequency deviations Δf and measurements of the line flow overloads Δp_L. Each price λ_i is in real-time announced at the corresponding node *i* and causes the prosumer at that node to adjust its power production according to its benefit maximization strategy based on (1). Note that the prosumer only knows this *local*, *current nodal price*, and has no knowledge of the system power imbalance or if some line in the network is being overloaded. The price forming controller (5) will however guarantee that these crucial *global* constraints are adequately reflected in the *local* nodal prices. Furthermore, since the KKT optimality conditions (derived from the ONP problem) are "built into" the controller, the prices are guaranteed to constantly drive the overall power system to the *globally economically optimal* work point. It is also important to note that the control law (5) does not involve the knowledge of local objective functions J_i of the prosumers (which are often confidential) and still it is able to deliver the correct prices. This is a consequence of structural properties of the KKT optimality conditions, which are further preserved in the controller (5).

Finally, we point out to the following advantageous feature of the presented controller. The matrices *B* and L^*, which are present in the optimality condition (6c) and in the controller dynamics (5a), are highly structured. This structure reflects the topology of the transmission system network and can be efficiently utilized for *distributed* implementation of the controller. It turns out that the price λ_i at some node *i* is directly related only to the prices in the neighboring nodes, i.e. to λ_j, where $j \in I(N_i)$. The control law (6) can therefore be implemented through a set of "nodal controllers", where one nodal controller *NC* is assigned to each node in the network, and each *NC* communicates only with the *NC*'s of the neighboring nodes.

The communication network graph among NC's is the same as the graph of the underlying physical network. Any change in the network topology requires only simple adjustments in NC's that are in close proximity to the location of the change. A distributed control structure is specially advantageous taking into account the large-scale of electrical power systems. Since in practice *B* is usually sparse, the number of neighbors for most of the nodes is small, e.g. two to four.

6.3 Example: price –based control of the IEEE New England power system

To illustrate the potential of the price-based control methodology in real-time control, we consider the widely used IEEE 39-bus New England test network [7]. The network topology, generators and loads are depicted in Figure 8. All generators in the system are modeled using the standard third order model used in automatic generation control studies [8], while quadratic functions are used to represent the variable production costs of the generators. The loads in the system are taken to be price inelastic, i.e. their value does not depend on the price. The simulation results of the distributed KKT price-based control scheme for real-time congestion management are presented in Figure 9. In the beginning of the simulation, the line flow limit in the line connecting the busses 25 and 26 was set to infinity, and the corresponding steady-state operating point is characterized by the unique price of 39.16 for all busses in the system. At time instant of 5s, the line limit constraint is imposed for this transmission line. The solid lines in Figure 9 are simulated trajectories of nodal prices, calculated according to the KKT control strategy, for the generator buses (i.e. for buses 30 to 39) which is where the generators are connected. In the same figure, dotted lines indicate the off-line calculated values of the corresponding steady-state economically optimal nodal prices. For clarity, the trajectories of the remaining 29 nodal prices were not plotted. In the simulation, all these trajectories converge to the corresponding optimal values of nodal prices as well and the obtained results clearly illustrate the efficiency of the proposed distributed control scheme.

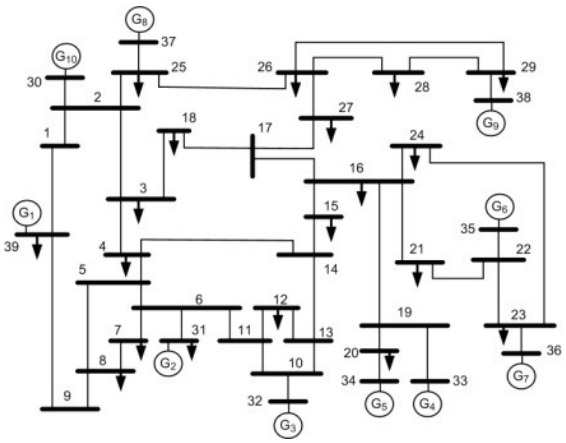


Figure 8. IEEE 39-bus New England test system.

The obtained simulation results illustrate the efficiency of the proposed price-based control scheme in solving a *real-time* congestion management problem, which is considered to be one of the toughest problems in electricity market design prices were not plotted. In the simulation, all these trajectories converge to the corresponding optimal values of nodal prices as well and the obtained results clearly illustrate the efficiency of the proposed distributed control scheme.

The obtained simulation results illustrate the efficiency of the proposed price-based control scheme in solving a *real-time*

congestion management problem, which is considered to be one of the toughest problems in electricity market design [14].

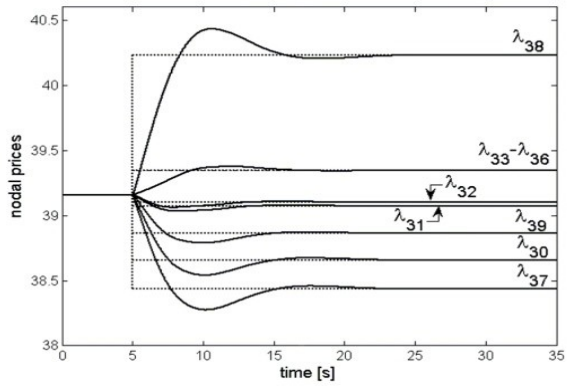


Figure 9. Trajectories of real-time updated nodal prices for generator buses, i.e., for busses 30-39 where the generators are connected.

Present situation: PC is enforced, can even introduce imbalance costs and is not paid for. Still, PC is necessary and can react autonomously without interaction by the TSO.
Proposed situation: In [2-4] it is shown that when the market dynamics are comparable with the power system dynamics, real-time, price-based control is a realistic option and can replace PC. BRPs have proper and consistent financial incentives to make economic viable decisions about power and ancillary services. A real-time price signal, calculated by the TSO, invites them to adjust their prosumption. The proposed AS markets guarantee a cost-effective and reliable solution for the ancillary services [2,6,9,11].

7. CONCLUSIONS

The present arrangements for ancillary services for power and energy balance show insufficient and inconsistent incentives for BRPs and TSOs to deal with risks and economic operation. The introduction of an *ahead market* for ancillary services guarantees an *energy balance*. It enforces that the estimation of the size and the character of these ancillary services are determined by the BRPs themselves. The BRP takes the decision to distribute its resources among the energy and the AS market. The AS settlements improve the existing secondary control arrangements. Primary control can be replaced by a *real-time, price-based, distributed control strategy* operated by the TSO with real-time changing imbalance prices depending on the area control error. This new, consistent primary control strategy ensures a *proper power balance* in the control area of a TSO, satisfying global constraints and local economic preferences. The proposed ahead AS market and the real-time, price-based distributed control strategy guarantee a cost-effective and reliable solution for ancillary services.

REFERENCES

[1] P.P.J. van den Bosch, A. Jokic, J. Frunt, W.L. Kling, F. Nobel, P. Boonekamp, W. de Boer, R.M. Hermans, Incentives-based ancillary services for power system integrity, Proceedings 6th International Conference on the European Electricity Market, Leuven, May 2009.

[2] Alvarado, F.L., Meng, J., DeMarco, C.L., Mota, W.S., Stability Analysis of Interconnected Power Systems with Market Dynamics, IEEE Transactions on Power Systems, 16 (4), pp. 695-701, 2001.

[3] Alvarado, F.L., Understanding Locational Reserves and Reliability Needs in Electricity Markets, Hawaii International Conference on System Sciences, USA, 2006.

[4] Jokic, M. Lazar, P.P.J. van den Bosch (2009). *Real-time control of power systems using nodal prices, International Journal of Electrical Power & Energy Systems,* Vol. 31, pp. 522-530

[5] Christie, R.D., Wollenberg, B.F., Wangensteen, I., Transmission Management in the Deregulated Environment, Proceedings of the IEEE, 88 (2), pp. 170-195., 2000.

[6] Y. Rebours, D. Kirschen, M. Trotignon, and S. Rossignol, A Survey of Frequency and Voltage Control Ancillary Services–Part II: Economic Features, *IEEE Transactions on power systems*, vol. 22, 2007, p. 358–366.

[7] DeMarco, C.L., Control Structures for Competitive, Market-driven Power Systems, IEEE Conference on Decision and Control, 2001.

[8] Jokic, A. Price-based Optimal Control of Electrical Power Systems, PhD thesis, Eindhoven University of Technology, Eindhoven, The Netherlands, 2007.

[9] Jokic, A., M. Lazar, P.P.J. van den Bosch (2009*). On constrained steady-state regulation: Dynamic KKT controllers, IEEE Transactions on Automatic Control,* Vol. 54, No. 9, pp. 2250-2254

[10] Tractabel Engineering, Study of the Interaction and Dependencies of Balancing Markets, Intraday Trade and Automatically Activated Reserves, Feb 2009. http://ec.europa.eu/energy/gas_electricity/ studies/doc/electricity/2009_balancing_markets.pdf

[11] A.C.R.M. Damoiseaux, A. Jokic, M. Lazar, A. Alessio, P.P.J. van den Bosch, I. Hiskens, A. Bemporad, Assessment of Decentralized Model Predictive Control Techniques for Power Networks, In Proceedings of the 16th Power Systems Computation Conference (PSCC 2008), Glasgow, UK, July 2008.

[12] I-K. Cho, S.P. Meyn, Dynamics of Ancillary Service Prices in Power Distribution Systems, in Proceedings of the 42nd IEEE Conference on Decision and Control, USA, 2003

[13] M.A. Pai, *Energy Function Analysis for Power System Stability*, Kluwer Academic Publishers, 1989.

[14] C. Harris, "Electricity Markets: Pricing, Structures and Economics", Wiley, 2006.

[15] P. Kundur, *Power System Stability and Control*, McGraw-Hill, 1994.

[16] A. Jokic, *Price-based Optimal Control of Electrical Power Systems*, PhD thesis, Eindhoven University of Technology, The Netherlands, 2007.

[17] S. Boyd and L. Vandenberghe, *Convex optimization*, Cambridge University Press, 2004.

[18] J.M. Schumacher (2004). *Complementarity systems in optimization*. Mathematical programming B. Vol. 101, pp 263-296.

Solutions to Riccati Differential Equations in LQ Problems

Corneliu Botan and Florin Ostafi

Abstract— **Linear quadratic (LQ) optimal problems with finite final time are considered. The studied problems are with free end-point and with fixed end-point, respectively. Some new results referring to the solutions to Riccati differential equation are presented. A unified approach for both problems (with fixed and free end-point) allows establishing of the analytical formulae for the solution to the same Riccati equation, with different terminal conditions. Certain procedures which start from an initial adequate condition are established for the problem with free end-point. It is avoided by this way the necessity of the computing in inverse time of the solution to Riccati equation.**

I. Introduction

ALTHOUGH the optimal control has doubtless advantages and there are many important theoretical results, the number of applications is nowadays very small. We appreciate that a cause of the reluctance in this direction is the complexity of the algorithms, so that each simplification in implementation of the procedures are welcome.

We consider that in the next period the introducing of optimal control will become a necessity, especially for energy saving. One of applications is the electrical drive systems, where the optimization is appreciated as a main direction of the developing in the future [1]. On the other hand, the computing time in such applications represents a strong restriction, tacking into account the fact that the usual sampling period has only few milliseconds and the modern control techniques of the electrical drives (like the vector control) imply a great amount of operations. This example justifies the above mentioned necessity of the simplifying of the current existing methods for optimal control.

The present paper is dedicated to the mentioned aim, with references to the linear quadratic (LQ) optimal problems with finite final time and fixed or free end-point. This well known problem is treated in very much number of papers or books. We mention hear for exemplification [2], [3], [4], [5], [6] as fundamental books from different periods, referring to this problem.

The paper refers to the Riccati differential equation, which is the key for solving of the mentioned problems. The analytical formulae for the solutions to Riccati equation are proposed. The analytical formula for LQ problem with free end-point allows the computation of the initial value of the solution to Riccati equation, and then to develop certain procedures based on this value. This fact avoids one of the main difficulties of the existing methods, namely the necessity of the solving Riccati equation in inverse time. All the presented techniques and formulae have advantages in comparison with existing methods since they make shorter the computing time and allow an easier implementation.

The paper also put in evidence a similitude between free and fixed end-point LQ problems. In fact, the same Riccati equation arises in both cases. The difference consists in terminal (initial or final) condition which holds in every case.

Certain common aspects for both mentioned problems are presented in the next section. The sections 3 and 4 presents the basics results for LQ problems with fixed and free end-point, respectively. Some general remarks, simulation results and conclusions end the paper.

Manuscript received April 26, 2010. This work was supported by The National Centre for Programs Management from Romania under the research grant SICONA – 12100/2008.

C. Botan is with the Technical University of Iasi, Romania (phone: +40232 278683; fax: +40232 230751; e-mail: cbotan@ac.tuiasi.ro).

F. Ostafi is with the Technical University of Iasi, Romania (e-mail: fostafi@ac.tuiasi.ro).

II. General Aspects

Certain general aspects referring to both LQ studied problems (with fixed and free end-point) are presented below.

A. Problems formulation and basic optimality conditions

The LQ problems refer to a linear system and a quadratic criterion. In the sequel, only the problems with finite final time are considered. Usually LQ denomination is adopted for the problems with free end-point, but the problems with fixed end-point can be considered in the same category, since they also refer to a linear system and a quadratic cost function.

A linear time-invariant system is considered (some remarks to the time-variant case are presented in the section 5)

$$\dot{x}(t) = Ax(t) + Bu(t),\ x \in \mathbf{R}^n,\ u \in \mathbf{R}^m. \quad (1)$$

The general form of the considered criterion is

$$J = \frac{1}{2}x^T(t_f)Sx(t_f) + \frac{1}{2}\int_{t_0}^{t_f}\left[x^T(t)Qx(t) + u^T(t)Pu(t)\right]dt \quad (2)$$

(T denotes the transposition), $S \geq 0, Q \geq 0, P > 0$.

The initial moment t_0 and the initial state $x(t_0) = x^0$ are fixed. The following problems are considered:

P1 *(with fixed end-point)*: find the optimal control $u(t)$ which transfer the system (1) from x^0 in the final state $x(t_f) = x^f = 0$ (t_f is fixed) so that to be minimized the

performance index (2), with $S = 0$.

P2 *(with free end-point)*: find the optimal control $u(t)$ which transfer the system (1) from x^0 in a free point (t_f is fixed) so that to be minimized the performance index (2).

The problems **P1** and **P2** will be studied using a common procedure, the aim referring especially to the Riccati differential equation. Although there are frequently references to procedures for **P1** problems that do not appeal to Riccati equation, one will indicate that it is useful to use this equation in such problems and there is a similitude between the two cases. There are many references for **P2** problems and corresponding Riccati equations. We mention here [2],…,[6]. In [2] and [4] are also presented important aspects referring to **P1** problems. Some especial aspects referring to Riccati equation in **P1** problems are indicated in [7], [8]. The Hamiltonian of the problem is

$$H = \frac{1}{2}\left[x^T(t)Qx(t) + u^T(t)Pu(t)\right] + \\ + \lambda^T(t)\left[Ax(t) + Bu(t)\right], \tag{3}$$

where $\lambda(t) \in \mathbf{R}^n$ is the co-state vector. The necessary optimality condition $\partial H / \partial u = 0$ leads to

$$u(t) = -P^{-1}(t)B(t)\lambda(t). \tag{4}$$

The Hamilton canonical equations, combined with (4) can be written in the form

$$\dot{\gamma}(t) = G\gamma(t), \tag{5}$$

where

$$\gamma(t) = \begin{bmatrix} x(t) \\ \lambda(t) \end{bmatrix} \in \mathbf{R}^n,\ G = \begin{bmatrix} A & -N \\ -Q & -A^T \end{bmatrix} \in \mathbf{R}^{2n\times 2n},\ N = BP^{-1}B^T. \tag{6}$$

The transition matrix for G is

$$\Gamma(t,\theta) = \begin{bmatrix} \Gamma_{11}(t,\theta) & \Gamma_{12}(t,\theta) \\ \Gamma_{21}(t,\theta) & \Gamma_{22}(t,\theta) \end{bmatrix} \in \mathbf{R}^{2n\times 2n}, \\ \Gamma_{ij}(t,\theta) \in \mathbf{R}^{n\times n},\ i,j = 1,2. \tag{7}$$

The solution to the system (5) is

$$\gamma(t) = \Gamma(t,\theta)\gamma(\theta). \tag{8}$$

Depending on the terminal conditions of the problem, one will adopt $\theta = t_0$ or $\theta = t_f$.

If it is desired to obtain a feedback optimal control, the co-state vector $\lambda(t)$ from (4) has to be expressed as a function of the state vector

$$\lambda(t) = \tilde{R}(t)x(t),\ \tilde{R}(t) \in \mathbf{R}^{n\times n} \tag{9}$$

and than the optimal control becomes

$$u(t) = -P^{-1}B^T\tilde{R}(t)x(t). \tag{10}$$

Using (5), (6), (9) and (10) one obtain that $\tilde{R}(t)$ satisfies the well known Riccati matrix differential equation (RMDE)

$$\dot{\tilde{R}}(t) = \tilde{R}(t)N\tilde{R}(t) - A^T\tilde{R}(t) - A\tilde{R}(t) - Q. \tag{11}$$

Remark 1: All presented equations are valid for both **P1** and **P2** problems, since no reference to terminal conditions were used.

Remark 2: There are procedures for **P1** problem based on the direct using of the solution (8) to the equation (5), but this solution has a complicated form, especially for feedback control. For the **P2** problem, the equations (10) and (11) are used.

A common approach for the both problems will be used in the sequel, with accent on the solution to the RMDE. The same RMDE appears in both problems, the difference consisting in the terminal conditions for this equation.

A great difficulty in the **P2** problem case is the fact that RMDE must be solved in inverse time, starting from

$$\tilde{R}(t_f) = S. \tag{12}$$

This implies to beforehand compute and memorize the solution to RMDE at every sampling moment, or to take again the inverse time computing at every sampling period. According to these procedures, it is not possible to establish a direct time iterative method for RMDE solving.

The present paper establishes similar analytical computing formulae for RMDE solution for **P1** and **P2** problems. Moreover, one can find the initial value $\tilde{R}(t_0)$ for the **P2** problem and this fact allows establishing of an analytical formula which starts from the initial moment t_0, or allows the introducing of a direct time numerical iterative techniques.

B. Transformation of the canonical equations

The solution (8) to the system (5) is not adequate for analytical developments, since the matrix blocks Γ_{ij}, *i,j=1,2* , of the transition matrix Γ can not be analytical computed. In order to overcome this difficulty, it is proposed to search the co-state variable $\lambda(t)$ in the form

$$\lambda(t) = Rx(t) + v(t) \tag{13}$$

instead of (9). *R* and *v(t)* in (13) are unknown constant matrix and variable vector, respectively. The relationship (13) allows the changing of the variables introducing the vector

$$\rho(t) = \begin{bmatrix} x(t)^T & v(t)^T \end{bmatrix}^T \in \mathbf{R}^{2n} \tag{14}$$

The system for new variables becomes

$$\dot{\rho}(t) = H\rho(t), \tag{15}$$

with

$$H = \begin{bmatrix} F & -N \\ H_{21} & -F^T \end{bmatrix}, \tag{16}$$

with

$$F = A - NR\,. \tag{17}$$

It is easy to verify that the matrix block H_{21} is zero if the matrix *R* satisfies the equation

$$RNR - R^T A - A^T R - Q = 0 \tag{18}$$

and therefore

$$H = \begin{bmatrix} F & -N \\ 0 & -F^T \end{bmatrix} \tag{19}$$

Evidently, (18) is the Riccati matrix algebraic equation, which appears in the similar optimal control problem with infinite final time.

The solution to the equation (15) is

$$\rho(t) = \Omega(t,\theta)\rho(\theta)\,, \tag{20}$$

where $\Omega(t,\theta)$ is the transition matrix for *H*. Taking into account the form (19) for *H* and the relationships

$$\dot{\Omega}(t,\theta) = G\Omega(t,\theta) \tag{21}$$

and

$$\Omega(\theta,\theta) = I_{2n}\,, \tag{22}$$

one can prove that

$$\Omega(t,\theta) = \begin{bmatrix} \Psi(t,\theta) & \Omega_{12}(t,\theta) \\ 0 & \Phi(t,\theta) \end{bmatrix}, \tag{23}$$

where $\Psi(.)$ and $\Phi(.)$ are the transition matrices for *F* and $-F^T$, respectively and $\Omega_{12}(.)$ satisfies the equation

$$\dot{\Omega}_{12}(t,\theta) = F\Omega_{12}(t,\theta) - N\Phi(t,\theta), \quad \Omega_{12}(\theta,\theta) = 0 \tag{24}$$

and it is

$$\Omega_{12}(t,\theta) = \int_t^\theta \Psi(t,\tau)N\Phi(\tau,\theta)d\tau\,. \tag{25}$$

By this way, the solution to the system (15) can be established using only *n* x *n* transition matrices. This is an advantage in comparison with the well known procedures based on the factorization of the solution to the RMDE and derived from Radon's Lemma [6]; the last mentioned procedures use *2n* x *2n* transition matrices. The indicated advantage results from the particular form (19) of the matrix *H* of the new system (15). The matrix blocks of the matrix $\Omega(t,\theta)$ can be computed significantly easier than ones of the $\Gamma(t,\theta)$. The matrix blocks of $\Gamma(t,\theta)$ can be expressed in terms of $\Omega(t,\theta)$; based on the above transformations, one can prove that

$$\begin{aligned} &\Gamma_{11}(\cdot) = \Psi(\cdot) - \Omega_{12}(\cdot)R, && \Gamma_{12}(\cdot) = \Omega_{12}(\cdot) \\ &\Gamma_{12}(\cdot) = R\Psi(\cdot) - R\Omega_{12}(\cdot)R - \Phi(\cdot)R, && \Gamma_{22}(\cdot) = R\Omega_{12}(\cdot) + \Phi(\cdot) \end{aligned} \tag{26}$$

III. The Problem with Fixed End-Point (P1)

One obtain for the first equation (8), for $t = t_f$ and $\theta = t_0$ the final state vector $x(t_f) = \Gamma_{11}(t_f,t_0)x^0 + \Gamma_{12}(t_f,t_0)\lambda^0$.

Since $x(t_f) = 0$, it results

$$\lambda^0 = \lambda(t_0) = -\Gamma_{12}^{-1}(t_f,t_0)\Gamma_{11}(t_f,t_0)x^0\,. \tag{27}$$

One can prove [9] that the matrix $\Gamma_{12}(t_f,t_0)$ is a non-singular matrix if the pair (A,B) is completely controllable.

Comparing (27) with (9) for $t = t_0$ yields

$$\tilde{R}(t_0) = -\Gamma_{12}^{-1}(t_f,t_0)\Gamma_{11}(t_f,t_0) = R + \Omega_{12}^{-1}(t_f,t_0)\Psi(t_f,t_0)\,. \tag{28}$$

For $t = t_0$, one obtains from (13)

$$v^0 = v(t_0) = \lambda^0 - Rx^0 \tag{29}$$

and taking into account (27) and (28), one obtains

$$v^0 = \left[\tilde{R}(t_0) - R\right]x^0\,. \tag{30}$$

Then, it results from the first equation from (20), written for $\theta = t_0$,

$$x(t) = M_1(t,t_0)x^0\,, \tag{31}$$

with

$$M_1(t,t_0) = \Psi(t,t_0) + \Omega_{12}(t,t_0)\left[\tilde{R}(t_0) - R\right]. \tag{32}$$

We are now in position to formulate the

Theorem 1: The solution to RMDE (11) with the initial condition $\tilde{R}(t_0)$ is given by

$$\tilde{R}(t) = R + \Phi(t,t_0)\left(\tilde{R}(t_0) - R\right)M_1^{-1}(t,t_0)\,. \tag{33}$$

Proof: The solution to the second equation from (15) is

$$v(t) = \Phi(t,t_0)v^0\,. \tag{34}$$

Then, one can write from (20) and (30):

$$\lambda(t) = Rx(t) + \Phi(t,t_0)\left(\tilde{R}(t_0) - R\right)x^0\,.$$

In this equation x^0 is replaced with $x(t)$ from (31) and comparing with (9), it results (33) ■

Remark 3: The matrix $M_1(t,t_0)$ is non-singular since defines the transition from $x(t_0)$ to $x(t)$. However, it results from (31) that $M_1(t_f,t_0) = 0$ because it was imposed

$x(t_f)=0$. In this case, the computing fails for $t=t_f$ and we must renounce to perform the last steps (the simulation results have indicated that it is sufficient to eliminate no more than one step).

Remark 4: The relationship (33) represents an analytical computing formula for the solution to RMDE which appears in the LQ problem with fixed end-point and having specified the initial value $\tilde{R}(t_0)$. This formula can be used as such, but it is possible to introduce in addition certain iterative computing. For instance, the transition matrices can be computed with

$$\begin{aligned}\Psi_{i+1}&=\Psi(t_{i+1},t_0)=\Psi_i\cdot\Psi_\delta,\ \Psi_0=I,\\ \Phi_{i+1}&=\Phi(t_{i+1},t_0)=\Phi_i\cdot\Phi_\delta,\ \Phi_0=I,\end{aligned}\tag{35}$$

where $\Psi_\delta=\Psi(t_{i+1},t_i)$ and similar for Φ_δ. An iterative computing can be also performed for $\Omega_{12}(.)$ starting from (24)

$$\Omega_{12}(t_{i+1},t_0)=\left(I-\delta F\right)\Omega_{12}(t_i,t_0)-\delta N\Phi(t_i,t_0),\tag{36}$$

with $\delta=t_{i+1}-t_i$ and $\Omega_{12}(t_0,t_0)=0$.

IV. The Problem with Free End-Point (P2)

The RDME (11) is solved in this case starting from the final value (12). A similar procedure is used, namely it is established a relation in the form (13) between $\lambda(t)$ and $x(t)$ and the result is compared with (9). For this aim, it is expressed from the first equation (20) written for $\theta=t_f$

$$x(t)=\Psi(t,t_f)x(t_f)+\Omega_{12}(t,t_f)v(t_f),$$

or

$$x(t)=M_2(t,t_f)x(t_f),\tag{37}$$

with

$$M_2(t,t_f)=\Psi(t,t_f)+\Omega_{12}(t,t_f)(S-R).\tag{38}$$

The formulae (37) and (38) were obtained from the previous relationship, taking into account that

$$v(t_f)=(S-R)x(t_f).\tag{39}$$

This last expression results from (13) written for $t=t_f$. From the second equation (20), it results

$$v(t)=\Phi(t,t_f)v(t_f)=\Phi(t,t_f)(S-R)x(t_f).\tag{40}$$

It is possible now to formulate the following

Theorem 2: The solution to the RMDE (11) with the final condition (12) is

$$\tilde{R}(t)=R+\Phi(t,t_f)(S-R)M_2^{-1}(t,t_f).\tag{41}$$

Proof: The vector $v(t)$ can be expressed as a function of $x(t)$ taking into account (37) and then it can write

$$\lambda(t)=\left[R+\Phi(t,t_f)(S-R)M_2^{-1}(t,t_f)\right]x(t).$$

Comparing this equation with (9), one obtains (41). The matrix $M_2(t,t_f)$ is non-singular since it express the transition between $x(t_f)$ and $x(t)$. ■

An analytical computing formula is given by the Theorem 2 for the solution to the **P2** problem. A solution in a closely form with (41) is proved in [10] by straightforward computing. As in the Section 3, this formula implies only *n* x *n* transition matrices. As in other methods, the computing has t_f as reference moment, but anyway, the use of the formula (41) has significant advantages from the simplification and the precision point of view.

Moreover, the Theorem 2 allows the establishing of the initial value $\tilde{R}(t_0)$ and, on this basis, certain procedures using this terminal condition can be developed. For this goal, it can write from (41) for $t=t_0$

$$\tilde{R}(t_0)=R+W,\tag{42}$$

with

$$W=\Phi(t_0,t_f)(S-R)M_2^{-1}(t_0,t_f).\tag{43}$$

Theorem 3: The solution to RMDE (11) with the final condition (12) is

$$\tilde{R}(t)=R+\Phi(t,t_0)WM_0^{-1}(t,t_0),\tag{44}$$

with

$$M_0(t,t_0)=\Psi(t,t_0)+\Omega_{12}(t,t_0)\left[\tilde{R}(t_0)-R\right].\tag{45}$$

Proof: From the equation (9) written for $t=t_0$ and from the first equation (8), it results

$$x(t)=[\Gamma_{11}(t,t_0)+\Gamma_{12}(t,t_0)\tilde{R}(t_0)]x^0.$$

If the matrices are replaced from (26), one can write

$$x(t)=M_0(t,t_0)x^0,\tag{46}$$

with $M_0(t,t_0)$ non-singular and given by (45).

The second equation (8) can be written

$$\lambda(t)=[\Gamma_{21}(t,t_0)+\Gamma_{22}(t,t_0)\tilde{R}(t_0)]x^0.$$

If x^0 is replaced from (46) and it is compared the last expression with (9), one obtains (44) ■

Iterative procedures for real time computing of the variant matrices in (44) can be adopted similarly with those mentioned for (33).

V. REMARKS

(a) Taking into account the transitions expressed by (37) and (46), it results

$$M_0(t_f,t_0)=M_2^{-1}(t_0,t_f)\,. \tag{47}$$

(b) One can remark that the formulae given by the above three theorems are similar to some extent. All of them use only *n* x *n* transition matrix. They are solution to the same RMDE, but for different terminal conditions.

(c) The established formulae can be extended to the time variant case, when the matrices in (1) and / or (2) are time variant. For instance, instead of (41) (Theorem 2), the solution to RMDE is

$$\tilde{R}(t)=\overline{R}(t)+\Phi(t,t_f)(S-\overline{S})M_2^{-1}(t,t_f)\,, \tag{48}$$

where $\overline{R}(t)$ is a particular solution of RMDE with final condition $\overline{R}(t_f)=\overline{S}$. It should be noted that the implementation advantages for the time variant case are not very significant. Indeed, in many cases, the finding of a particular solution can introduce the same difficulties as the original problem.

(d) The above mentioned solutions (including also (48)) extend a well known solution to the differential scalar Riccati equation. In this case, it is necessary to know a particular solution (as $\overline{R}(t)$ in (48), or *R* in previous theorems) and then to solve a linear differential equation.

(e) The analytical established formulae can be applied for implementation, but in both cases, different numerical iterative (in direct time) procedures can be used, starting from the known initial matrix $\tilde{R}(t_0)$. For instance, a simple way is to approximate the derivative in (11) and than one obtain

$$\begin{aligned}&\tilde{R}_{i+1}=\tilde{R}_i+\delta(\tilde{R}_iN\tilde{R}_i-\tilde{R}_iA-A^T\tilde{R}_i-Q);\\&\delta=t_{i+1}-t_i=const.,\end{aligned} \tag{49}$$

where $\tilde{R}_i=\tilde{R}(t_i)$ and the initialization is $\tilde{R}_0=\tilde{R}(t_0)$, specific for each problem. One has to remark that some procedures of this type (for instance, (49)) impose a great number of steps for an adequate accuracy.

(f) The optimal control can be expressed from (4) and (13):

$$u(t)=u_f(t)+u_c(t)\,, \tag{50}$$

where

$$u_f(t)=-P^{-1}B^TRx(t) \tag{51}$$

is a feedback component (identical with the feedback in the similar LQ problem with infinite final time) and

$$u_c(t)=-P^{-1}B^TRv(t)=-P^{-1}B^T\Phi(t,t_0)v(t_0) \tag{52}$$

is a corrective component.

The difference between the **P1** and **P2** problems consists in the initialization $v(t_0)$. The procedures of this type will be named in the sequel as first procedure for optimal controller implementation (FPOCI). Similarly, the procedures based on (10) and above established solutions for RMDE will be named as second procedures for optimal controller implementation (SPOCI).

(g) It is well known that the formulated problems have unique solution (supplementary controllability condition is necessary for **P1** problems). For this reason, all different methods have to lead to the same result (neglecting the numerical computing errors).

(h) Similar results can be obtained for the discrete time LQ problems. The space restrictions do not allow presentation of the discrete case in this paper.

VI. EXAMPLES

Two categories of examples and simulation results will be presented below. The first one refers to the computing of the solution to RMDE for free end-point problem. Three examples were considered (for a second, fourth and tenth order system, respectively) and in each case were used three computing methods: two methods use the analytical formulae given by Theorems 2 and 3 and the third one uses a numerical iterative technique, starting from the initial value (for instance, based on (49)). A comparison between the results indicates a very high precision and coincidence for the analytical formulae. The maximum value of the ratio $\rho_2=\|\tilde{R}_1(t)-\tilde{R}_2(t)\|/\|\tilde{R}_1(t)\|$ is less than $6\cdot10^{-13}$. An acceptable precision for the third (numerical) method (ratio $\rho_3=\|\tilde{R}_1(t)-\tilde{R}_3(t)\|/\|\tilde{R}_1(t)\|$ less than $3\cdot10^{-3}$) is achieved only for a significant decrease of the increment δ. In the previous expressions, $\tilde{R}_1(t),\tilde{R}_2(t),\tilde{R}_3(t)$ denote the solutions to RMDE obtained with the above mentioned methods (equations (41), (44) and (49), respectively). The mentioned values for ρ_2 and ρ_3 are maximal for all exemplified cases and for all *t*. The value of ρ_3 depends on the increment δ. For instance, for the considered second order system (see below the adopted matrices), one obtained $\rho_3=3.2\cdot10^{-2}$ for $\delta=T/10$ (T is the sampling period) and $\rho_3=0.63\cdot10^{-2}$ for $\delta=T/50$.

Besides precision, the computing time was analyzed. For this aim, adequate MATLAB functions (TIC and TOC) were used in order to appreciate the total computing time of the optimal control on the interval $[t_0,t_f]$. Close values were obtained for the two analytical methods (based on (41) and (44)) and a value three times greater for the numerical procedure (based on (49)) for $\delta=T/50$. By comparison, this last case offers a value twelve times better as the classical procedure, based on the same iterative formula (49),

but used in inverse time, starting from $\tilde{R}(t_f)$.

The second category of examples and simulation refers to the behaviour of the optimal control for both **P1** and **P2** problems. The Fig. 1 and Fig. 2 present the variation of the optimal control and state variables for the **P1** and **P2** problems, respectively.

The simulations were performed for the system (1) and the criterion (2) with the matrices (the adopted example is for a drive system with brushless d.c. motor):

$$A=\begin{bmatrix}-0.04 & 20\\ -3.5 & -19\end{bmatrix}, B=\begin{bmatrix}0\\ 6\end{bmatrix}, S=\begin{bmatrix}1 & 0\\ 0 & 0\end{bmatrix}, Q=\begin{bmatrix}1 & 0\\ 0 & 3\end{bmatrix}, P=1$$

In both figure, the results based on two methods are presented: FPOCI (line curves) and SPOCI (x-mark curve) – see Section 4, point (f). The explanation of the very good coincidence is the fact that the both methods are in essence analytical and only small numerical computing errors occur.

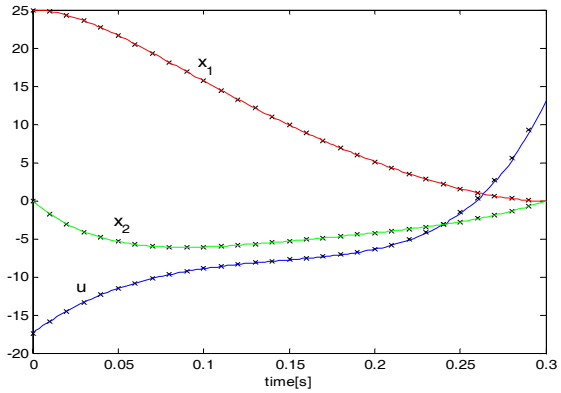


Fig. 1. Behaviour of the optimal system – fixed end-point case (for FPOCI and SPOCI)

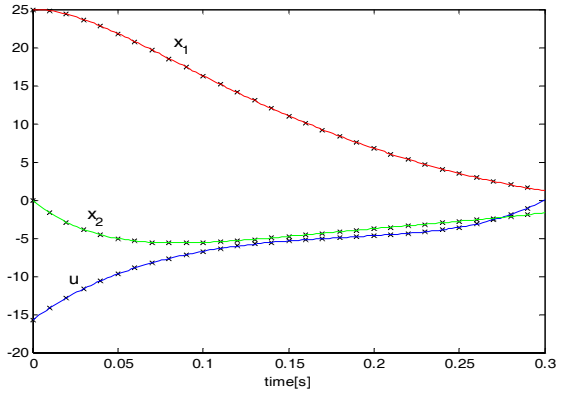


Fig. 2. Behaviour of the optimal system – free end-point case (for FPOCI and SPOCI)

One can assert, as a general remark resulted from the above examples, that the analytical procedures (based on the Theorems 1, 2 and 3) have doubtless advantages in implementation, from the precision and computing time points of view.

VII. Conclusion

- Certain new solutions to Riccati differential equations in the LQ problems are proposed.

- A unified procedure for the problems with free and fixed end-point is approached.

- A Riccati equation with adequate initial condition and the corresponding analytical solution are established for LQ problems with fixed end point.

- Two analytical solutions to Riccati equation for the LQ problem with free end-point are proposed. One of them starts from an initial condition, avoiding the inverse time computing. A recurrent numerical method for direct time computing of the solution to Riccati equation is also indicated.

- All the proposed methods ensure a high precision, a significant decrease of the computing time and an easier implementation by comparison with the existing procedures.

References

[1] R.D. Lorenz, "Future Motor Drive Technology Issues and their Evolution," in *Proc. 12th Int. Power Electronics and Motion Control Conf., EPE-PEMC 2006*, Portoroz, Slovenia, 2006, CD-ROM.

[2] M. Athans and P.L. Falb, *Optimal control*. New York: McGraw Hill, 1966, ch. 5, 6, 9.

[3] B.D.O. Anderson and J.B. Moore, *Optimal control*. New Jersey: Prentice-Hall, 1990, ch. 1, 2.

[4] E.B. Lee and L. Markus, *Foundations of optimal control theory*. New York: John Wiley, 1967.

[5] J. B. Burl, *Linear optimal control*. Menlo Park, CA: Addison-Wesley, 1999, ch. 6.

[6] H. Abou-Kandil, G. Freiling, V. Ionescu, and G. Jank, *Matrix Riccati ecuations in control and systems theory*. Basel: Birkhauser, 2003.

[7] B. Friedland, "On solutions of the Riccati equation in optimization problems," *IEEE Transactions AC*, pp. 303 – 304, June 1967.

[8] P. Brunovsky and J. Komornik, "The Riccati equation solution of the linear quadratic problem with constrained terminal state," *IEEE Transaction AC*, vol. 26, no. 2, pp. 398 – 402, April 1981.

[9] C. Botan, F. Ostafi, and A. Onea, "A solution to the fixed end-point problem," in *Advanced in automatic control*, Ed. Boston: Kluver Academic Press, 2003, pp. 9–20.

[10] I. Rusnak, "Almost analytic representation for the solution of the differential Riccati equation," *IEEE Trans. Automat. Contr.*, vol.33, no.2, pp. 191 – 193, Feb. 1988.

Minimum Energy of Electrical Servo Drive Systems

Corneliu Botan, Florin Ostafi, and Marcel Ratoi

***Abstract*— An optimal control problem for a servo drive system is presented. A convenient method for implementation is indicated. In this direction, a simple algorithm and a cascade based structure are presented. The last one ensures the tracking of the optimal prescribed variation of the angle, which is independent of the load torque.**

I. Introduction

THE optimal control [1], [2] of the electrical drive and servo drive systems allows the reduction of the energy consumption. This type of control is applied for the steady state and / or for transient state. The paper deals with the last case, which is important for the drive systems with frequent changes of the speed and for servo drives, which operate permanently in transient state. In the same time, a good behaviour in the transient period can be achieved.

Besides the energy saving, the reducing of the rated power of the motor is possible in some cases. Indeed, the motor power is chosen from heating considerations and the optimal control offers thus the possibility to adopt a motor with a smaller rated power. In this way, the optimal control ensures not only the energy saving but allows the decrease of the cost, weight and volume. It should be noted that in certain applications the decrease of the weight of a sub-ensemble leads to the diminution of the energy consumption of the all plant. This situation can be mentioned for instance in robotics.

The optimal control of electrical drive and servo drive systems represents an important way for energy saving and it is considered as one of trend in the future in this domain. Although there are numerous studies dedicated to the optimal control of the electrical drive and servo drive systems, for different types of motors, criteria, or used methods, the number of applications is nowadays very small. We appreciate that a cause of the reluctance in this direction is the complexity of the algorithms. However, the methods proposed by authors for different motor types allow the achievement of an easy implementation and therefore, it is useful to apply the optimal control, since the energy losses in the motor windings decrease up to 30% in the electromechanical transient process, by comparison with classical control procedures. These aspects are also interested for servo drive systems, tacking into account that they work very much time in transient state.

The aim of this paper is to present a solution with a simple implementation and to sustain the theoretical aspects with simulation and experimental results.

The paper refers to a linear electrical servo drive system controlled on the basis of the linear quadratic optimal problem. In order to obtain a fast response, a finite final time is imposed.

The system with only one control variable *u(t)* is described by the equations [4], [5], [6]:

$$\begin{aligned} \dot{\theta} &= a_{12}\omega \\ \dot{\omega} &= a_{22}\omega + a_{23}i + cm \\ \dot{i} &= a_{32}\omega + a_{33}i + bu \end{aligned} \quad (1)$$

where θ is the angular displacement of the controlled axis, ω is the motor speed, m is the load torque, $a_{ij}, i, j = 1,2,3$, b and c are parameters of the servo drive system. If a brush d.c. motor is used, i and u are the rotor current and voltage, respectively (the flux is constant). For brushless, synchronous and asynchronous motors, the equations (1) can be adopted with adequate assumptions for a dq reference frame (mainly, the i_d component is constant or very small); in this case, i and u in (1) correspond to the q current and voltage components. The system (1) can be written in the form

$$\dot{x}(t) = Ax(t) + Bu(t) + w(t), \quad (2)$$

where $x(t) = \left[x_1(t) \quad x_2(t) \quad x_3(t)\right]^{\mathrm{T}} = \left[\theta(t) \quad \omega(t) \quad i(t)\right]^{\mathrm{T}}$ is the state vector (T denotes the transposition), *u(t)* is the control variable, and $w^{\mathrm{T}}(t) = [0 \;\; c \;\; 0]^{\mathrm{T}} m(t)$ is the disturbance vector. The matrices in (2) result immediately from (1).

The aim of the optimal control is to obtain a convenient transient response, a small error of the final position and reduced energy consumption. The following criterion is introduced for this purpose:

$$\begin{aligned} J = \frac{1}{2}[s_1\left(\theta(t_f) - \theta_d\right)^2 + s_2\omega^2(t_f)] + \\ + \frac{1}{2}\int_0^{t_f}[q_1(\theta(t) - \theta_d)^2 + q_3 i^2(t) + pu^2(t)]dt. \end{aligned} \quad (3)$$

Manuscript received May 7, 2010. This work was supported by The National Centre for Programs Management from Romania under the research grant SICONA – 12100/2008.

C. Botan is with the Technical University of Iasi, Romania (phone: +40232 278 683; fax: +40232 230751; e-mail: cbotan@ac.tuiasi.ro).

F. Ostafi is with the Technical University of Iasi, Romania (e-mail: fostafi@ac.tuiasi.ro).

M. Ratoi is with the Technical University of Iasi, Romania (e-mail: ratoi@tuiasi.ro)

where t_f is the final time, θ_d is the desired value of the position, and s_1, s_2, q_1, q_3, $p > 0$ are the weight constants. The first term in the criterion (3) penalises the final error $\theta(t_f)$ - θ_d and the final value $\omega(t_f)$, which must be very small. The integral in (3) penalises the transient error, the great value of the control variable and the energy losses in the windings. The performance index (3) is adopted in the optimal control problems with free end-point. For servo drive systems, the problem with fixed end-point is more suitable [7]. In this case, Mayer component lacks in (3), i.e. $s_1 = 0$ and $s_2 = 0$.

The criterion (3) can be written in the general form

$$J = \frac{1}{2}[x(t_f) - x_d]^T S[x(t_f) - x_d] + \\ + \frac{1}{2}\int_0^{t_f} [(x(t) - x_d)^T(t) Q(x(t) - x_d) + u^T(t) P u(t)]dt, \quad (4)$$

with $x \in \mathbf{R^n}$, $u \in \mathbf{R^m}$, $S \geq 0$, $Q \geq 0$, $P > 0$ - matrices of the appropriate dimensions. For our problem, $S=diag(s_1,s_2,0)$ in the free end point case, or $S=0$ in the fixed end-point case, $Q=diag(q_1,0,q_3)$, $P=p$ and $x_d = \begin{bmatrix}\theta_d & 0 & 0\end{bmatrix}^T$ is the desired vector.

The optimal control problem refers to the system (2) and criterion (4). The solution to this problem is [1]

$$u(t) = -P^{-1}B^T \tilde{R}(t) x(t), \quad (5)$$

where $\tilde{R}(t)$ is a positive defined matrix, obtained as a solution to a Riccati matrix differential equation. Since this matrix is time-variant, the optimal controller is also time-variant. This fact implies a difficult implementation, augmented by the fact that the Riccati equation have to be integrated in inverse time, starting from the final condition $\tilde{R}(t) = S$ (in the free end-point problems), or has a very complicated form in the fixed end-point case. In order to avoid these difficulties, a new solution is proposed, which allows to use a time-invariant controller and a corrective component depending on the initial state $x(0)$. This controller ensures the same solution as one offered by (5). Moreover, the optimal control from the energetic point of view of the drive and servo drive systems implies to beforehand know the shape of the load torque and to estimate its magnitude at the beginning of the optimization interval [7]. The paper presents a method which avoids the estimation of the load torque and this fact significantly simplifies the implementation.

II. Main Results

In the sequel it is presented only the problem with fixed end point, starting from some results obtained by authors in [8]. From the canonical Hamiltonian equations [1], it results

$$\begin{aligned} \dot{\tilde{x}}(t) &= A\tilde{x}(t) - N\lambda(t) + \tilde{w}(t), \\ \dot{\lambda}(t) &= -Q\tilde{x}(t) - A^T\lambda(t), \end{aligned} \quad (6)$$

where $\tilde{x}(t) = x(t) - x_d$, $\tilde{w}(t) = Ax_d + w(t)$, $N = BP^{-1}B^T$.

The system (6) with terminal conditions $\tilde{x}(t_0)=\tilde{x}^0$ and $\tilde{x}(t_f)=0$ can be solved, but the optimal control has a complicated form, especially for the feedback control, when the inverse of a time variant matrix occurs. This form is difficult to use in real time computing. In order to avoid this complication, the main idea is to perform a change of variables

$$[\tilde{x} \;\; \lambda] = U[\tilde{x} \;\; v], \quad (7)$$

where $U = \begin{bmatrix} I_n & 0 \\ R & I_n \end{bmatrix}$, with R - solution to the corresponding algebraic matrix Riccati equation and I_n - n x n identity matrix. Finally, one can find that the optimal control can be written in the form

$$u(t) = u_f(t) + u_S(t), \quad (8)$$

where

$$u_f(t) = -P^{-1}B^T R\tilde{x}(t) \quad (9)$$

is the feedback component (this component is identical with one obtained in the similar linear quadratic problem but with infinite final time) and

$$u_s(t) = -P^{-1}B^T v(t) \quad (10)$$

is a supplementary component, depending on the vector $v(t)$, given by

$$v(t) = \Phi(t,t_0)v(t_0) + \beta_2(t,t_0), \quad (11)$$

with

$$v(t_0) = (L_1 - R)x^0 - \Gamma_{12}^{-1}(t_f,t_0)\alpha_1(t_f,t_0). \quad (12)$$

In the last two equations,

$$\begin{aligned} &\beta_2(t,t_0) = \int_{t_0}^{t} -\Phi(t,\tau)R\tilde{w}(\tau)d\tau, \\ &\alpha_1(t_f,t_0) = \int_{t_0}^{t_f} \Gamma_{21}(t,\tau)\tilde{w}(\tau)d\tau, \\ &L_1 = -\Gamma_{12}^{-1}(t_f,t_0)\Gamma_{11}(t_f,t_0), \quad (13) \\ &\Gamma_{12}(t,t_c) = \int_{t}^{t_c} \Psi(t,\tau)N\phi(\tau,t_c)d\tau, \\ &\Gamma_{11} = \Psi - \Gamma_{12}R, \; \Gamma_{21} = R\Psi - R\Gamma_{12}R - \Phi R, \\ &\Gamma_{22} = \Phi + R\Gamma_{12} \end{aligned}$$

and $\Psi(.)$ and $\Phi(.)$ are the transition matrices for $(A-NR)$ and $-(A-NR)^T$, respectively. It is proved that the inverse

matrix $\Gamma_{12}^{-1}(t_f, t_0)$ exists if the system (1) is completely controllable [8].

Remark1: The above formulae are quite complicated, but the most part of the computing is performed off-line, in the design stage of the controller. The real-time computing implies only the computing of the optimal control $u(t)$ and this imposes to establish a usual feedback component $u_f(t)$ and a supplementary component $u_s(t)$. This last component contains only two time variant elements – the matrix $\Phi(t, t_0)$ and the vector $\beta_2(t,t_f)$. These variant elements can be recurrently computed and thus, the supplementary component can be easy established and the global computing effort for $u(t)$ is not too complicated by comparison with the case of the usual state feedback control.

Remark 2: The computation of optimal control $u(t)$ implies the knowledge of the vector $\alpha_1(t_0,t_f)$. This supposes that the exogenous variables are beforehand known on the optimization interval $[t_0,t_f]$ and this implies to know the value of the disturbance $m(t)$ on this interval. Only the knowledge of the shape of $m(t)$ is sufficient (for instance $m(t)$ = constant) if the disturbance torque is measured or estimated at the initial moment. A disturbance observer can be introduced in the controller structure on this purpose.

Remark 3: The optimal control is ensured on the interval $[t_0,t_f]$. In many cases it is desired to maintain the desired values of the variables for $t > t_f$. It is necessary in this case to change the control law, for instance to introduce a usual linear feedback control.

The optimal control of the servo drive system can be with fixed or free final time t_f (the initial moment t_0 is fixed). In the last case, the optimal value of the final time can be established from the corresponding transversality condition. This problem is attractive, but it has two disadvantages:
- the shape of the load torque $m(t)$ must be beforehand known and its magnitude must be estimated in the initial moment, as it was mentioned in the Remark 2;
- the optimal value of t_f has big values for small m and a limit value for t_f has to be imposed.

In the problem with fixed final time, the estimation of the load torque can be avoided, as it will be indicated in the sequel. Various condition can be chosen in this case for establish of t_f. For instance, a mean value for speed

$$(\theta(t_f) - \theta(t_0)) / (t_f - t_o) \ ,$$

or a value for the acceleration

$$(\theta(t_f) - \theta(t_0)) / (t_f - t_o)^2$$

can be imposed.

A short value imposed for the time transfer leads to an increased value for current and therefore for the energy consumption.

The above remarks assert that the implementation based on the previous relations can be easy performed. But the implementation can be still significantly simplified using the remark presented below. Numerous performed tests showed that the speed variation $\omega(t)$ on the interval $[t_0,t_f]$ does not depend on the load torque $m(t)$. This fact can be explained starting from the first equation (1), which leads to

$$a_{12} \int_{t_0}^{t_f} \omega(t)dt = \theta(t_f) - \theta(t_0) \qquad (14)$$

Since the terminal values of the angular displacement are fixed, the integral in (14) has a constant value. The optimal speed has a variation with a certain form (which results from the above relations) and the imposed terminal values $\omega(t_0)$ and $\omega(t_f)$ (usually they are zero); thus the constant value of the integral (14) is achieved only for unchanged $\omega(t)$, $t \in [t_0, t_f]$ even if $m(t)$ has different values. For instance, if the load torque increases, the motor optimal current increases too, so that the optimal variation of the speed is obtained. Further, if the variation of the optimal speed is independent on *m(t)*, it results that the angle optimal variation $\theta^*(t)$ is also independent on *m(t).*

This important remark suggests an easy way for implementation of the optimal controller, using the prescribed optimal variation for speed, or of the angle. Since these variations are independent on $m(t)$, they can be computed from the above equations with the important simplification $\tilde{w} = 0$ (i.e. $\beta_2 = 0$ and $\alpha_1 = 0$ in (11) and (12)). In the sequel only the ideal variation of the angle will be considered.

The structure of the optimal servo drive system is presented in Fig.1, where θ_d is the imposed value of the angular displacement, $\theta^*(t)$ is the prescribed ideal optimal variation for angle, OC is a block which compute the ideal trajectory $\theta^*(t)$, C1 and C2 are the controllers and M is the motor. Different type of controllers can be chosen - for instance a predictive controller, an optimal tracking type one. The performed tests have indicated that simple PI controllers ensure a good concordance between θ and θ^* and between ω and ω^*, respectively, as it is indicated in the Fig. 2.

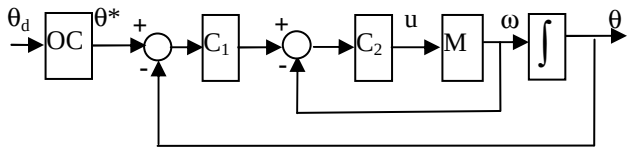


Fig. 1. Suboptimal system structure.

For $t>t_f$, the prescribed value for θ must be maintained to the desired value θ_d, so that the computed variation of $\theta^*(t)$ must be maintained to the value θ_d for $t > t_f$. This condition is ensured by the block OC.

The system implemented by this way is suboptimal, but the difference between this system and an optimal one is very small. On the other hand, the implementation is very simple in comparison with other solution for optimal control. Of course, the concordance between the desired variation $\theta^*(t)$ and the obtained one is affected if the current limitations interfere, but the behaviour of the system is acceptable.

The proposed structure offers three important advantages:

- the estimation of the load torque is avoided;
- the optimal control can be implemented even in the case when the shape of the load torque is not beforehand known;
- the optimal ideal trajectory $\theta^*(t)$ must not be computed in real time at every sampling period and can be obtained in a very short time at the beginning of the optimization interval.

III. Simulation and Experimental Results

The figures 2, 3, 4 and 5 present some simulation results. The performance index value J and the copper energy losses E were computed in each case.

A comparison between the ideal optimal variables (dotted curves) and ones obtained in the suboptimal system with the above structure (continuous line curves) is indicated in Fig. 2, for m = 0.25m_n, where m_n is the rated torque of the motor. The unavoidable delay of the angle and of the speed especially in the initial period forces the control variable and the servomotor is obliged to increase its current and therefore, the performance index and the energy losses increase in comparison with the ideal optimal control. However, the concordance between the speeds and especially the angles is acceptable. The increase of the performance index is about 35%. From practical point of view, it is more interesting the variation of the energy losses; this increase with about 18%.

Fig. 3 presents the behaviour of the optimal servo drive system for different load torques (m_1 = 0.25m_n, m_2 = m_n): the voltage and current increase with the load torque, but the speed and the angular displacement are practically the same. This property remains valid even the load torque has a step variation in the transient period (from m_1 = 0.25m_n, to m_2 = m_n at the moment 0.5s), as it is indicated in the Fig. 4.

One can remark that a significant modification of the load torque has a small influence on the energy losses. This is a general aspect: the most part of the energy losses corresponds to the consumption for the acceleration and deceleration of the system.

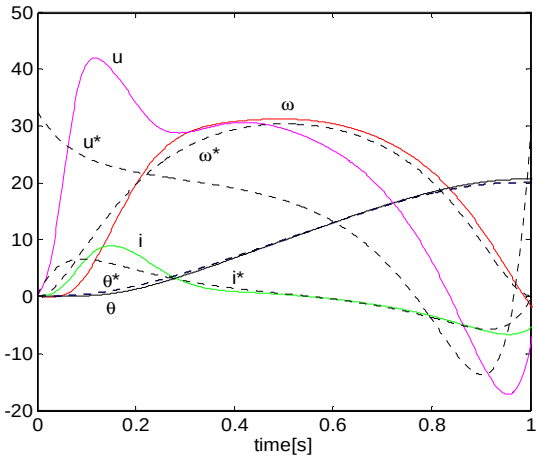


Fig. 2. Variations of the control and state variables - ideal and real cases.

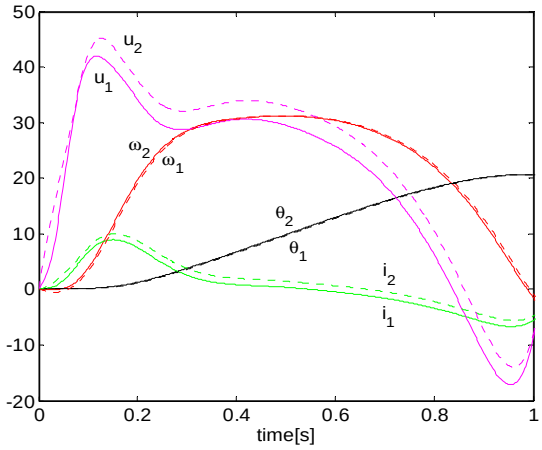


Fig. 3. The behaviour of the suboptimal system for different load torque. J_1= 982, J_2= 1130, E_1= 41.5J, E_2= 48J.

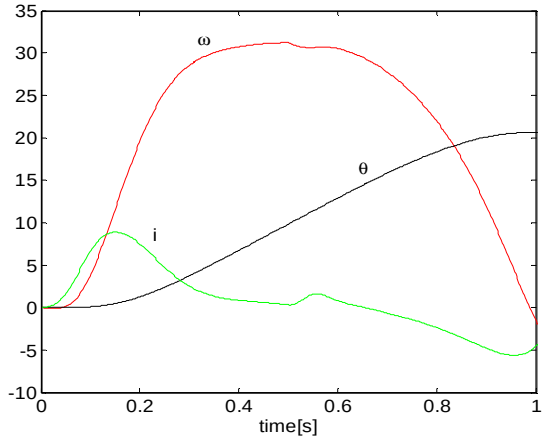


Fig. 4. The behaviour of the suboptimal system in the case of a step variation of the load torque at t=0.5s.

The influence of the decrease of the duration of the optimal transient process can be observed in Fig. 5 in comparison with Fig. 2.

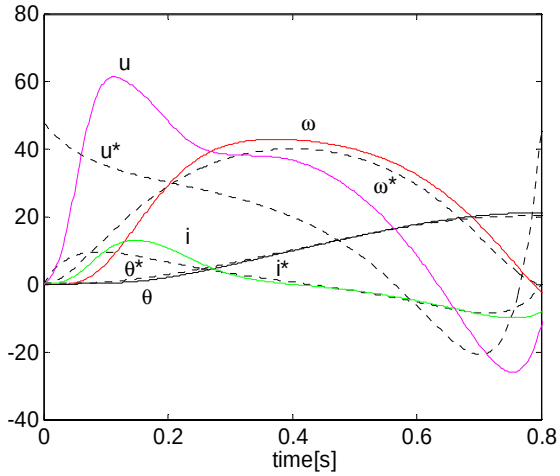


Fig. 5. The behaviour of the optimal and suboptimal system for t_f = 0.8s.

A decrease of the final time from 1s to 0.8s leads to an increase of the performance index with about 20% and to a significant increase of the energy losses, which are about two times greater.

Experimental results referring to an optimal servo drive system with a PMSM [9], [10] (with rated data 2kW, 330V, 4.4A, 4500 rpm, 4Nm, 225 Hz and inertia 0.0072 Nms2/rad) for a 2 Nm load torque are presented in Fig. 6.

A vector control technique was applied and dSPACE tools were used for implementation and tests.

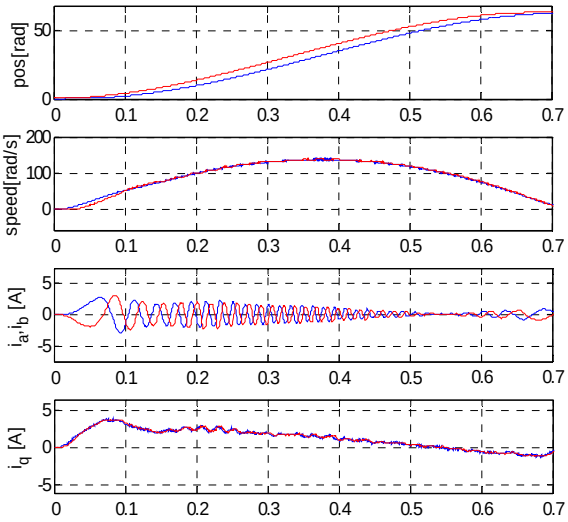


Fig. 6. The behaviour of the suboptimal servo drive system – experimental results.

The ideal optimal values (red colour) and real suboptimal values (blue colour) for displacement, speed and current components (in a-b stator reference frame and d-q reference frame) are indicated. The shapes of these variables are similar with ones obtained in the simulation tests. A reduced difference between ideal optimal variations and real ones can be remarked.

IV. Conclusions

The optimal control of the servo drive systems is studied. Two important simplification of the implementation are indicated:

- An algorithm which can be easy implemented so that the global computing effort for the optimal control does not rise significantly above the usual state feedback control.
- A cascade based structure, which ensures the desired optimal variation of the angle. This structure uses the proven property referring to the fact that the optimal speed and optimal angle variations are independent on the load torque. The proposed structure avoids the necessity of the estimation of the load torque.

References

[1] B.D.O. Anderson and J.B. Moore, *Optimal control*. New Jersey: Prentice-Hall, 1990.

[2] J. B. Burl, *Linear optimal control*. Menlo Park, CA: Addison-Wesley, 1999.

[3] R.D. Lorenz, "Future Motor Drive Technology Issues and their Evolution," in *Proc. 12th Int. Power Electronics and Motion Control Conf., EPE-PEMC 2006*, Portoroz, Slovenia, 2006, CD-ROM.

[4] I. Boldea and S.A. Nasar, *Electric Drives*. CRC Press Taylor and Francis, 2006

[5] W. Leonhard, *Control of electric Drives*. Springer Verlag, 2001.

[6] C. Mademlis, N. Margaris, "Loss Minimization in Vector-Controlled Interior Permanent-Magnet Synchronous Motor Drives," *IEEE Trans. on Ind. Electronics*, Vol. 49, No. 6, pp. 1344-1347, 2002.

[7] C. Botan; V. Horga, and M. Ratoi, "Optimal Control of the Electrical Drives with Permanent Magnet Synchronous Machine," in *Proc. of 19th IEEE, IES International Symposium on Power Electronics, Electrical Drives, Automation and Motion, Speedam 2008*, Ischia, Italy, June 2008, CD-ROM.

[8] C. Botan, F. Ostafi, and A. Onea, "A solution to the fixed end-point problem," in *Advanced in automatic control*, Ed. Boston: Kluver Academic Press, 2003, pp. 9–20.

[9] C. Cavallaro, A.O. DiTommaso, R. Miceli, A. Raciti, G.R. Galluzzo, and M. Trapanese, "Efficiency Enhancement of Permanent-Magnet Synchronous Motor Drives by Online Loss Minimization Approaches," *IEEE Trans. on Ind. Electronics*, vol. 52, 4, pp. 1153-1160, Aug. 2002.

[10] K. Kurihara and M.A. Rahman, "High-efficiency line-start interior permanent-magnet synchronous motors," *IEEE Trans. on Ind. Applic.*, vol.40, 3, pp. 789- 796, May-June 2004.

Self-adaptable Security Architecture for Power-aware Embedded Systems

N. Botezatu, V. Manta, and A. Stan

***Abstract*—Securing embedded systems is a challenging and important research topic due to limited computational and memory resources. Moreover, battery powered embedded systems introduce power constraints that make the problem of deploying security more difficult. This problem may be addressed by optimizing the trade-off between minimizing energy consumption and maintaining a proper security level.**

This paper proposes a self-adaptable security architecture for embedded systems. The proposed method points out a conceptual blueprint needed for the implementation of such self-adaptable mechanisms. An example case study is described in order to better understand how an adaptable security mechanism can be implemented, also pointing out the effect on energy consumption.

I. Introduction

MANY embedded systems interact with the real world. Form microwave ovens to MP3 players and photo cameras to cell phones, embedded computing has a huge impact on our everyday life. The research and technological advances that made all this possible, determined an increase in the system's complexity, also spurring a downside: the raise of security costs. The development of new, faster, more feature rich embedded systems has boosted the emergence and improvement of security methods in parallel to ranges of new and more power types of attacks.

While security method's and protocol's designers address the security problem from a functional perspective (the traditional way), all embedded systems are constrained, in a way or another, by the available internal resources and the external environment dynamics. Also, security is often perceived by the system designers as an additional or optional feature. In fact, security should be considered in the design process along costs, performance and power. The problems arising from the aforementioned, between necessities and capabilities, are described and classified as a set of gaps [1].

One of these gaps, the *battery gap*, emphasizes the overheads in energy consumption introduced by supporting security on battery powered devices. This is due to the slow pace of advancement in battery technology that has not kept up with the progress in processing capabilities (and consequently energy consumption). Therefore, the primary challenge in providing security in battery powered embedded devices is to minimize power consumption and maximize security. Due to the conflicting nature of the two elements, there is an intrinsic need to understand the relation between energy consumption and security parameters.

One way to approach this problem is by adaptable security. By changing in some automated or semi-automated way, the security mechanisms respond to internal or external system events, consequently causing a variation in energy consumption. The link between this variation and the dynamics of adaptable security must be determined in order to assess and optimize the energy consumption level. As the paper name implies, we propose a conceptual design for the infrastructure needed for adapting security, also providing a fundamental operational structure. We also suggest a functional description of the architecture's requirements by presenting an implementation example.

The paper is organized as follows. Section II briefly outlines the related work presented in the literature. Section III explains the architecture and describes its components, whereas in Section IV we present an implementation example. Section V highlights some perspectives for further research.

II. Related Work

We focus on giving an overview on papers that cover the aspects of adaptable security or energy consumption, due to the lack of papers covering the two topics at theoretical level. We also centered over the papers presenting these aspect at a more practical level.

Reference [2] proposes some guidelines for designing a standardized Adaptive Security Infrastructure (ASI). The problems raised by this concept are presented in a structured way and include: the components of an ASI, the principles and issues of formalization and the term of *security policy* described as a system specification. Also, the author proposes some broad research directions, covering the discussed topics.

In [3] the self-adaptable security is addressed at system level, centered on the concept of security policy. The authors propose a domain-independent methodology for system security adaptability. The security policy adaptability scheme allows the system to keep a constant security level even when certain stimuli influence it.

Reference [4] presents an overview of the current trends in energy-efficient computing. The strengths and weaknesses of present power management are discussed,

Manuscript received April 28, 2010.

N. Botezatu is a PhD student at "Gh. Asachi" Technical University of Iasi, Faculty of Automatic Control and Computer Engineering (e-mail: nbotezatu@cs.tuiasi.ro)

V. Manta is with the "Gh. Asachi" Technical University of Iasi, Faculty of Automatic Control and Computer Engineering (e-mail: vmanta@cs.tuiasi.ro)

A. Stan is with the "Gh. Asachi" Technical University of Iasi, Faculty of Automatic Control and Computer Engineering (e-mail: andreis@cs.tuiasi.ro)

considering the ACPI standard as a case study. Also, the author's vision on energy consumption reduction is presented as an optimization problem. Three stages are outlined for approaching a solution on energy-efficiency: constructing a power model; determining the performance requirements of tasks or the workload; implementing means of deciding an energy-efficient configuration of the hardware at all times while operating.

In [5], the authors acknowledge the importance of constructing power-aware applications. The paper also briefly surveys some of the most recent directions in supporting power-efficiency for battery powered devices. Furthermore, a power efficiency framework is proposed which addresses both power consumption measurements and high level power efficiency metrics in a unified way, thus providing real-time feedback to applications.

Next, in [6] the authors, using real-life experimentation based profiling, measure and model the power consumption of some cryptographic algorithms. Also, they propose a way for minimizing vulnerability subject to power constraints. The vulnerability metric is defined as a quantity dependent on the success probability of a cryptanalysis attack. This metric is used as the objective function to formulate two optimization problems. Moreover, the paper proposes algorithms that solve these problems, giving optimal power consumption and security vulnerability levels.

The research presented in [7] deals with the implementation of security primitives on reconfigurable hardware. The reconfiguration thresholds are based on external events that point out any incoming attacks. The paper also presents a functional description for the monitoring blocks used to detect attacks and for the SSC (system security controller) and SPC (security primitive controller) that are used in the reconfiguration process.

III. Self-adaptable Architecture

In this section we present our vision on the security self-adaptable method. As shown in Fig. 1 the architecture is based on three functional blocks: *Sensing*, *Analysis* and *Enforcement*. The data sets required as input for the blocks are represented by the *System status*, *System descriptors* and *System goals*.

The workflow of the architecture is the following: when the Sensing block detects a change in the parameters defined in the data sets, it signals the Analysis block to determine what changes in security must be made in order to match some requirements defined in the System goals. Last, the Enforcement block applies changes to the system. The following subsections present the six new concepts.

A. System status

The system status may be viewed uniquely defined by a set of specific parameters. These parameters can be grouped as internal or external, where the internal ones are specific to the systems hardware configuration (e.g. what functional blocks does the system have that can influence the status) and to the software applications that can run on the system (e.g. the application's requirements concerning different hardware properties of the system). As for the external parameters, they are represented by the systems external conditions as the user of the system (e.g. tasks that are defined or ran by the user) and the environment in which the system is operating (e.g. sunlight condition for systems with photovoltaic power cells). All this data is acquired through the use of monitoring devices and sensors.

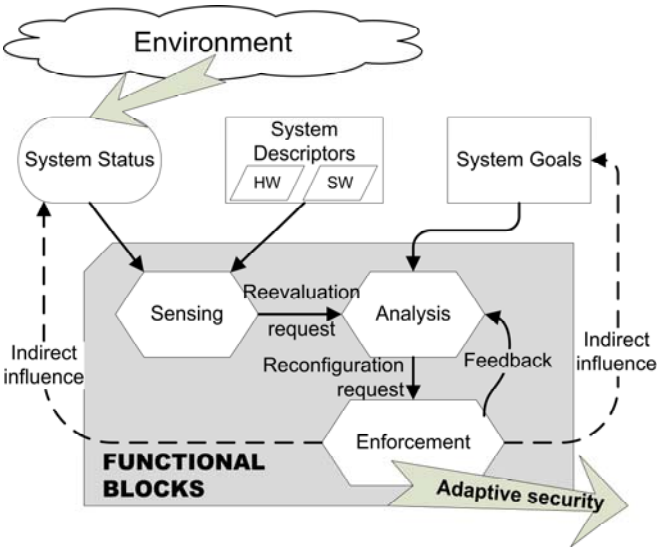


Fig. 1. Adaptive security architecture blocks and related relations

In general, for an application that needs to know the system status there must be defined a list of parameters that can influence the status of the system in the current context of operation. All the values of the parameters, the valid values transitions that are important for the applications, all the connections between the parameters and their values that are valid at system runtime must be analyzed. More, in order to have accurate information about the system, a comprehensive set of monitored parameters must be defined.

Also, the rate at which the system status is sampled is important. Different parameters may be sampled at different rates based on the speed of their temporal variation. The sampling rate may vary for the same parameter for specific intervals of its values (e.g. the sampling rate for the voltage of a NiMH cell can be raised for a Depth of Discharge (DOD) over 80%, due to the increase in voltage decline rate).

Specific to the current architecture are the following properties that outline the system status:

- Current security level for every running application;
- The available energy of the system, expressed conveniently as a combination of drawn current, battery capacity, discharge rate, nominal power etc.

An example of selected parameters is presented in Section IV.

B. System descriptors

The system descriptors are represented by a collection of data that describe the hardware and software components of the system. The data is organized in a hierarchical manner, based on a class collection that describes and groups the

systems elements with respect to their energy and security properties. The base model for this branching hierarchy is presented in Fig. 2. The first three levels of the class collection make a broad classification of the system components, as from level four onwards specific classes may be derived in order to describe the system components (e.g. from the energy suppliers class we can derive new classes to describe batteries, photovoltaic cells or other types of supplies).

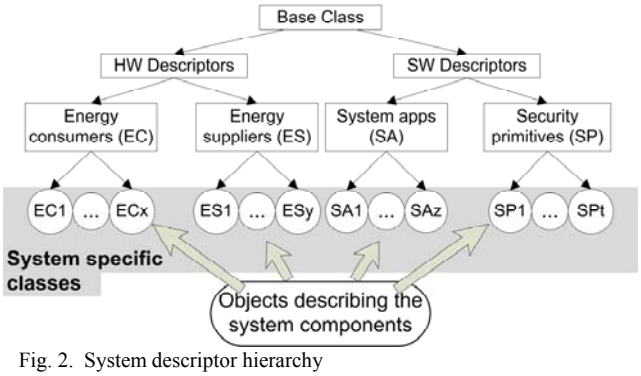


Fig. 2. System descriptor hierarchy

The properties of the objects that describe the hardware components depend on the operating modes used during application execution. A complete profile for every system element is obtained by describing the energy properties for all the operating modes, based on usage patterns. For example, if we consider a LCD backlight, the energy consumption must be determined and described for every backlight intensity mode available. By defining these properties, a connection is established between energy consumption, workload and operating mode.

For the software components, security properties must be considered. Also, the objects describing the software components are used to determine their energy, workload and security footprints. In order to achieve this, the properties for the software components also specify the hardware resources used and how they are used.

Concerning the security primitives, their properties are described for all the usage options available (e.g. different key lengths, variable number of encryption rounds) as a usage cost (e.g. energy consumption, hardware needs). Also, for the security primitives the throughput must be considered as a property for every usage option.

C. System goals

The system goals are determined based on the correlations between the energy resources of the system and its security demands. *Ravi et al.* in [1] have identified several gaps regarding the capabilities and demands of current embedded systems. In the context of the current architecture, the system goals are to be determined taking into account two of the proposed gaps: the battery and the flexibility gaps. The first one highlights that the current energy consumption overheads of supporting security on battery powered embedded devices are very high, whereas the second one emphasizes an embedded system's need to run a large and diverse set of security primitives and methods. Also, the authors state the existence of a processing gap because the current embedded architectures cannot keep up with the rising computational demands of new and more complex security algorithms. To sum up, the system goals must be described as a maximization of operating time while maintaining the requested security levels for the system. In order to better understand this concept, the following example is considered: an embedded system has two running applications A1 and A2. A1 processes two types of data, the first one requiring a security level of S1 and the second one S2. A2 needs a security level of S3. The system, using the built in security primitives can provide a maximum security level of Smax, where Smax=S3>S1>S2. So, if the system would consider a constant security level for all the applications, the extra security delivered for application A1 would consume a part of the systems energy to no purpose. In order to decrease the used energy, the security level would have to be adapted to the application's needs, this process being transparent to the user. Moreover, if the security demands of an application varies at runtime, based on some environment variations (e.g. the transition from an unsecure wireless network to a secure one), this fact must also be considered when determining the system goals.

D. Sensing block

The sensing block monitors the state of the system. The states of the system are defined as the elements of a finite set $S = \{s_1, s_2, \ldots, s_n\}$, those states being determined based on the combinations and correlations between the environment parameters (System status) and the system descriptors. When a state transition occurs this block decides if a request to reevaluate the security is sent to the Analysis block. Its purpose is to determine the state of the system and to filter out the transitions that are not relevant in the context of energy consumption optimization. On a time basis, the sensing block samples the environment parameters and system descriptors and evaluates an *f* function which has the aforementioned parameters, thus determining the system state.

In order to filter out the transitions not relevant to the system a prediction module is used. Its role is to express a probability for any status transition. This means that following a state transition, the system must remain in the new state long enough to account for the extra energy used by the reevaluation of security and the related security setup cost (e.g. generation of new encryption keys).

The mechanism of the prediction module is explained hereinafter. First, all the transitions have a default probability value. In order to evaluate the validity of a transition, its probability is compared to a threshold value. If the test is passed (initially all transitions are considered valid) and a reevaluation request is sent to the Analysis block, the energy level used by the analysis step is assessed. Also, the energy consumed by the system, for enforcing security, until the next state transition is determined. Based on these results the probability is modified by applying a correction factor accordingly.

E. Analysis block

This block receives security reevaluation requests from the Sensing block, based on system states transitions. Depending on the goals and the new state of the system, it decides what security elements should be modified consequently.

When designing this module, its response time should be taken into account because it has the potential to be the most expensive of the entire architecture. Moreover, the extra delay introduced by adapting the security primitives used may have a negative impact on security (e.g. unsafe operation from the state change until the enforcement of the new security parameters) as well as on the energy consumption (e.g. the overhead introduced by the analysis), resulting in a performance decline.

After the analysis process, if there are available a number of possible security adaptations sets, equivalent for the system goals, the timing factor must be considered. For a better understanding, let us account the following example: two primitives could be used for securing an application. The first one has a high setup cost (and a subsequent higher enforcement delay) and a low usage cost (e.g. in a mJ/bit metric), whereas the second one has a smaller setup cost and a higher usage one. In this case, the analysis block should select to enforce the second set of security methods as first option.

F. Enforcement block

This block receives security reconfiguration requests from the analysis module. It tries to apply the changes to the system, the result (success or fail) being feed-backed to the analysis module. If the reconfiguration request cannot be satisfied, the analysis module must propose other security alternatives.

It also offers an interface to the OS (if any) and the underlying system applications, acting as a bridge between the system's security primitives and methods and the running applications.

Dictating security changes can also indirectly influence the system's status and goals. For example, one of the status parameters is the battery voltage or one of the goals is based on the same parameter. Knowing that a change in the system's load (sink/source current), based on security adjustment, determines a voltage value variation, a threshold cross could determine a false response. Therefore, care must be taken in order to minimize the negative impact of these influences.

IV. Implementation Example

We propose a simple implementation example in order to show how the functions of the fore mentioned blocks can be described, without presenting any implementation details. First, the software and hardware parts of the system will be described grouped after the base classes illustrated in the previous section.

The system we built runs two applications: the first one resides permanently and has two operating modes; the two modes alternate on a time basis, the application spending 5 minutes in the first mode (referred to as A1m1), after that spending 10 minutes in the second mode (referred to as A1m2). The second application runs for 30 minutes, only after an external event arises (e.g. in our hardware setup the event was the push of a button).

As for the security primitives, there are two symmetric key ciphers implemented on the system: Data Encryption Standard (DES) and Fast Data Encipherment Algorithm (FEAL) [8]. The DES algorithm can be used with 16 encryption round and with 8 encryption rounds (referred as DES16 and DES8), while the FEAL cipher is used in the default form.

TABLE I
System States and Security Requirements

Running Application	Battery DoD	State	Security Requirements
A1m1	-	S1	A1 → DES16
A1m2	-	S2	A1 → DES
A1m1 & A2	< 50%	S3	A1 → DES16 A2 → DES8
	≥ 50%	S4	A1 → DES16 A2 → FEAL
A1m2 & A2	< 50%	S5	A1 → DES A2 → DES8
	≥ 50%	S6	A1 → DES A2 → FEAL

TABLE II
Encryption Algorithms Characteristics

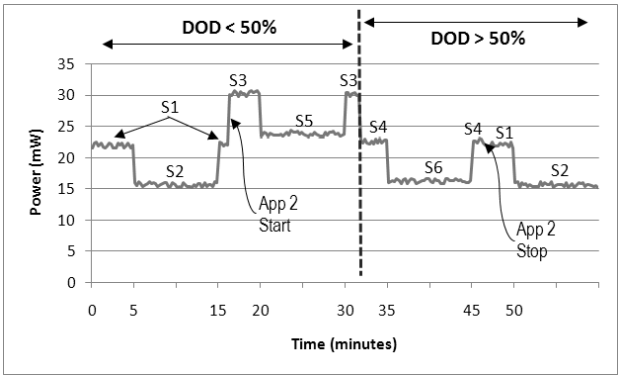
Algorithm	Encryption time for 64 bit data (μsec)	MCU workload (%)	Power (mW)
DES16	1380	47,57	22,05
DES8	775	26,72	15,67
FEAL	53	1,83	8,05

Now, the security requirements for the applications can be defined. The first application when operating in mode 1 needs to be secured using DES16 and when operating in mode 2 it has a more general requirement, needing DES. This means that either DES16 or DES8 can be used. The second application considers FEAL to maintain a sufficient security level and DES8 to provide good security. We will see later how this requirement is translated into measurable properties.

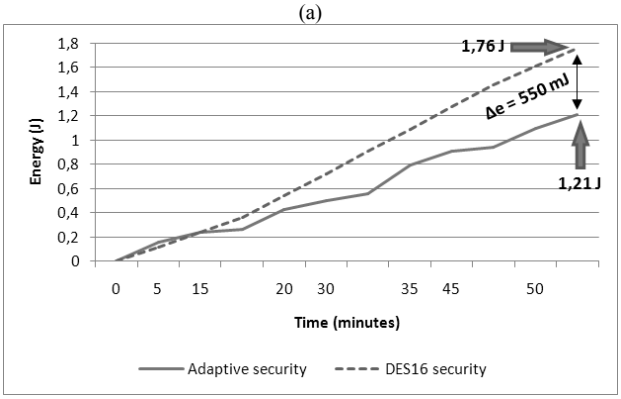
From the hardware perspective, we only consider the MCU as energy consumer, due to the variation of energy consumption in run and sleep modes. The systems battery is considered as an energy supplier class object, its sole property being the initial energy level. The chosen metric for estimating the systems energy status is the battery's depth of discharge (DOD), expressed in percents. The DOD is approximated based on the battery capacity, the system power consumption and the system's components operating time vs. modes. The hardware platform used is based on an ARM Cortex-M3 MCU and on rechargeable NiMH batteries (we used batteries with a capacity of 2000 mAh) [9].

We have chosen two parameters with dynamic variation at runtime (system status) that influence the system's state: the DOD and a parameter that shows what applications are running and in what mode (if applicable). Based on this information, Table I shows the system states that can be

identified by the sensing block. For the parameter showing the DOD level we considered a threshold at 50% value. This threshold is used to influence the selection of security primitive used for application 2: below 50% DES8 is used and FEAL above this value.



(a)



(b)

Fig. 3 (a) Power level and (b) energy consumption variation at runtime

The system goals are the following:

- to provide the security level required by the applications – security goal
- to use the security methods with the lowest energy consumption level – energy goal

The Analysis block functions are constant, because the system goals do not change over time. This means that the same security primitives are constantly chosen for every state based on a look-up table, also shown in Table I. The security goal is satisfied by providing the desired security for every application as illustrated in the System descriptors. The energy goal is met for states S2, S5 and S6, when application 1 is operating in mode 2 and its security requirements are satisfied by DES. The system chooses to use DES8 due to its lower energy consumption.

The Enforcement block matches the security algorithms, indicated by the Analysis block, with the applications data in order to encrypt them.

Concerning the operation of the system, when the conditions of state transition are met (the variation of DOD over the 50% threshold or a change in the application's operating mode) the algorithms for securing the system are selected. Based on the system specifications, which were randomly selected, the process of securing the running applications consists of the encryption of 80 bytes of data, for every application, in a time window of 29 milliseconds. The remaining idle time, after all the data bytes are encrypted, is spent by the MCU in a low-power consumption sleep mode. The time, workload and power consumption characteristics for the security primitives, relative to the system's specifications, are presented in Table II [9], [10].

Fig. 3(a) presents the variation of the power consumed by the system over a period of 60 minutes. The system passes through all possible states, the operating period of application 2 being asynchronous with the mode change of application 1. In the first 16 minutes and 15 seconds only application 1 is running on the system, alternating it's mode of operation two times. After that application 2 also start, thus implying the state change from S1 to S3. After another 15 and a half minutes, the value of the battery's DOD passes the 50% threshold causing a transition from S3 to S4. The change in the security requirement for application 2, from DES8 to FEAL, brings a significant variation to the consumed power level. When application 2 terminates execution, due to the small/low consumption level of the FEAL primitive, the variation of the power level has lower amplitude.

The energy consumed for the operating period is shown in Fig. 3(b) If both applications are secured with the safest available primitive, meaning DES16 [6], and the variations in the battery's DOD level or application operating modes are not considered, an increase of 550 mJ in the consumed energy level can be observed. In the current context, this method for providing uniform security at system level may be considered to be the simplest available, even the safest available, but is certainly not the cheapest available. Therefore, by using this adaptive security method, a decrease of roughly 31% in energy consumption is obtained at runtime.

This example shows that even when the security is adapted to the system's need in a simple manner, important energy savings can be achieved.

V. Conclusion and Further Work

We have presented our vision on how to address security self-adaptability in order to minimize energy consumption. The proposed methodology points out the main functionality needed for the implementation of such self-adaptable mechanisms, along the interactions within the system and with the environment. An example case study is described, in order to better understand how an adaptable security mechanism can be implemented, also pointing out the effect on energy consumption.

Further work implies the implementation of the self-adaptable security method on different embedded architectures. This involves enhancements for the system's status, descriptors and goals entities in order to define them in a more consistent and homogeneous way. Also, different implementations for the functional blocks will be investigated. The evaluation of the blocks under real life scenarios is important in order to appreciate the relation

between the required system performance and the used resources.

REFERENCES

[1] S. Ravi, P. Kocher and S. Hattangady, "Security in Embedded Systems: Design Challenges", *ACM Transactions on Embedded Computing Systems*, vol. 2, no. 3, pp. 461–491, Aug. 2004

[2] L. Marcus, "Introduction to Logical Foundations of an Adaptive Security Infrastructure", presented at the Workshop on Logical Foundations of an Adaptive Security Infrastructure, Turku, Finland, July 12–13, 2004

[3] A. Ferrante, A. V. Taddeo, M. Sami, F. Mantovani and J. Fridkins, "Self-adaptive Security at Application Level: a Proposal", in *Proceedings of the 4th Workshop on Embedded Systems Security*, Grenoble, France, 2009, Paper 4

[4] D. Brown and C. Reams, "Toward Energy-Efficient Computing", in *Communications of the ACM*, vol. 53, no. 3, pp. 50–58, March 2010

[5] D. Tudor and M. Marcu, "Designing a Power Efficiency Framework for Battery Powered Systems" in *Proceedings of SYSTOR 2009: The Israeli Experimental Systems Conference*, Haifa, Israel, 2009, Paper 5

[6] R. Chandramouli, S. Bapatla and K. P. Subbalakshmi, "Battery Power-Aware Encryption", in *ACM Transactions on Information and System Security*, vol. 9, no. 2, pp. 162–180, May 2006

[7] G. Gogniat, T. Wolf and W. Burleson, "Reconfigurable Security Architecture for Embedded Systems" in *Proceedings of the 39th Hawaii International Conference on System Sciences*, Kauai, Hawaii, USA, 2006

[8] A. Menezes, P. van Oorschot and S. Vanstone, *Handbook of applied cryptography*, CRC Press, 2001

[9] N. Botezatu, A. Stan and L. Panduru, "Power-aware Framework for Encrypted Communications", in *Proceedings of the 20th DAAAM World Symposium "Intelligent Manufacturing & Automation: Theory, Practice & Education"*, Vienna, Austria, 2009, pp. 825–826

[10] A. Stan and N. Botezatu, "Data Encryption Methods for Power-aware Embedded Systems used in Patient Monitoring", in *Proceedings of the 10th International Carpathian Control Conference*, Zakopane, Poland, 2009, pp. 269–272

Run-time Feasibility Verification in Event Driven Operating Systems with Static Priorities

Catalin Braescu and Lavinia Ferariu

Abstract— **This paper presents a run-time feasibility self-testing algorithm for the dynamic reconfiguration of embedded applications, compliant with event-driven real-time operating systems based on static scheduling. It exploits the tracer facilities to provide on-line evaluation of the execution times and laxities' prediction updating. Appropriate reconfiguration of the application is ensured in accordance to the current processor utilization by means of schedule feasibility verification performed at the activation/ termination of each process. Therefore, the embedded system can benefit of increased predictability and flexibility, as the proposed algorithm can detect overloading conditions difficult to be defined apriorically, whilst preserving high average resource utilization. The suggested algorithm is customized for OSEK/VDO real-time operating system and the performances of the suggested approach are illustrated on a multiple embedded controller with independent loops, designed for a DC motor with separate excitation.**

I. Introduction

Providing an attractive trade-off between performances and costs, embedded systems have gain increased interest in industrial applications. Software embedded applications are modularly built as a combination of periodic and aperiodic/sporadic processes which compete to receive available active resources. During on-line operation, the mono-processor tasks' execution is supervised by a scheduler, according to priorities assigned dynamically or statically. The scheduler decides which task takes the processor, whenever interrogated. Its calling may occur periodically or event driven, though last alternative is preferred, as it may provide increased flexibility and responsiveness, at convenient resource consumption.

Usually, the periodic tasks are devoted to solve critical operations according to a deterministic workload model [1]. No violation of their hard deadlines is accepted, yet in deterministic approaches the feasibility of the plan can be simply analyzed over a hyper-period, assuming the worst possible scenario. However, in numerous situations the use of aperiodic or sporadic tasks meant to treat external events becomes necessary, thus leading to difficulties in providing full application predictability. Moreover, even the execution of interrupt routines (ISR) is not validated by the scheduler, any accepted interrupt leads to additional processor overload and has to be considered for a valuable feasibility analysis. When dealing with sporadic tasks, the goal of the scheduler is frequently limited to merely ensure a deterministic run-time execution. One assumes the risk of rejecting several non-periodic processes, even assigned with hard deadlines, to gain run-time predictability, meaning that a sporadic task is accepted only if its deadline, as well as the deadlines of all other already accepted tasks (periodic or sporadic) could be satisfied [2].

Standing as a key issue in terms of application feasibility, numerous scheduling algorithms have been investigated. Most common static scheduling policies are rate monotonic (RM) [3] or deadline monotonic (DM), whilst earliest deadline first (EDF) or minimum laxity first (MLF) are frequently used for dynamic planning [4], [5]. The success of these approaches could be explained by specific proofs of optimality, available for mono-processor systems with periodic, preemptive, independent tasks. Though, when dealing with non-periodic processes and/or unexpected delays, the guarantee of building a feasible schedule is lost, even if such schedules exist.

It is worth mentioning that most real-time operating systems employ fixed priorities scheduling policies, targeting increased predictability and safety [6], [7]. They exploit the microcontrollers' built-in interrupt handling facilities to provide necessary responsiveness to external events, in order to keep a simpler tasks' architecture. Therefore the static configuration of the application has to be set in terms of the worst achievable conditions, which usually also involve maximum processor utilization. Even such overloading could occur very rarely, no compromise is permitted in terms of plan feasibility, so the payoff results in providing overall fewer facilities, reduced accuracy and/or low responsiveness, etc.

OSEK [6], [8] is an example of real time operating system based on fixed processes' priorities. It has been already accepted as a standard for multitasking embedded systems in automotive industry. The OSEK-based application can use various number of predefined OSEK objects, customised by means of specific attributes, yet all configurations must be carried out offline. As support for feasibility testing during application development stage, OSEK/VDX offers valuable debugging and testing tools,

This work was supported by The National Centre for Programs Management from Romania under the research grant SICONA – 12100/2008 and was partially financed by Continental Automotive Romania.

C. Braescu is with the "Gheorghe Asachi" Technical University of Iasi, Bd. Dimitrie Mangeron 27, Iasi, Romania (e-mail: cbraescu@ac.tuiasi.ro).

L. Ferariu is with the "Gheorghe Asachi" Technical University of Iasi, Bd. Dimitrie Mangeron 27, Iasi, Romania (e-mail: lferaru@ac.tuiasi.ro).

which permit to monitor the tasks' execution and the stack usage.

This paper exploits the run-time tracer facilities provided by the real-time operating system, to suggest a solution for dynamic reconfiguration of OSEK-based applications, based on run-time feasibility self-testing. The embedded system gains both in predictability and flexibility, making use of an on-line re-evaluation of the execution times and laxities' prediction. By enabling the switch between several operating modes, the application is dynamically tailored to the current processor utilization, making possible to handle potential overloading or under loading conditions, without altering the feasibility of the schedule [9], [10]. At one side, the approach detects unexpected overloading which are difficult to anticipate during the design stage and enables adequate corrections, meant to ensure data consistency and the validity of application responses. At the other side, tasks' design becomes more flexible, as the application is not necessarily configured merely in terms of the worst potential scenario, thus leading to enhanced overall facilities and increased average processor utilization.

The paper is organised as follows. Section II defines the problem to solve within the context of OSEK-based applications. A detailed description of the proposed algorithm is presented in Section III, whilst Section IV illustrates the applicability of the approach on a multiple embedded controller with independent control loops, designed for a DC motor with separate excitation. Conclusions are delivered in Section V.

II. Problem Statement

The proposed on-line feasibility self-testing takes into account the inherent characteristics of OSEK [6], [8]. Note that this real-time operating system is event – driven and employs static priorities. Whenever the scheduler is called, it gives the processor to the active task assigned with the highest priority, treating the queue of ready processes separately for each priority level, according to FIFO (first in, first out) algorithm. When working in preemptive mode, the scheduler is implicitly called if the running task is terminated or switches to an waiting state, if the queue of active tasks is changed, if a resource is released, or if an ISR2 process is finished and no other ISR has to be executed. In non-preemptive mode, the running task decides itself when to give control to the scheduler, by an explicit call or by freeing the processor.

Consequently, the feasibility of the plan cannot be guaranteed by OSEK, even if preemptive, independent tasks are considered. The designer is in charge with appropriate application configuration, according to maximum expected system overload (maximum number of simultaneous active tasks, maximum number of accepted interrupts, maximum execution times, maximum delays, etc.). Mainly, the control on tasks' execution is carried out by means of priorities apriorically assigned, based on any desired policy. If in certain operating conditions the application has to handle specific critical deadlines associated to aperiodic/sporadic processes of highest priorities (ISRs or tasks), the common solution consists in employing as larger periods as possible for all periodic tasks of lower priorities. Though, this static configuration may lead to assumed reduced resource utilization and/or low performances. Additionally, note that, in numerous situations, the worst case is difficult to define before hand.

Let us consider a set of concurrent processes, $\mathbf{P} = \{P_i, i = \overline{1, p}\}$, each one assigned with relative deadline D_i and maximum execution time e_i. For the sake of simplicity, one assumes that the processes are sorted in **P** according to their type, the boundary indexes being $1 \le m \le n \le p$. $P_{i,i=\overline{1,m}}$ are critical periodic tasks activated according to phases Φ_i and periods p_i, with $D_i \le p_i$. In addition, **P** may include sporadic and aperiodic tasks ($P_{i,i=\overline{m+1,n}}$), some of them allowed with higher priorities than a part of periodic tasks. Last elements of **P** may be ISR processes ($P_{i,i=\overline{n+1,p}}$), which implicitly gain higher priorities than any task of the embedded system. No multiple simultaneous activations are allowed. One considers that no violation of hard real-time constraints is accepted for periodic and sporadic tasks, as well as for interrupt routines, any positive tardiness in the processes' execution making application useless and corrupted.

All priorities are established before hand, yet the application can gain increased flexibility by means of a dynamic configuration of the jobs executed by each task at a certain activation and/or by on-line tuning of periods p_i, $i = \overline{1, m}$. In this attempt, the goal of the proposed algorithm is to ensure on-line detection of overloading and underloading conditions and to provide adequate on-line adjustment of application facilities, with on-line feasibility verification for the resulted plan. This strategy may lead to enhanced resource usage and heightened robustness in embedded applications compliant with real time operating systems with fixed processes' priorities.

In automatic control, different dynamic application configurations can be imagined [11], [12]. For instance, the numerical control algorithms can be implemented in terms of multiple sampling frequencies. Default small sampling periods can be set for enabling a prompt detection of disturbances [13], whilst employing delta formalism for a simple and accurate discretization [14], [15]. Whenever a system overload occurs (e. g. due to the necessity of plant model updating or caused by certain fault detections), the controller can switch to lower sampling periods, in order to

preserve the consistency of the control law [16], [17].

III. Algorithm Overview

The main idea of the algorithm consists in on-line determination of processes' execution times and updating of related laxities. On-line verification of real-time constraints is done at the end of each process, taking into account if some processes are apriorically marked as dependent.

For monitoring the execution of an application, OSEK [9] gives the opportunity to implement customized Hook routines and Tracer functions. PreTaskHook/PostTaskHook and PreISRHook/PostISRHook are implicitly called at the activation and the termination of a task or an ISR, respectively. Additionally, OSEK Tracer module calls the `Tracer_TaskChange` or the `Tracer_Interrupt` function, whenever a process state transition is detected. Similar facilities are provided by most real-time operating systems with static priorities, e.g. microC/OSII, Erika. The programmer can use these points as milestones for verifying the feasibility of the plan and for deciding if the current configuration of the application is appropriate. To benefit of enriched information, the recommendation is to exploit OSEK Tracer, as this permits the on-line updating of all e_i values, $i = \overline{1, p}$, with benefits on the robustness of laxities' prediction. Though, the feasibility of the plan has to be tested only at the activation/termination of each process, as, generally, the programmer has no control on the intermediary points where the context switches are decided by the scheduler. Note that even a task is built as a sequence of jobs, in preemptive mode any job can be preempted to give access to a process assigned with higher priority, so no relation could be established between the ends of component jobs and potential switching points.

As the number of processes is known apriorically, one can statically define a specific array of structures aimed at individual processes' monitoring. The structure associated to process P_i (denoted generically `info`) stores the process ID, the type of the process (preemptive/ nonpreemtive task or ISR, periodic or aperiodic process, etc.), the current state, the list of predecessors and successors, the current deadline D_i, the maximum execution time e_i, the total amount of execution time resulted during current activation, specific variables which permit to switch to alternative configuration of the application, etc. High resolution of time monitoring can be provided by means of on-chip timers. All timer transitions from 0 can be detected with a specific ISR process, keeping in `info` a separate track for each process, for a consistent evaluation of execution times and laxities. Moreover, if quite different time ranges are involved for distinct processes of the application, one can use multiple timers with different input clock frequencies and constants, as the address of corresponding data registers can be also stored in `info`.

Note that OSEK implicitly ensures the validity of a plan, if the precedence constraints are satisfied, by means of an adequate design. In a mono-processor system, a valid plan satisfies the following conditions: no job is scheduled before its release time, the total amount of processor time assigned to each job is equal to its execution time and all the precedence and resource usage constraints are satisfied.

If $P_i, i = \overline{1, p}$ are independent, then the test feasibility is separately performed for each process, without updating D_i in accordance to most recent execution times. In that case, one can usually consider $\Phi_i = 0$ and $D_i = p_i$. This kind of configuration can be achieved in numerous applications, by gathering the dependent processes into a single one.

If some processes are dependent, note that in mono-processor applications, the precedence constraints may be solved by analyzing the feasibility of the plan in terms of resulted effective release times and effective deadlines. Although, to provide increased robustness, the approach exploit the fact that all precedence constraints are statically configured. Therefore, the verification of plan feasibility is not performed only from the standpoint of the terminated process, but also for all its successors, by updating the corresponding D_i. This can be interpreted as an on-line adjustment of effective deadlines, in terms of most recent e_i. The number of successors and their IDs are stored in specific fields of `info`, to enable adequate on-line tests. Note that OSEK offers several alternatives for implementing sequences of processes with preceding constraints, such as event-based synchronizations and simple chaining of tasks (by means of `ChainTask` service).

An adequate balance has to be ensured between the granularity of employed tasks and application responsiveness to under loading/overloading conditions. This may involve splitting some tasks in several distinct processes with corresponding smaller e_i, as most real time operating systems provide timing and code execution control facilities available only at process level. On the other hand, attention has to be paid to the number of resulted precedence constraints, as a large number of successors leads to a large number of additional deadlines' verifications. Naturally, in mono-processor embedded systems, any task can be considered as a sequence of jobs described by the precedence graph illustrated in Fig. 1. This particular type of splitting can be easily implemented, by using `ChainTask` service to link the dependent processes.

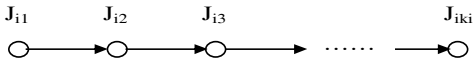


Fig. 1. Precedence graph for P_i 's jobs.

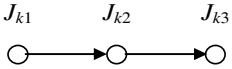


Fig. 2. Precedence graph corresponding to control law implementation

In addition, note that most real time operating systems limit the maximum number of allowed processes.

Several strategies could be used to build alternative operating scenarios. One can consider each process as a collection of compulsory and optional jobs. The application can be tailored by enabling/discarding several optional jobs in underloading/overloading conditions, by means of global volatile flags. Another possibility is to adjust the period of periodic tasks within predefined ranges, $p_{i_k}, k = \overline{1, q_i}$, $i = \overline{1, m}$ with corresponding configuration of related jobs. The decision could be taken individually, for each periodic process, or for groups of processes. In the last case, the simplest design can result by verifying the feasibility over the corresponding hyper-period, which can be computed as the minimum multiple of values $p_{i_k}, k = \overline{1, q_i}$ corresponding to all the processes belonging to the group. Note that in this case, at the termination point, each member of the group asks to verify the deadlines for all the other members, based on the list of related IDs, stored in `info`.

IV. Application

The performances of the proposed algorithm are illustrated on a multiple embedded controller with independent loops. The system uses the facilities offered by the OSEK real-time operating system and Freescale 16-bit MC9S12XDP512 microcontroller. The later has a large palette of on-chip peripherals, including two 10-bit analog-to-digital converters (ATD), an 8-channel pulse-width modulator (PWM), three serial peripheral interfaces (SPI), two 8-bit micro-timers and four 16-bit periodic interrupt timers (PITs).

Generally, each digital SISO (single-input single-output) controller can manage a certain process variable, according to specific sampling periods (Fig. 2). Any configuration of control loops is allowed, from independent ones to cascade – based architectures. Though, the hardware configuration imposes several limitations. As the microcontroller is not equipped with digital to analog (DA) converters, each analogue command could be sent by means of a PWM. Additionally, PWMs can be also used to implement the ATD triggers, leading to $i \le 6$ (Fig. 3). Based on available ATD facilities, only two controllers with independent sampling periods can be implemented, yet multiple control loops can be employed assuming dependent sampling periods, if ATDs are enforced to perform a sequence of readings, not merely individual acquisitions, whenever the ATD triggers are detected.

Each control law can be solved by means of a periodic process, P_k, comprising a job for ATD conversion (J_{k1}), a job for command computation (J_{k2}) and, additionally, if command DA conversion is necessary (Fig. 3), a job for command transmission (J_{k3}).

Details regarding the customization of the approach for the control of a DC motor with separate excitation are given in the following. The plant used for the experimental trials is part of an ELWE laboratory set-up, as shown in Fig.4. A cascade-based architecture is adopted, involving two independent (speed and current) control loops. The speed controller delivers the reference for the current control loop, therefore, it comprises two jobs only, denoted J_{11} and J_{12}, whilst the current controller needs all three component jobs J_{21}, J_{22} and J_{23} presented in Fig. 2. For providing increased flexibility and responsiveness, these jobs were implemented as separate dependent processes. More precisely, J_{11} and J_{21} correspond to Category 2 ISR processes (ATDO_ISR and ATDI_ISR, respectively), executed at the end of corresponding periodic AD conversions. ATDO_ISR activates the task TASKO_CMDC, which implement J_{12} to compute the command within the speed loop. ATDI_ISR is chained with TASKI_CMDC and TASKI_CMDT, which correspond to J_{22} and J_{23}, respectively. TASKI_CMDT sends the command to the DC motor within the inner current control loop.

The on-line validation of the schedule feasibility is done by calling the `fFesTest()` function each time a task or an ISR is ending. The application could be reshaped by modifying the sampling frequency according to presets values (e.g., the experimental trials considered 2kHz - initial value, 1kHz, 500Hz and 50Hz).

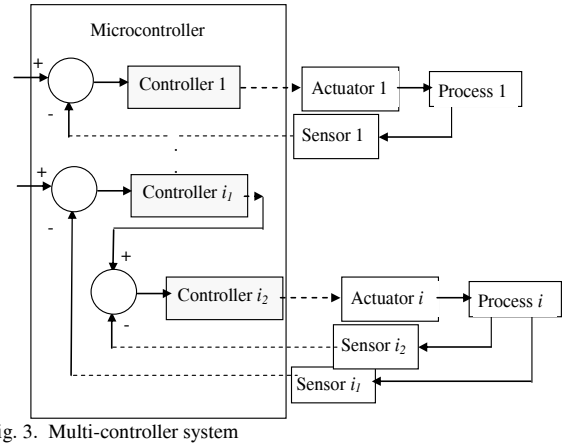


Fig. 3. Multi-controller system

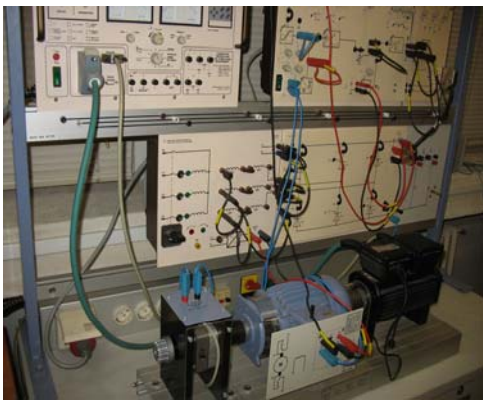


Fig. 4. Controlled DC motor with separate excitation

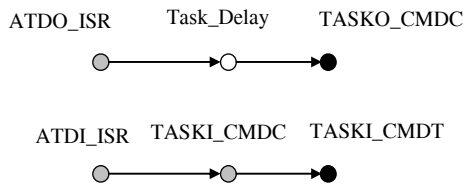


Fig. 5. Application precedence graph (including Task_Delay). The feasibility test is executed at the termination of all processes marked with grey and black and, if required, the sampling frequency changing is enabled at the end of the processes marked wih black.

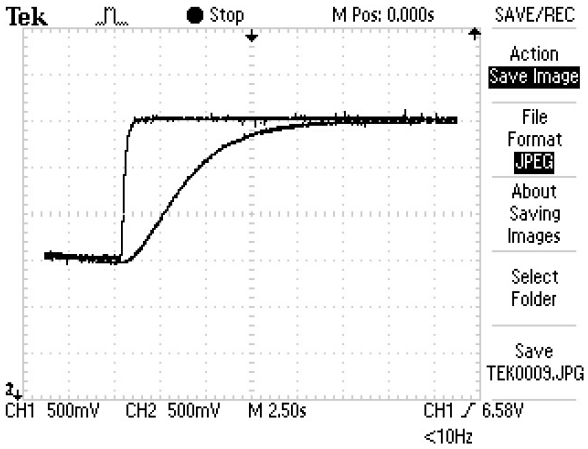


Fig. 6. The performances of the automatic system – the reference compared with the plant output in overload situation when no on-line feasibility testing is employed.

If overloading conditions are notified and a frequency decrease is necessary, a flag is set, indicating that any feasibility test for the processes implementing the current control law are suspended, in order to keep unnecessary overheads low. In the meantime, the sampling period is decreased as soon as possible, so that the system could benefit of higher responsiveness. As the feasibility test function is separately implemented for each controller, a dynamic and efficient workload management is in place.

Both TASKO_CMDC and TASKI_CMDC consider PI control laws tuned according to available approximate linear model describing the plant behavior in terms of input-output formalism. The transfer functions for the power converter control unit, the current feedback and the tacho-generator feedback were presented in a previous paper [18].

The undesired increasing of the processor's utilization was simulated by executing an additional task (Task_Delay), before each activation of TASKO_CMDC. Task_Delay is assigned with the highest priority and introduces a variable delay, manageable by means of a control register user-set value. The precedence graph indicated in Fig. 5 summaries the constraints imposed within the application.

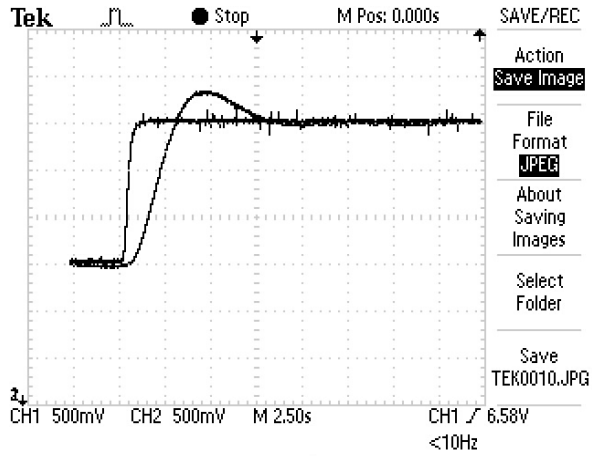


Fig. 7. The performances of the automatic system – the reference compared with the plant output, when feasibility self –testing is used.

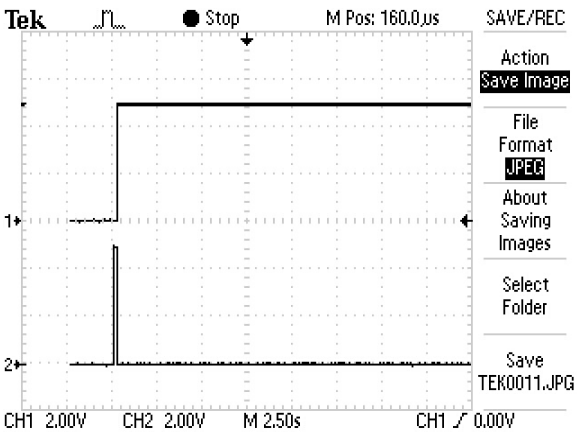


Fig. 8. Sampling frequency commutations decided by the on-line feasibility self-testing algorithm in overloading/underloading conditions. Top plot is for PK0 and the bottom one for PK1.

Without the on-line feasibility test, the performances of the cascade control system result unsatisfactory during overloading conditions (Fig.6) (higher rising time and

longer transition to the stationary state).

Note that the activation of Task_Delay was triggered by the increasing of the speed controller's reference from 800rpm (1.5V) to 1600rpm (3V). As the delays occur before command calculation, some activations of TASKO_CMDC are lost, leading to improper controller behavior. Obviously, the solution does not consist in setting multiple activations of TASKO_CMDC (as its late executions are out of interest), yet in configuring the employed control law for a higher sampling period known as valid in terms of plant dynamics.

If the on-line feasibility self-testing is used, the automatic system gains in accuracy and responsiveness, as displayed in Fig. 7. Note that this behavior is very close to the one achieved in normal conditions, when no overloading is simulated, assuming the same models of the controllers. The sampling frequency commutations are indicated by the digital outputs PK0 and PK1 (Fig. 8, top and bottom plot, respectively). PK0 is associated with the speed controller (value 0 indicates the default highest frequency operating mode and value 1 indicates the use of a lower sampling frequency), whilst PK1 is similarly related to the current control loop. The feasibility self testing was able to expose unreachable deadlines for enabling the reduction of the sampling frequency. The frequency switching is independently managed for each controller, when overloading is detected (Fig.8). First decision consists in an initial frequency adjustment in both speed and control loop, as basis for preserving the consistency of the control algorithms, with benefits for the overall behavior of the automatic system. Although, in this case, the sampling frequency reduction within the speed control lop is sufficient for meeting all deadlines. So, when the processor's utilization permits, the feasibility self-test procedure enables the reuse of higher sampling period for the current controller (Fig. 8).

V. CONCLUSION

The paper presents an algorithm for dynamic reconfiguration of OSEK-based applications, based on run-time feasibility testing.

The suggested approach benefits of the OSEK's predictability and safety. Additionally, heightened flexibility is provided relative to run-time application configuration, in accordance to processor's utilization. By using the tracer-type facilities offered by the real-time operating system, the designer is able to handle overloading conditions which are not apriorically defined, whilst preserving the schedule's feasibility. Moreover, increased overall performances are achieved within the framework of static scheduling policies, as the application is not necessarily tailored solely in terms of the worst scenario.

The algorithm performances were experimentally revealed for an embedded cascade controller customized for a DC motor with separate excitation and Freescale 16-bit MC9S12XDP512 microcontroller.

REFERENCES

[1] Li Q., Yao C., *Real-Time Concepts for Embedded Systems*, CMP Books, USA, ISBN 1-57820-124-1, 2003.

[2] R. Zurawski, *Embedded Systems Handbook*, Ed. CRC Press, 2006.

[3] G. Buttazzo, "Research Trends in Real-Time Computing for Embedded Systems", *ACM SIGBED Review*, vol. 3, no. 3, July 2006.

[4] G. Buttazzo and P. Gai, "Efficient EDF Implementation for Small Embedded Systems", *Proc. of the 2nd Workshop on Operating Systems Platforms for Embedded Real-Time applications (OSPERT 2006)*, Dresden, Germany, July 2006.

[5] C. Diederichs, U. Margull, F. Slomka and G. Wirrer, "An Application-Based EDF Scheduler for OSEK/VDX", *Proc. of the Conference on Design, Automation and Test in Europe*, Munich, Germany, 2008.

[6] J. Lemieux, *Programming in the OSEK/VDX Environment*, Ed. Elsevier, 2001.

[7] Q. Li, "Fundamentals of RTOS-Based Digital Controller Implementation", In Hristu-Varsakelis, Levine (ed.), *Handbook of Networked and Embedded Control Systems*, pp. 173-196, Control Engineering, Birkhäuser, 2005.

[8] ***, *OSEK/VDK Operating System Specifications*, http://www.osek-vdx.org.

[9] G. Buttazzo, M. Velasco and P. Marti, "Quality-of-Control Management in Overloaded Rea-Time Systems", *IEEE Transactions on Computers*, Vol. 56, No. 2, pp. 253-266, February 2007.

[10] G.C. Buttazzo, G. Lipari, M. Caccamo and L. Abeni, "Elastic Scheduling for Flexible Workload Management", *IEEE Transactions on Computers*, Vol. 51, No. 3, pp. 289-302, March 2002.

[11] R. Marau, P. Leite, L. Almeida, M. Velasco, P. Marti and J.M. Fuertes, "Implementing Flexible Embedded Control on a Simple Real-Time Multitasking Kernel", Research report ESAII-RR-07-10, Automatic Control Department, Technical University of Catalonia, Barcelona, Spain, April, 2007.

[12] R. Marau, P. Leite, M. Velasco, P. Marti, L. Almeida, P. Pedreiras and J.M. Fuertes, "Performing Flexible Control on Low-Cost Microcontrollers Using a Minimal Real-Time Kernel", *IEEE Transactions on Industrial Informatics*, vol. 4, no. 2, May 2008.

[13] K.J. Åström and R.M. Murray, *Feedback Systems: An Introduction for Scientists and Engineers*, Princeton University Press, Princeton, 2008.

[14] M.A. Chadwick, V. Kadirkamanathan and S.A. Billings, *Analysis of fast-sampled non-linear systems: Generalised frequency response functions for δ operator models*. Signal Processing 86 (11), 3246–3257, 2006.

[15] K. Shim and M.E. Sawan, "Singularly perturbed unified time systems with low sensitivity to model reduction using delta operators", *International Journal of Systems Science*, 37 (4), 243–251, 2006.

[16] P. Gai and G. Buttazzo, "An Open Source Real-Time Kernel for Control Applications", *Proc. of the 47th Italian Conference of Factory Automation (ANIPLA 2003)*, Brescia, Italy, November 21-22, 2003.

[17] M. Marinoni, T. Facchinetti, G. Buttazzo and G. Franchino, "An Embedded Real-Time System for Autonomous Flight Control", *Proc. of the 50th Int. Congress of ANIPLA on Methodologies for Emerging Technologies in Automation (ANIPLA 2006)*, Rome, Nov. 2006.

[18] F.C. Braescu, L. Ferariu and C. Lazar, "OSEK-Based Multiple Controllers with Schedule Feasibility Self-testing", *Proc. of the International Symposium on Power Electronics, Electrical Drives, Automation and Motion*, Pisa, Italy, June 2010.

ESTIMATION AND CONTROL IN THE PRODUCTION OF PHB

Elena Bunciu

***Abstract*—The purpose of this paper is to present a procedure for the estimation and control of the poly-β-hydroxybutyrate (PHB) production by mixed culture. In the mixed system sugars such as glucose obtained from food processing waste were converted into lactate by Lactobacillius delbrueckii and lactate was converted in turn to poly-β-hydroxybutyrate (PHB) by Ralstonia eutropha strain E2 (previously Alcaligenes sp.) into a fermented one as a model system.**

I. INTRODUCTION

In the last few years there has increased the interest for the biological methods for the production of biodegradable polymers. In this category a major class is represented by the family of polyhydroxyalkanoates (PHAs) [1].

A very important member in the class of PHAs is Poly- β -hydroxybutyrate (PHB). PHB has some properties that make it potentially competitive with synthetic polymers, being a biodegradable material [2, 3]; it is accumulated as a reserve energy material by various microorganisms, including species of bacteria such as the genera Alcaligenes, Bacillus, Pseudomonas, and Rhizobium [4–6].

PHB can be applied in the medical, agricultural, and industrial fields [7]. Despite this fact, the commercial production of PHB has had a slow development.

There have been several studies that indicated that Ralstonia eutropha is the preferred organism because it is easy to grow and it is able to accumulate a large quantity of PHB in a simple medium [8].

Previous studies have revealed that Ralstonia eutropha can be classified into several kinetically different groups according to K_s (the apparent half-saturation constant in Haldane's equation) and K_{sI} (the apparent inhibition constant) values [9].

Ralstonia eutropha strain E2 (previously Alcaligenes sp.), did not belong to any group. It is a phenol-degrading bacterium expressing a phenol-oxygenating activity with low K_s and an extremely high K_{sI} [9].

A lot of studies have revealed that fed batch cultivation is considered to be the preferred method of PHB production because this is an energy storage polymer that the cells synthesize under unfavorable growth conditions, typically a shortage of nitrogen or sulphur and the fed batch operations allow the concentrations of these nutrients and the carbon source to be dynamically controlled in an optimal manner [8].

Elena Bunciu is with the Automation Department within the Faculty of Automation, Computers and Electronic, Craiova, Romania; e-mail: bunciu.elena@yahoo.com).

Mixed systems are used in several fermentation processes. In many fermentation systems, microorganisms assimilate one substrate and transform it into an intermediate metabolite, also converted from other microorganisms into a metabolite – target product. In the case of PHB production, the sugars obtained from food processing waste were converted to lactate by Lactobacillius delbrueckii and lactate was converted to PHB by Ralstonia eutropha within the reactor. The control design of such processes it is a complicated task [10, 11]. The cell concentration, in all experiments [12], was measured by optical density at 6600 g/l with a spectrophotometer. The lactate concentration was measured by an enzyme kit (F-kit l-lactic acid 139084, Boehringer Mannheim, Germany). The ammonium sulphate concentration was measured in the same way as the ammonium ionic concentration by the enzymatic method [12].

Using the improved method of Law and Slepecky [13] there was determined the amount of PHB. The glucose concentration was determined by a glucose sensor (BF-400, Biot, Tokyo, Japan) with FIA (flow injection analyzer system) [11].

Starting from the determination of the control law, we continued by considering all the states in the system known to see how this is acting in open loop, and then in close loop.

Then, we considered that the process model was not completely known, and estimated the unknown parameters.

The paper is organized as follows: section II is dedicated to mathematical modeling of the process, in Section III we have presented some nonlinear and adaptive control strategies; simulations results presented in Section IV illustrate the performance of the proposed control algorithms and, finally, Section V concludes the paper.

II. MATHEMATICAL MODELS

In the past, several mathematical models have been simulated for the lactic acid fermentation. In recent studies, there has been presented one model that can be described by the dynamics of Lactobacillius delbrueckii growth based on mass balance [12].

The mathematical model that will be used in this study is the following [11]:

$$\frac{dX_1}{dt} = \mu_1(S,P,D)X_1 - \frac{F}{V}X_1 \qquad (1)$$

$$\frac{dS}{dt} = -\upsilon_1(S,P,D)X_1 + \frac{F}{V}(S_{in} - S) \qquad (2)$$

$$\frac{dP}{dt} = \sigma_1(S,P,D)X_1 - \upsilon_2(S,P,D)X_2 - \frac{F}{V}P \quad (3)$$

In these equations the following symbols appear:

μ_1 – *specific growth rate of L.Delbrueckii*

υ_1 – *specific glucose consumption rate*

σ_1 – *specific lactate production rate*

μ_2 – *specific growth rate of R.Eutropha*

υ_2 – *specific substrat consumption rate*

σ_2 – *specific PHB production rate*

μ_{m1}, μ_{m2} – *maximum value of* μ_1, μ_2

D – *dilution rate*

X_1 – *biomass concentration of L.Delbrueckii*

X_2 – *biomass concentration of R.Eutropha*

S – *substrate concentration of L.Delbrueckii*

P – *synthesis product concentration*

For the modeling on Ralstonia eutropha growth, the model below will complete the previous model with the following equations [11]:

$$\frac{dX_2}{dt} = \mu_2(P,D,N)X_2 - \frac{F}{V}X_2 \quad (4)$$

$$\frac{dN}{dt} = -\upsilon_3(S,P,D)X_2 - \frac{F}{V}N \quad (5)$$

$$\frac{dQ}{dt} = \sigma_2(N)X_2 - \frac{F}{V}Q \quad (6)$$

In these equations, the expressions of $\mu_1, \upsilon_1, \sigma_1, \mu_2, \upsilon_2$, σ_2 and υ_3 were assumed to be of the following form [11]:

$$\mu_1(S,P,D) = \frac{\mu_{m1}S}{K_S + S}\left(1 - \frac{P}{Pm}\right) \quad (7)$$

$$\upsilon_1(S,P,D) = \frac{\sigma_1(S,P,D)}{Y_{P/S}} \quad (8)$$

$$\sigma_1(S,P,D) = \alpha\mu_1(S,P,D) + \beta(S,D) \quad (9)$$

$$\beta(S,D) = \frac{\beta_m DS}{K_S + S} \quad (10)$$

$$\upsilon_2(S,P,D) = \frac{\mu_2(P,D,N)}{Y_{X_2/P}} \quad (11)$$

$$\mu_2(P,D,N) = \frac{\mu_{m2}P}{K_P + P + P^2/K_i}\left(\frac{N}{K_N + N}\right) \quad (12)$$

$$\upsilon_3(S,P,D) = \frac{\mu_2(P,D,N)}{Y_{X_2/N}} \quad (13)$$

$$\sigma_2(N) = q_m\left(\frac{K_N}{K_N + N}\right) \quad (14)$$

Where, we have:

Q – *PHB concentration*

N – *substrate concentration of R.Eutropha*

$Y_{P/S}$, $Y_{X_2/P}$, $Y_{X_2/N}$ are yield coefficients.

The numerical values for the required parameters are [12]:

$\alpha = 1.23$ $\quad \beta_m = 1.80$ $\quad \mu_{m1} = 0.375$

$\mu_{m2} = 0.734$ $\quad K_i = 2.5$ $\quad K_N = 0.146$

$K_p = 6$ $\quad K_s = 35.8$ $\quad P_m = 42.9$

$Y_{X_2/P} = 0.204$ $\quad Y_{P/S} = 0.698$ $\quad Y_{X_2/N} = 2.41$

$D = 3$ *ppm*

III. Control Strategies

A. Control Objective

The control objective consists in the adjusting of the system in order to produce PHB. To be exact, considering that the process model (1)-(6) is incompletely known, its parameters are time varying and not all the states are available for the measurements, the control goal is to maintain the process at some operating points, which correspond to a maximal lactic production rate [14].

We have chosen that the operating point to be kept around the point $S^* = 5$ g/l and then around $S^* = 15$ g/l.

B. Exactly Linearizing Feedback Controller

First, we want to evaluate the ideal case where all the knowledge concerning the process (kinetics, yield coefficients and state variables) are available [14].

Let us consider a closed loop system with a dynamic in accordance with a stable first order linear system, which is described by the equation below [15]:

$$\frac{d}{dt}\left(S^* - S\right) + \lambda_1\left(S^* - S\right) = 0 \quad (15)$$

Assuming $\lambda_1 = 0.5$, $\frac{dS^*}{dt} = 0$, we will have:

$$\frac{dS}{dt} = \lambda_1\left(S^* - S\right) \quad (16)$$

Replacing $\frac{dS}{dt}$ in (2) with (16), we will obtain:

$$\lambda_1\left(S^* - S\right) = -\upsilon_1(S,P,D)X_1 + \frac{F}{V}(S_{in} - S) \quad (17)$$

The control law, which is deduced from the above equation, is given by:

$$F = \frac{V}{S - S_{in}}\left(\lambda_1\left(S^* - S\right) + \eta_1 X_1\right) \qquad (18)$$

IV. Simulation Results

In the simulations below the following initial conditions are used:

$$X_1 = 0.5\,(g/l); \quad S = 9\,(g/l); \quad P = 0\,(g/l);$$
$$X_2 = 0.2\,(g/l); \quad N = 0.3\,(g/l); \quad Q = 0\,(g/l);$$

The first step is to simulate our open loop system. The obtained graphic, by considering $F = 2$, is presented in Fig. 1.

For the closed loop simulation using the command law (18), we will obtain the graphics presented in Fig. 2 and Fig. 3.

In Fig. 2 there is presented the form taken by S when S^* is smaller than initial value of S, namely $S^* = 5(g/l)$.

In Fig. 3 S^* is higher than the initial value of S, $S^* = 15(g/l)$.

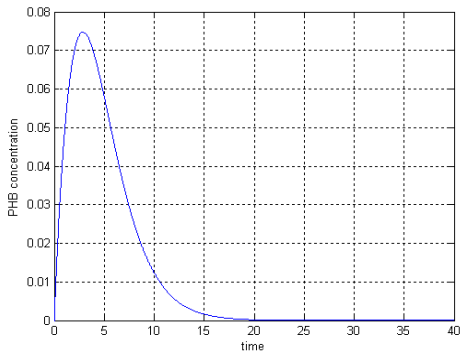


Fig.1. PHB concentration in an open loop

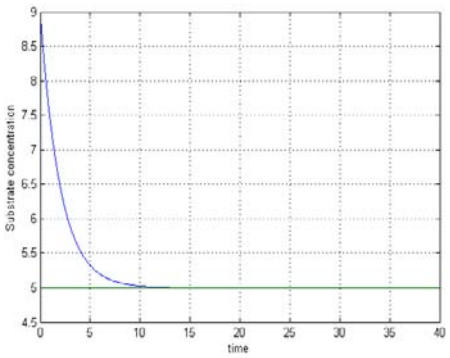


Fig.2. Substrate concentration when the reference is 5 (g/l)

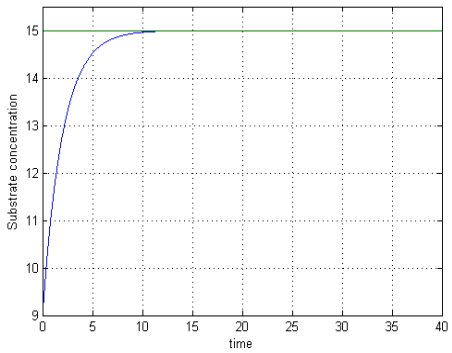


Fig.3. Substrate concentration when the reference is 15 (g/l)

For the next step, we wish to estimate the values of X_1 and X_2.

For the practical implementation of the control law, we must know the variables X_1, S, X_2, N and all the production and consumption rates involved in this process [15].

Due to the fact that variables X_1 and X_2 are not measurable on-line, the control law will become an adaptive law, by replacing the real and unknown values of X_1 and X_2 with their estimations provided on-line by a state observer [15].

For the estimation of X_1 variable, we will use an asymptotic observer with a structure presented below.

We define an auxiliary value z_1, assumed to be of the following form:

$$z_1 = \frac{\alpha}{Y_{P/S}} X_1 + S \qquad (19)$$

The dynamics of z_1 variable is explained by the following stable linear equation:

$$\frac{d\hat{z}_1}{dt} = -\frac{F}{V}\hat{z}_1 + \frac{F}{V}S_{in} \qquad (20)$$

From (19) and (20) we can determine the estimated value $\hat{X}_1$ of X_1, which will be:

$$\hat{X}_1 = \left(\hat{z}_1 - S\right)\frac{Y_{P/S}}{\alpha} \qquad (21)$$

For the estimation of X_2 we will use the same method we have used in the estimation of X_1, but we define another variable z_2, assumed to be the following:

$$z_2 = \frac{1}{Y_{X_2/n}} X_2 + N \qquad (22)$$

The dynamics of z_1 variable is explained by the following stable linear equation:

$$\frac{d\hat{z}_2}{dt} = -\frac{F}{V}\hat{z}_2 \qquad (23)$$

Using the above two equations we can write that the estimated value of X_2 is:

$$\hat{X}_2 = Y_{X_2/N}\left(\hat{z}_2 - N\right) \qquad (24)$$

In order to obtain the Fig. 2 and Fig. 3, we will use the following initial values:

$$\hat{z}_1 = 10$$
$$\hat{z}_2 = 0.4 \qquad (25)$$

These were the values which better estimated the real parameters.

In the Fig. 4 and Fig. 5 we present comparatively the real value and estimated value of the two biomass concentrations.

The form of the input substrate concentration (S_{in}) is presented in Fig. 6.

Moving on, for estimating the specific growth rate of Ralstonia eutropha and the specific growth rate of Lactobacillius delbrueckii, we will use an estimator based for modifying the model described by the equations (1)-(6) in a linear regressive form presented in (27)-(30). [15]

In the linear regressive model the following symbols appear:

$$\xi = \begin{pmatrix} X_1 & S & P & X_2 & N & Q \end{pmatrix}^T \qquad (26)$$

K is the matrix of the production coefficients

I_6 is the unity matrix

$\Omega = diag\{-\omega_i\},\ i = 1:6$.

$\Psi^T(t)$, $\Psi_0(t)$, are the output of the linear filters.

$$\frac{d\Psi^T(t)}{dt} = \Omega\Psi^T(t) + KH(\xi) \qquad (27)$$

$$\frac{d\Psi_0(t)}{dt} = \Omega\Psi_0(t) - \left(\Omega + \frac{F}{V}I_6\right)\xi + u \qquad (28)$$

$$\frac{d\hat{\alpha}(t)}{dt} = \Gamma\Psi\left(\xi - \Psi_0 - \Psi^T\hat{\alpha}\right) \qquad (29)$$

$$\frac{d\Gamma}{dt} = -\Gamma\Psi\Psi^T\Gamma \qquad (30)$$

The specific growth rate of Ralstonia eutropha and the specific growth rate of Lactobacillius delbrueckii considered unknown can be written in the form:

$$\mu_1(S,P,D) = P\alpha_1 \qquad (31)$$

$$\mu_2(P,D,N) = P\alpha_2 \qquad (32)$$

We can specify that:

$$\alpha_1 = \frac{\mu_{m1}S}{K_S + S}\left(\frac{1}{P} - \frac{1}{Pm}\right) \qquad (33)$$

$$\alpha_2 = \frac{\mu_{m2}}{K_P + P + P^2/K_i}\left(\frac{N}{K_N + N}\right) \qquad (34)$$

For the presented model we will consider:

$$\Psi^T = \begin{bmatrix} \alpha & -\frac{1}{Y_{X_2/P}} \\ 0 & -\frac{1}{Y_{X_2/N}} \end{bmatrix} \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix} \qquad (35)$$

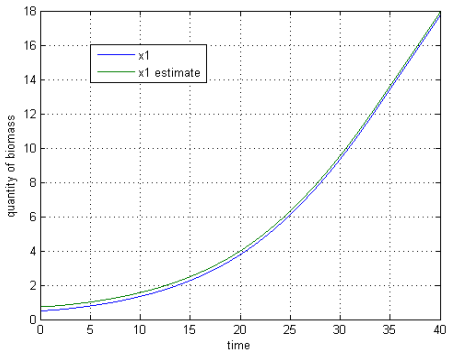


Fig.4. Estimation of the biomass concentration of Lactobacillius delbrueckii

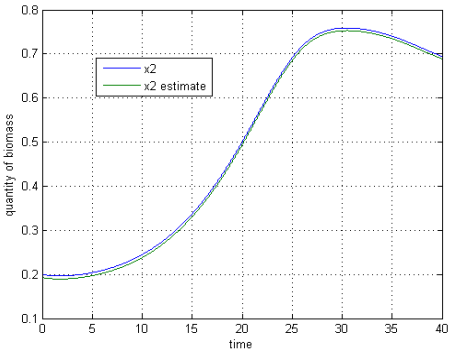


Fig.5. Estimation of the biomass concentration of Ralstonia eutropha

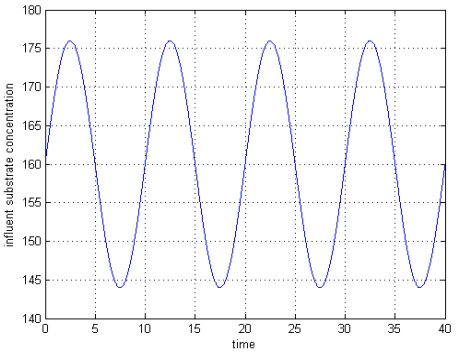


Fig.6. Influent substrate concentration

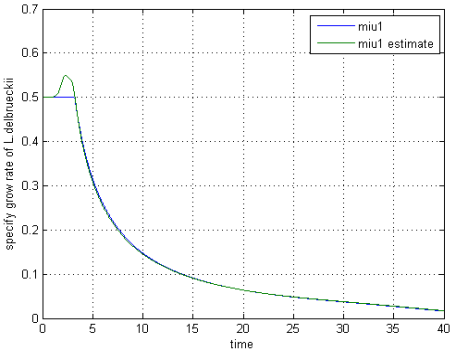


Fig.7. Estimation of specific grow rate of Lactobacillius delbrueckii

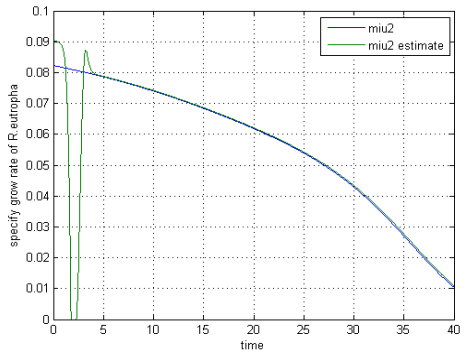


Fig.8. Estimation of specific grow rate of Ralstonia eutropha

V. CONCLUSION

In this paper we have estimated the biomass concentration and the specific growth rate from the system described by the dynamics of Lactobacillius delbrueckii combined with the model of Ralstonia eutropha growth.

For the determined control law, the sugar feed rate is obtained by using a stable first order linear system that is equalized with the substrate equation from the model (1) - (6).

By modifying the control law obtained in the necessary way, we estimated all the biomass concentration and specific grow rate included in the system.

Depending on the value of reference chosen for S, the initial values of $\hat{z}_1$ changed in order to obtain a better estimation.

We had to modify the $\hat{z}_1$ initial value because the control law depended only on the biomass concentration of Ralstonia eutropha.

For the biomass concentration estimation, we modify the control law (18) using $\hat{x}_1$ instead of x_1.

For the growth rate estimation, we modify the control law (18) using $P\alpha_1$ instead of μ_1.

These changes were made because we considered that x_1, x_2 were unknown and the control law had to depend on their estimation.

In the Fig. 4 and Fig. 5 we can see that the both biomasses are very well estimated.

In Fig. 7 and 8 we observed that, estimations between 0.5 and 3.2 both growth rates are not quite accurate, but after that, the estimation takes the same shape as the real value itself.

ACKNOWLEDGMENT

This work was partially supported by the strategic grant POSDRU/88/1.5/S/50783, Project ID50783 (2009), co-financed by the European Social Fund – Investing in People, within the Sectorial Operational Programme Human Resources Development 2007 – 2013.

REFERENCES

[1] Ojumu TV, Yu I, Solomon BO (2004) Production of Polyhydroxyalkanoates, a bacterial biodegradable polymer, African Journal of Biotechnology, 3, p 18-24

[2] Kato N, Konishi H, Shimao M, Sakazawa C (1992) Production of 3-hydroxybutyric acid trimer by Bacillus megaterium B-124. J Ferment Bioeng 73: p 246–247

[3] Patnaik PR (2005) Perspectives in the modeling and optimization of PHB production by pure and mixed cultures. Crit Rev Biotechnol 25: p 153–171

[4] Lee SY (1996) Bacterial polyhydroxyalkanoates. Biotechnol Bioeng 49: p 1–14

[5] Mercan N, Aslim B, Yüksekdağ ZN, Beyatli Y (2002) Production of poly-b-hydroxybutyrate (PHB) by some Rhizobium bacteria. Turk J Biol 26: p 215–219

[6] Arun A, Murrugappan RM, Ravindran ADD, Balaji S (2006) Utilization of various industrial wastes for the production of poly-b-hydroxybutyrate (PHB) by Alcaligenes eutrophus. Afr J Biotechnol 5: p 1524–1527.

[7] Braunegg G, Lefebvre G, Genser KL (1998) Polyhydroxyalkanoates, biopolyesters from renewable resources: physiological and engineering aspects. J Biotechnol 65:p 127–161

[8] Patnaik P (2006), Dispersion optimization to enhance PHB production in fed-batch cultures of Ralstonia eutropha, Bioresource Technology, 97, p 1994-2001

[9] Hino S, Watanabe K, Takahashi N (1998) - Phenol hydroxylase cloned from Ralstonia eutropha strain E2 exhibits novel kinetic properties, Microbiology, 144, p 1765-1772

[10] Ishizaki, Ueda T. (1995), Growth kinetics and product inhibition of Lactococcus Lactis 10-1 culture in xylose medium, Journal Ferment. Bioeng., 80, p. 287-290

[11] Tohyama M., Patarinska T., Qiang Z., Shimuzi K. (2002), Modeling of the mixed culture and periodic control for PHB production, Journal. Biochemical Engineering, 10, p..157-173

[12] Popova S.(2006), On-line state and parameters estimation based on measurements of the glucose in mixed culture system, Journal Biotechnology & Biotechnological Equipment, 2006, 3/20-p.208-214

[13] Law J. H., Slepecky R.A. (1961), Assay of poly-b-hydroxybutric acid, Journal Bacterial., p. 33-36

[14] Petre E., Selişteanu D., Ionete C. (2001), Nonlinear adaptive control strategies for lactic fermentation process, The 11-th Int. Symposium on Modeling, Simulation and System's Identiication – SIMSIS-11, Galaţi, Romania, p. 124-129.

[15] Bastin G., Dochain D.(1990) On-line estimation and adaptive control of bioreactors, Elsevier, Amsterdam

Graph Genetic Programming Toolbox for Neural Identification

B. Burlacu and L. Ferariu

Abstract— **This paper presents a new graph genetic programming based approach devoted to the design of feed forward hybrid neural models. The suggested methodology permits the construction of partially interconnected heterogeneous architectures with local and global neurons, as a basis for providing enhanced approximation capabilities for a large class of nonlinear system identification problems. The use of graph genetic programming allows the direct encoding of the neural networks and, consequently, a simpler interpretation of the hierarchical individuals. To avoid the production of incorrect models, the authors recommend a special minimally sufficient set of functions and customized compatible genetic operators, which exploit the inherent characteristics of the neural architectures (modularity, high interconnectivity and high level of parallelism). The software package features high scalability and good stress performances, by means of efficient data structures management and multithreaded execution on multi core processors. As it supports low memory consumption and reduced execution times, the C application is suitable for complex nonlinear system identification problems.**

I. Introduction

Collaborative neuro - genetic systems represent a valuable approach for complex nonlinear identification problems. They provide increased flexibility in selecting appropriate model structure and parameters for a wide range of applications, without demanding rich aprioric information [1], [2], [3]. On one hand, the neuro-genetic systems exploit the learning ability of the neural networks, as well as the computational efficiency provided by highly interconnected and parallel neural architectures. On the other hand, the evolutionary algorithms allow a more robust and flexible design of the model structure and/or the selection of corresponding parameters, whilst dealing with difficulties specific to industrial environment, such as noisy data, time – variance and nonlinearities.

Based on increased adaptation capabilities provided by the neuro-genetic symbiosis, numerous approaches have been suggested in the related literature [1]. As the neural topologies inherently involve a hierarchical representation, an important issue refers to the adopted evolutionary encoding. Obviously, the most compact direct representation is the graph–based one, yet this is not a common encryption in the standard evolutionary loop. The genetic algorithms are limited to the encoding of the hierarchical topologies by means of supplementary "control" genes, which leads to longer chromosomes and, consequently, to a more difficult exploration. Furthermore, the genetic programming (GP) implicitly works on tree-based individuals and the extension to graphs requires the design of specific genetic operators.

This paper employs the graph GP [3], [4] to ensure a concomitant selection of neural structures and parameters for the **h**ybrid **n**eural **n**etworks (HNN). By exploiting the modularity and the parallelism of the neural network, the authors propose a special definition of primitives' sets and corresponding specific crossover and mutation operators. These techniques guarantee that any recursive combination of functional and terminal nodes leads to valid individuals, from both a phenotypic and genotypic standpoint. Moreover, despite previous graph evolutionary algorithms which deal with a limited variety of structural templates (especially with MLP), the modular generation of graph-encrypted individuals enables the development of partially interconnected hybrid networks, which are expected to deliver increased performances for wide classes of applications [1], [3]. The hybrid neural networks accept combinations of hidden neurons with global and local responses, thus being suitable for both interpolation and extrapolation problems [2].

The paper is organized as follows. Section II browses the main features of the suggested design algorithm, whilst Section III discusses details regarding the proposed C implementation, with emphasis on data structure design and algorithms adopted for population initialization and offspring generation. Section IV comments the applicability of the approach to chaotic time series prediction via several experimental trials and last section is devoted to conclusions.

II. Algorithm Overview

The algorithm starts with a randomly generated initial population of graph-based individuals, each one encrypting a possible hybrid neural model. The quality of each solution is assessed in terms of output squared error computed over

Manuscript received April 30, 2010. This work was supported by The National Centre for Programs Management from Romania under the research grant SICONA – 12100/2008.

B. Burlacu and L. Ferariu are with the Department of Automatic Control and Industrial Informatics, "Gheorghe Asachi" Technical University of Iasi, Romania, Bd. D. Mangeron 27, IS 700 050 RO (e-mail: bburlacu@ ac.tuiasi.ro, lferaru@ac.tuiasi.ro).

the whole training data set and, subsequently, the fitness values, stated as selection probabilities, are determined by means of linear ranking. At each generation, the best individuals are selected into the recombination pool, by using stochastic universal sampling. Afterwards, specific crossovers and mutations are applied to produce new models, with potentially better performances. Note that the genetic operators act at both structural and parametric level. The best offspring replace the less adapted individuals in the population which is passed to the next generation.

The suggested design algorithm is compliant with feed-forward, partially interconnected HNN with external delay blocks. A neuron may receive input stimuli from any subset of neurons of the previous layer, as well as from any subset of the neural inputs. To provide enhanced adaption capabilities in interpolation and extrapolation problems, the hidden layers can comprise of any combination of neural units with local or global response. Note that the neurons with global response are able to bring increased generalization capabilities and robustness, as they aggregate their numerous inputs by means of a dot product operator, yet, the local neurons provide high computational efficiency due to the use of Euclidian distance input operator, which permits the activation of each neuron within a small area around a certain input sample.

The algorithm makes use of graph GP, namely it employs a direct encoding of neural architectures, by means of directed acyclic graphs (DAG). As opposed to the tree-based encoding involved in classic GP [3], [4], the DAG allows the encryption of each neural connection as a supplementary graph link, which leads to an efficient reuse of the available structural blocks and a significant reduction of memory consumption in the context of highly interconnected architectures [5]. The suggested encoding accepts only variable – type terminals, placed on the leaves of the graph. When series – parallel, input - output identification schemes are adopted, the terminals' set, $\mathbf{T}$, may contain lagged plant input and output measurements $\mathbf{T} = [\mathbf{u}(k),..,\mathbf{u}(k-n_u),\mathbf{y}(k-1),..,\mathbf{y}(k-n_y)]$, where $\mathbf{u} \in \Re^m$ and $\mathbf{y} \in \Re^n$ denote the inputs and the outputs of the system, k indicates the current sampling instant and n_u, n_y represent the maximum permitted input and output lags, respectively.

To exploit the modularity of the neural topologies inside the structure of the hierarchical DAG-based individuals, each neuron is encrypted by a single node of the graph. The set of functions, $\mathbf{O}$, contains the functions describing the input-output mapping performed by the accepted neural unit. Therefore, a simple extension to any desired hybrid neural architecture is permitted, yet, more important, one guarantees that any recursive combination between the elements of $\mathbf{O}$ and $\mathbf{T}$ encode a valid neural network. The functions of $\mathbf{O}$ accept variable arity and embed the corresponding neural parameters. The proposed encryption provides simple interpretations of the neural models, as well as compact representations. However, the genetic operators have to be reconfigured to work also on the inner structure of the functional nodes.

This paper considers the combination of global perceptrons with or without functional links, respectively, and local Gaussian neurons with real or complex weights. The functions' set results $\mathbf{O} = \{f_{PS}, f_{PF}, f_{GR}, f_{GC}\}$. Here,

$$f_{PS}(\mathbf{z}, \boldsymbol{\theta}_{PS}) = \tanh(\sum_{i=1}^{no\_i} w_i z_i + b) \quad (1)$$

corresponds to the standard perceptron (PS) having the inputs $[z_i]_{i=\overline{1,no\_i}}$ and the neural parameters $\boldsymbol{\theta}_{PS} = [w_1,.., w_{no\_i}, b]' \in \Re^{no\_i+1}$ (where w_i and b denote the weight of the i^{th} input connection and the bias, respectively) [6], and

$$f_{PF}(\mathbf{z}, \boldsymbol{\theta}_{PF}) = \tanh(\sum_{i=1}^{no\_i} (\sum_{j=1}^{P} w_{ij}^{c} (\cos \pi j z_i) + \sum_{j=1}^{P} w_{ij}^{s} (\sin \pi j z_i) + w_i z_i + b)) \quad (2)$$

is associated to a global neuron with functional links (PF), assuming the orthogonal trigonometric expansion of maximum order P and the extended set of parameters $\boldsymbol{\theta}_{PF} \in \Re^{1+no\_i\cdot(2P+1)}$ [7]. Also,

$$y_{GR} = f_{GR}(\mathbf{z}, \boldsymbol{\theta}_{GR}) = e^{-\frac{1}{2\sigma^2}\sum_{i=1}^{no\_i}(c_i - z_i)^2} \quad (3)$$

describes the behavior of a Gaussian neuron (GR) with real parameters $\boldsymbol{\theta}_{GR} = [c_1,...,c_{no\_i}, \sigma]' \in \Re^{no\_i+1}$, namely the centers $[c_i]_{i=1,..no\_i}$, and the spread σ [6], and

$$y_{GC} = f_{GC}(\mathbf{z}, \boldsymbol{\theta}_{RC}) = e^{-w^2\left[\left(\sum_{i=1}^{no\_i}\cos\alpha_i\cdot(z_i - c_i)\right)^2 + \left(\sum_{i=1}^{no\_i}\sin\alpha_i\cdot(z_i - c_i)\right)^2\right]}. \quad (4)$$

corresponds to a local neuron with radial basis function and complex weights (GC), which associates for each input connection a center, c_i and a complex weight, $w\, e^{\sqrt{-1}\cdot\alpha_i}$, leading to the set of parameters $\boldsymbol{\theta}_{GC} = [\alpha_1,..,\alpha_{no\_i}, c_1,...,c_{no\_i}, w]' \in \Re^{2no\_i+1}$ [8].

Note that although many homogenous neural networks (such as MLP, RBF) have been proved to be universal approximators of nonlinear, bounded, continuous functions, no construction theorem is available to guarantee a certain approximation performance. Therefore, in practical situations, when dealing with noisy, large training data sets

and neural input of high dimension, the design procedure might fail by selecting an unsatisfactory model. If an adaptive architecture configuration is provided, HNNs can lead to better approximation capabilities than the homogeneous neural networks [2]. However, previous work is limited to the design of HNNs with a single hidden layer, by means of fixed length chromosomes. By employing GP techniques which inherently work on individuals of various sizes and shapes, the present approach provides increased flexibility in neural architecture configuration.

For the sake of simplicity, the models are forced to include a single output neuron. In the case of multi-inputs multi-outputs systems, one can design a distinct neural model for the approximation of each system output, due to the fact that **T** permits to illustrate any possible interdependencies between the plant variables. To ensure non-zero response for a large range of neural inputs, the output neuron is assumed to be global. If lagged values of state variables are included in **T**, the approach can be used for the design of state based models for multivariable nonlinear systems.

Note that **O** and **T** satisfy the closure property [9], as data type consistency and data values consistency are ensured for any possible graph. Consequently, any individual is valid from GP perspective. Moreover, because no other types of nodes are necessary for producing the optimal solution, **O** is minimally sufficient and, if n_u and n_y are chosen large enough, **T** results sufficient, too. Obviously, GP can cope with several alien terminals, due to its capacity of selecting a subset of **T** inside each generated individual, so it is not compulsory to use minimum n_u and n_y, therefore these parameters can be simply tuned by trail and error.

III. Implementation Details

The application uses an explicit representation of nodes/neurons as a combination of nested data structures, with emphasis on a clear distinction between functions and terminals. Following a recursive pattern, `Terminal` and `Function` data structures are conveniently nested inside the more complex `Node` structure, allowing the construction of N-ary trees and a more efficient implementation of the genetic operators. The `Function` structure includes parameters which help keeping the track of nodes' children stored and managed by means of a dynamically allocated array of pointers to `Node` structures. Additionally, specific data structures allow operations over an individual from a neural network perspective, enabling it to be seen as a multilayered architecture. This is achieved by grouping together the tree nodes found at the same depth level and by allowing the traversing of such individuals layer by layer, instead of the conventional stack (queue) based or recursive techniques. The `NodeArray` structure keeps track of all tree nodes for a given depth, while the `NeuralNet` structure holds multiple layers (`Node` arrays), thus making possible to visit all inner nodes with two iterators, once a tree individual has been ”layerized”.

The stochastic procedure uses the `glibc` random number generator from [http:// www.gnu.org/ software/ libc/‖glibc] (linear modulo 32). Suitable functions provide an interface with the random source, to allow the selection of `Functions` and `Terminals`, as well as the initialization of the other node parameters (such as bias, weights, lags, etc). Note that the use of a mersenne twister or the linux random number source devices (/dev/random and /dev/urandom) is also possible (due to an abstract interface). When a new individual is created, its root node will have to be a global `Function` node. The depth value is randomly chosen and then, for each node, the type is stochastically generated, along with its corresponding parameters (e.g. number of children). Afterwards, the resulted tree-based individual is translated to a layered representation, via a procedure which joins together the pointer arrays inside the `Function` structures. The nodes from successive layers are randomly interconnected and the resulting connections are maintained for the life time of the individual. A distinction is made between the normal, direct connections which are established when a `Function` node is initially created (towards nodes which are regarded as its children), and the extra-connections that are established afterwards, between the `Function` node and the children of other nodes situated at the same level. This distinction is important because it allows several optimizations (e.g., the recursive procedures can visit each node only once) and helps the crossover to swap the selected sub-graphs.

As the main concept of the application is the modular development of the neural networks by means of GP-based optimization, one has to guarantee that, during the evolutionary loop, each graph-based individual encodes a valid configuration of neurons and terminals. In this context, validity means that the main traits of a neural network must be preserved (e.g., formally approved/established activation functions, along with neural parameters such as bias/dispersion of neurons, weights of incoming connections, etc). The activation functions have to be implemented according to a predefined calling convention, by involving another customized data structure, `NodeInputs`, which is used to pass functions’ parameters whenever the evaluation of an individual is performed. To support a modular design of the neural network and a simple management of different types of neurons, the program makes use of a dispatch table, namely a function pointer array which holds the addresses of all allowed activation functions. In this configuration, adding a new type of neuron is trivial, as it all comes down to defining the new activation function and adding it to the dispatch table.

This extra flexibility comes with the cost of an indirect call of the activation functions, which might translate into small performance penalties, but most modern x86/x64 CPUs can perform branch prediction on indirect calls, so the effect should be minimal and barely noticeable, if any. On the other hand, a dispatch table offers the flexibility of adding or removing such functions, without having to change other areas of the program.

The GP application is organized into several sub-modules which perform operations on `Terminals` and `Functions`, `Nodes` (of one of the two types), DAG-encoded `Individuals` and populations of individuals. This hierarchical organization lends itself to good scalability, making the program able to process very large graphs (with numerous nodes/interconnections), as basis for delivering good results for a variety of optimization problems. Additionally, the genetic operators are required to work on highly interconnected neural structures. Therefore, the design procedure has to be able to process a large number of incoming and outgoing neuron connections, the goal being to reuse as much information as possible in the resulted DAG-based offspring.

In the case of structural crossover, randomly selected sub-graphs are swapped between the parents. This requires specific preparations for the sub-graph externalization, which basically means that its connections with the rest of the parent's nodes must be replaced in both directions (incoming and outgoing). Let us denote the root of a sub-graph with *G*. The algorithm relies on the following statements: (S1) a node *N* is called directly included in *G* sub-graph, if there exists a path formed only with normal connections provided from *G* to *N*; (S2) a node *M* is named indirectly included in graph *G*, if it is not directly included in *G* and there is a path of normal and extra-connections from *G* to *M*. Fig. 1 explains the crossover algorithm, assuming that the sub-graph having the root node 7 has been selected for swapping with the second parent. Firstly, according to (S1) and (S2), the procedure identifies the subset of nodes indirectly included in the selected sub-graph (marked with in red Fig 1). Afterwards, following a one-to-one mapping, a hash table is created, having the subset nodes as keys, and copies of those nodes as values. From the sub-graph's perspective, in hash table terms, all outgoing connections towards keys must be replaced with connections towards values, so that the sub-graph will carry within itself the same information when swapped, without disrupting the configuration that is left behind. The replacement procedure employs special checks in order to preserve the consistency of the DAG-encoded individual, because from the sub-graph's perspective, in some cases extra connections towards the keys must become normal connections towards values and, at the same time, the

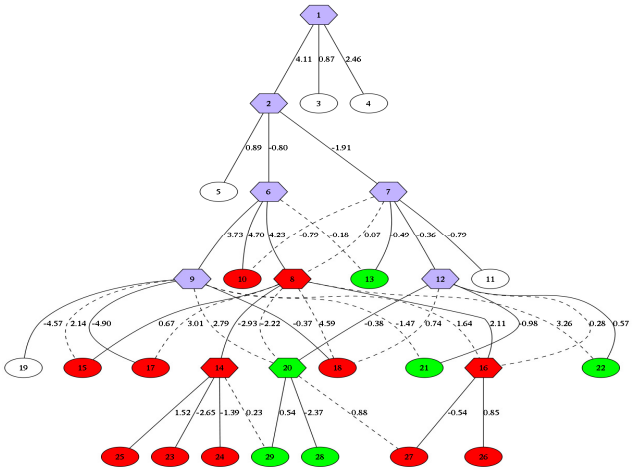


Figure 1. DAG-encoded individual selected for crossover. The neurons are marked with hexagons, the terminal with ellipses; the normal connections are denoted with solid lines and extra-connections (introduced at “layerization”) are represented by dotted links.

values can be connected together themselves, with either a normal or an extra connection. Let 8' be the copy of node 8 (the value associated to the key), node 10' be the copy of node 10, and so on. The extra-connections from node 7 towards nodes 8 and 10 will be replaced with normal connections towards nodes 8' and 10'. Then, node 8' will be connected with node 18'. However, since node 12 also has an extra-connection towards node 18', in order to preserve the consistency of the encoding, one of the nodes (node 12) will be connected to node 18' using a normal connection, while the other one (node 8') will have an extra connection towards node 18'. The procedure stops when all nodes in the sub-graph and all nodes in the hash table have been visited, and all connections have been replaced. Once the algorithm has worked out the connections that need to be replaced from the sub-graph's standpoint, it switches to an ”outside” view, identifying those nodes belonging to the selected sub-graph, which have incoming connections from nodes not belonging to the selected sub-graph. These connections will have to be replaced as well, in order to completely isolate the selected sub-graph and allow it to be swapped. The subset consisting of nodes 13, 20, 21, 22, 28, 29 is identified and again, a hash table is used. Applying the same replacement rules, node 6 will be connected with node 13' (the connection will no longer be an extra connection but a normal one), node 8 will be connected with node 20', and in turn, node 21' will have to be connected to nodes 28' and 29', using normal connections. Node 14 will receive an extra connection towards node 29' and nodes 9 and 12 will receive extra connections towards node 20'. Finally, node 9 is connected to nodes 21' and 22', using normal connections. Afterwards, the crossover involves an exchange of pointers from one side to the other.

It is already known that the crossover has the tendency of

producing larger and larger individuals, even without ensuring significant improvement of the objective values. This phenomenon, called bloat, may lead to the production of overfitted individuals, with expected bad generalization capabilities. As a preliminary protection of individuals' parsimony, the application eliminates all the solutions which exceed a predefined depth. Note that, additionally, due to the way the extra-connections could be processed within the DAG- based individuals, the structural crossover might lead to a "horizontal expansion", displayed by increased number of normal links and, therefore, increased number of nodes. Another downside of the crossover is the competing conventions problem. It refers to the possibility of producing less adapted offspring, when working on different fitted parents, which encode the same neural network (for instance, when some neural building blocks are permutated in the selected DAGs). In this context, the proposed GP application considers different types of mutation, applied with higher probabilities. Being unary operators, the mutations permit a better control on the complexity of the resulted offspring, and avoid competing conventions' problem occurrence. The parametric mutation randomly selects a node and alters the values of its corresponding parameters, whilst the link mutation adds extra-connections from function nodes/neurons to the children of other neurons situated on the same level. The node mutation changes the type of randomly selected nodes. When a `Function` node turns into a `Terminal` one, some extra consideration has to be given to the connections which might exist from outside nodes towards the children of the `Function` node, which will be structurally mutated. As the children of the node will no longer exist, all references from the outside nodes towards them must be removed, or copies must be provided. Lastly, the structural mutation replaces a stochastically selected sub-graph with a randomly generated one.

The evaluation procedure performs a recursive traversal of a DAG-encoded individual, by propagating the results of the computations upward, from the bottom of the graph towards the root node. When a `Terminal` is reached, the corresponding value is read based on its lag, the input label and the current sampling time. After all terminals return their values, the evaluation procedure calls the activation function for the `Function` node that they are connected to and the result is stored in a variable. Since the procedure is dealing with a large number of extra connections, a simple cache system is used: it sets an appropriate flag when a node is firstly visited, so that ulterior visits will retrieve the already saved result without wasting CPU time by re-evaluation. Because during a GP generation, one of the most important time consumers is the computation of the objective values, the application uses pthreads [https:// computing.llnl.gov/ tutorials/ pthreads/ #Overview] to distribute this task among the available CPU cores. Depending on the number of existing cores, the population (a `NodeArray` structure) is partitioned, by means of indexes and offset variables, so that each thread can perform the calculations independently (the threads are joined after evaluation of every individual). Additionally, the program can export tree individuals to plain text files using the dot language to represent graph structure (http://www.graphviz.org). Using the opensource Graph Visualization Software (Graphviz), the files can be converted to a wide variety of graphic formats.

IV. Application

The software package has been verified on Lorenz attractor time series prediction. The training and validation data sets, each one consisting of 250 samples, indicate different state trajectories collected for distinct initial conditions ([0.1,0.1,0.1] and [0.3,0.3,0.3], respectively), considering the chaotic state model $\dot{x} = 10(y - x)$, $\dot{y} = -xz + 27x - y$, $\dot{z} = xy - 8/3 * z$ and the sampling period $T_s = 0.1$. The HNN is designed to predict x trajectory. Therefore, $\mathbf{T}$ contains lagged values corresponding to all system state variables, $\mathbf{T} = [x(k-1), y(k-1), z(k-1), \ldots\ldots x(k-n_x), y(k-n_y), z(k-n_z)]$.

The main challenge for the application lies in delivering the high performances of accuracy and generalization required for the selected model, in compliance with the chaotic nature of Lorenz system. Several algorithm parameter sets were used to illustrate the behavior of the proposed GP-based approach (Table I). The population was initialized with simple HNNs having maximum three incoming connections per neuron, as the interconnectivity may significantly increase during the evolutionary loop, by means of genetic operators. Table I indicates the average accuracy performances, as well the quality of the best model resulted during 10 independent runs. Here, *MSE_L* and *MSE_T* represent the mean output squared errors obtained over scaled learning and testing data, respectively. For the sake of simplicity, one considers $n_x = n_y = n_z = n$. If $n = 1$, $\mathbf{T}$ results minimally sufficient (#1, #3) and the risk of overfiting is reduced. Although it seems simpler to find proper models by exploring fewer potential neural topologies, note that the individuals' performances reveal the symbiosis between structure and parameters, so GP can discard HNNs with appropriate architecture, yet inconvenient parameters. Consequently, the final model could be sometimes simpler than necessary, leading to improper *MSE_L*. This downside could be diminished when working on larger populations (#1 vs. # 3, #5) and, obviously, the hybridization with certain local optimization procedures could be beneficial. If $\mathbf{T}$ includes alien lagged

state variable values ($n > 1$), the use of small d permits to eliminate excessively complex individuals expected to include unnecessary intron building blocks, yet the risk producing overfitted individuals remains high and, often, the best selected models display worst performances on validation (#2, #4). The exploration can be also enforced by working on larger and diverse populations, although increasing the population size (and/or the number of generations) is not always a guarantee for performance improvement. Stress testing (#5, #6) reveals the capacity of the algorithm of working on large batches of neurons (about 230000 per generation), whilst preserving reasonable time performances (about 120-200 sec total execution time) and memory consumption (about 65 MB). All trials were carried out on a configuration with Code 2 Duo P7350 (2GHz, 3MB cache) and 3GB RAM. Note that each `Function` data structure needs 40 bytes and each `Terminal` structure requires 12 bytes. Lastly, the performances of the selected model #4 (including one PS output neuron and 2 hidden GR neurons) were compared with those provided by homogeneous neural networks having the same number of hidden neural units: an MLP trained for 5000 epochs with Levenberg-Marquardt algorithm, and an RBF designed by means of a constructive algorithm (which iteratively adds GR neurons). The designed HNN features better prediction and better generalization capabilities (Fig.2), making use of its particular compact partially interconnected heterogeneous structure.

TABLE I. EXPERIMENTAL RESULTS

#	N_{ind}/N_{gen}	n	d	Average values		Best model	
				MSE_L	*MSE_T*	*MSE_L*	*MSE_T*
1	1000/300	1	3	7.2	1.3	3.22	0.41
2	1000/300	3	3	1.1	1.69	0.31	0.54
3	5000/300	1	3	3.41	0.63	0.78	0.38
4	5000/300	3	3	0.76	1.04	0.39	0.44
5	10000/300	1	3	2.3	0.36	0.40	0.24
6	10000/300	1	4	0.94	0.27	0.63	0.34

Nind/Ngen = population size/ number of generations; *n*=maximum lag; *d*= maximum depth of graphs

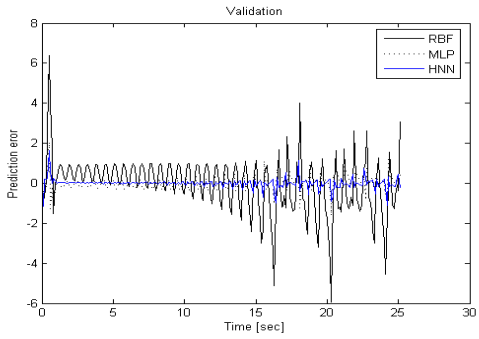


Figure 2. The prediction error provided by HNN /MLP/ RBF on non-scaled validation data set. Resulted mean output squared errors are, correspondingly, 15.08 / 35.2 / 458.6.

Comparable performances of accuracy and generalization can be achieved with larger homogeneous neural networks, including about 4 hidden PS, or 15 GR.

V. CONCLUSIONS

The proposed C based software permits a flexible selection of hybrid neural models, working concomitantly on the structure and the parameters of the model, by means of GP techniques. The partially interconnected architectures accepting any combination of global and local neurons organized in multiple hidden layers are encoded by means of DAGs, thus leading to an efficient reuse of information within the hierarchical individuals.

To ensure the validity of all constructed models both from phenotypic and genotypic standpoint, the prototype of the input-output mappings corresponding to all allowed neurons are directly included in the set of functions and special genetic operators are designed to manage the inner parameters of the nodes. This involves the management of normal and extra-connections for a correct sub-graphs swapping or reconfiguration by means of crossover or mutation. The implementation assures a low memory consumption and fast evaluation of individuals, leading to the possibility of working on massive and highly interconnected structures. The experimental trials indicate the ability of the suggested approach to solve difficult identification/modeling problems, while dealing with scarce aprioric information and severe requirements of accuracy.

REFERENCES

[1] P. J. Flemming, R. C. Purshouse "Evolutionary Algorithms in Control Systems Engineering: A Survey", *Control Engineering Practice* 10, 2002, pp. 1223-1241.

[2] L. Ferariu, M. Voicu, "Nonlinear System Identification Based on Evolutionary Dynamic Neural Networks wih Hybrid Structure" in *Proc. of IFAC Congress,* Prague, Czech Republic, 2005.

[3] Poli R., Langdon W. B., Mc Phee N. F., *A Field Guide to Genetic Programming*. Published via http://lulu.com (with contributions of J.R. Koza), [Online]. Available: http://www.gp-field-guide.org.uk, 2008.

[4] Affenzeler M., Winkler S., Wagner S., Beham A., *Genetic Algorithms and Genetic Programming: Modern Concepts and Practical Applications (Numerical Insights)*, CRC Press, 2009. J.

[5] A Walker and Miller, J. F. (2008). The Automatic Acquisition, Evolution and Reuse of Modules in Cartesian Genetic Programming. *IEEE Transactions on Evolutionary Computation*, 12 (4), 397-417.

[6] S. Haykin,. *Neural Networks - A Comprehensive Foundation*, McMillan College Publishing Company, New York, 2nd Edition, 1999.

[7] Patra J., Pal R., Chatterji, B., Panda G., "Identification of nonlinear dynamic systems using functional link artificial neural networks", *IEEE Transactions on System, Man and Cybernetic*s, Part B: Cybernetics, 29, 254–262, (1999).

[8] Igennik B., Tabib – Azar M., Le Clair S. R., "A net with complex weights", *IEEE Transactions on Neural Networks*, 12, 236–249, 2001.

[9] J. R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selectio*n. Cambridge, MA: MIT Press, 1992.

Experimental Studies Regarding for Faults Detection Using Residuals Generator Method

G. Canureci, M. Vinatoru, C. Maican

Abstract— **In this paper one will develop the faults detection and localization method using residual vectors, in order to emphasize the disturbances and faults on the outputs L_1 and L_2 of the control level plant with two coupled tanks "Quanser Water Level Control Two Tank Module". The design method of the residual generator, developed on Quanser plant, was implemented at coupled tanks system. For this reason there were used the facilities of Quanser acquisition and control system with acquisition card and the external connection mode at the transducers and actuator's.**

The control system level in tank 2 was developed under Matlab Simulink with a PI controller with feed forward response to control the modules through Q4DAQ water recirculation pump. Depending on the fault followed to be detected one can make configurations to define the residues with the proper implementation of the weighting matrices.

The proposed method was theoretically developed and experimentally verified in this plant and allowed detection and localization of two faults created in a real plant.

I. Introduction

A fault is any kind of malfunction in a system that is a deviation from the normal behavior in the plant or its instrumentation [1]. Fault detection is the indication that something is going wrong in the monitored system. Fault isolation is the determination of the exact location of the fault (the component which is faulty) [2, 7, 8].

Detection and isolation of faults are very important tasks in intelligent control systems because a fault can lead to a reduction in performance or even to breakdowns or catastrophes.

The laboratory plant Quanser can be the subject of a series of additive faults, disturbances and noises. The faults and disturbances can appear in processing equipment (example: the breaking of a pipe in the tank) or in auxiliary equipments (example: *faults of the transducers and the faults of the actuators)*. These may lead to a deterioration of the performances in the system and will also have an impact on safety, productivity and cost reductions of the plant [6].

Therefore, it is necessary to evaluate the performance of the system and to realize a diagnosis of the causes that may affect the loss or decrease in performances based on a model of detection and identification of the faults.

In this paper the structured residual approach is applied to obtain estimates of the envelopes for detecting faults in a coupled - Tanks System. This system has the typical characteristics of tanks, pipelines and pumps used in many industries. In this case, the faults are clogging and leakages.

This paper is organized as follows. In section 2 it is presented the residual generator method; the system under study which is the two tank system is described and it is presented in section 3. In this section, the identification of unmeasured disturbance variables (faults) using residual approach as reported in literature is explained. In section 4 it is presented the mathematical model of the plant in discrete form. In section 5 it is presented the design of the residual generator. In section 6 there are discussed the cases studied. Finally the conclusions are drawn and the purpose of further work is provided in section 7.

II. Residual Generator Method

Residuals are generated from the observable variable of the monitored plant that is, from the command values of the controlled inputs and outputs [1]. However, the presence of disturbances, noise and modeling errors also causes the residuals to become non-zero and thus to interfere with the detection of faults. Therefore the residual generator needs to be designed so that it is maximally unaffected by these nuisance inputs, which means that it is robust in the face of disturbance, noise and model errors.

Residuals generator method used in this paper was developed by Asokan A. and D. Sivakumar in 2007. Structured residual are designed in such a way that each residual responds to a different subset of faults and insensitive to the others.

When a particular fault occurs, some of the residuals respond and others do not. Then the pattern of the response set and the fault signature or fault code are characteristic to the particular fault [1, 6].

Example of fault code:

	R1	R2
L1	1	0
L2	0	1

The above fault code implies that fault L1 affects only residual R1 like L2 affects R2. In order to perform detection and isolation of set of faults, structured residuals can be used. The so called signature code describes the subset of residuals which react to each fault. The level is the same in the two tanks. When there is a fault the two residuals get affected. By simply monitoring the residuals it is possible to predict the change in behavior of the system from normal. But it is not possible to identify the location of the fault. So the residual has to be transformed to enhance isolation.

Let the plant output be [1]:

$$Y_p(q) = H(q)\,U(q) + G_D(q)\,L(q) \quad (1)$$

Where: H (q) is the transfer function under normal conditions, G_D (q) is fault transfer function, and U (q) is input to the system, D (q) fault input (leak) in the system.

Let the output of the model be given by:

$$M_y(q) = H(q)\,U(q) \quad (2)$$

Residual *R* (q) is defined as difference between process output and model output.

$$R(q) = Y_{ap}(q) - M_y(q) \quad (3)$$

Substituting the expressions for $Y_p(q)$ and $Y_m(q)$ in (3) result:

$$R(q) = G_D(q)\,L(q) \quad (4)$$

The above equation is valid only in the absence of the plant model mismatch and in the absence of state and measurement noise. From the above equations, it is obvious that the presence of fault will affect all the two residuals. It is difficult to identify the location of fault by monitoring the residuals.

Therefore in order to enhance the fault isolation it is required to transform the residuals. To transform raw residual R(q) into structured form $R_1(q)$, multiply R(q) with weighting matrix W(q) [1, 3, 4]:

$$R_1(q) = W(q)R(q) \quad (5)$$

Weighting matrix is chosen as to contain inverse of identified fault model to cancel the effects of fault transfer function present in residual.

$$W(q) = Z(q)G_D^{-1}(q) \quad (6)$$

where Z(q) is user defined matrix.

The method is based on the difference between the real plant with fault and the model without the fault. This difference is influenced both by the faults, the disturbances and the noises. Weighting matrix W(q) inserted to define structured residues R(q) is calculated so as to eliminate the influence of the disturbance and of the noises so that R(q) depends only by the faults.

In some situations, this is not always possible because it can result in improper forms for the matrix W(q), but in each case, small changes can be made by introducing the poles with the minimal influence on the dynamic system so that to ensure its physical reliability. These problems have been verified on the laboratory physical plant, presenting the experimental results for a number of cases of the faults, the disturbances and the noises which can appear in processing.

III. Description Of The Laboratory Plant

The laboratory plant for control level consists in a system with two coupled tanks [1, 4]. The coupled tanks system is a module "Quanser water level control two tank module" (Fig. 1), consisting in two tanks, a pump and a water basin; the two reservoirs are mounted on the platform, so that the liquid in the first tank is leaking through a hole located at its base in the second tank and finally in the water basin situated to the base platform.

The transported water flow of recirculation pump feeds simultaneously the two reservoirs $R_{superior}$ and $R_{inferior}$ (denoted R_1 and R_2) through the valves S_1 and S_2. The total flow of the pump controlled by the U_C voltage will be considered as the input variable u_2.

The laboratory plant contains two coupled tanks addition, a data acquisition card UPM 2405 Quanser with 4 analogical inputs and 4 analogical outputs, WinCon software and RTW (Real Time Workshop) for the real time simulation.

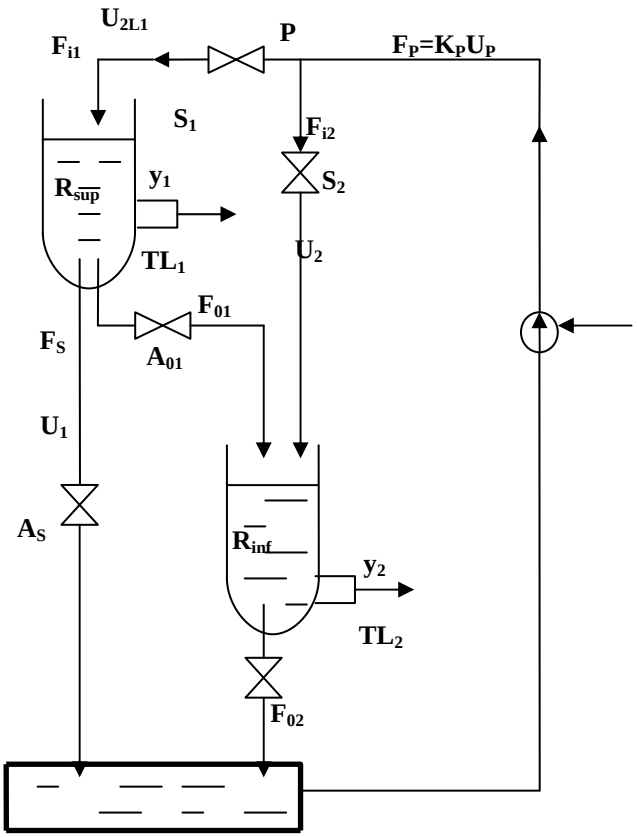


Fig. 1. The Hydraulic diagram of the plant's control level.

IV. Mathematical Model Of The Plant In Discrete Form

In this section one can see the following aspects:

- the residuals are defined as the difference between measured variables in a real plant and the equivalent variables in the faultless model of the plant; the input model applies the same command variables as the real plant;
- the proposed algorithm follows as faults detection and

localization system to be implemented on the same computer (PLC – Programmable Logical Controller), which provides the control functions of the technological plant; in this way, they are directly accessible in the computer both real measured variables of the process, and the command variables provided by the control algorithm implemented on PLC;

- At the same time, the model should easily reduce the additional requirements of data required by the PLC;

- in order to implement the faults detection and isolation diagram it is necessary to determine the mathematical model of the plant as accurate as possible, to emphasize the influence of the faults and the disturbances on observable variables and then the reduction of the processing requirements.

As stated above, for defining the residuals, a mathematical model of the laboratory plant had to be developed.

This model was defined by (7) and (8), highlighting the command variable, the disturbances and the possible faults which may occur in the plant [1, 6, 7, 8].

The nonlinear mathematical model for the two reservoirs is:

$$\frac{dL_1}{dt} = \frac{K_P(U_P - U_{Pd}) - A_{01}\sqrt{2g}\sqrt{L_1} - A_{0d}\sqrt{2g}\sqrt{L_1}}{A_{t1}} \quad (7)$$

$$\frac{dL_2}{dt} = \frac{-A_{02}\sqrt{2g}\sqrt{L_2} + A_{01}\sqrt{2g}\sqrt{L_1}}{A_{t2}} \quad (8)$$

Where: A_{t1} and A_{t2} are transversal sections of the tanks, F_{i1}, F_{i2} are the volumetric inflow rate of tank 1 and tank 2: $F_{i1}=k_pU_p$, $F_{i2}=F_{o1}$; F_{o1}, F_{o2} the outflow rate from tank 1 and respectively tank 2; the outlet cross-section area of tank 1 and tank 2 can be calculated by the relations: $A_{o1}=1/4\,\pi\,D^2_{o1}$, $A_{o2}=1/4\,\pi\,D^2_{o2}$, D_{o1}, D_{o2} outlet diameter in tank 1 and tank 2.

These equations are linear; leading to the complete discrete form (9) and (10), written in generalized variables and can be easily used in defining the residuals [2].

He used the discrete method using share associated rows With this method we obtained the best results [5].

In the relations (9) and (10), $y_1(t)$ and $y_2(t)$ represent the measured values of the transduscers which are used in the structure of the residual generator and in here appear faults of the transdurcers $\Delta y_1(t)$ and $\Delta y_2(t)$.

Through the internal model of the process and of the model these faults do not appear [2].

$$y_1(t) = -\frac{10{,}307\,q^{-1}}{1-0{,}8767\,q^{-1}}\cdot u_1(t) + \frac{0{,}0937\,q^{-1}}{1-0{,}8767\,q^{-1}}\cdot u_2(t) + \Delta y_1(t) - \\ - \frac{10{,}307\,q^{-1}}{1-0{,}8767\,q^{-1}}\cdot \Delta u_1(t) - \frac{0{,}0937\,q^{-1}}{1-0{,}8767\,q^{-1}}\cdot \Delta u_2(t) \quad (9)$$

$$y_2(t) = \frac{0{,}0580\,q^{-1}}{1-0{,}9363\,q^{-1}}\cdot y_1(t) + \frac{0{,}0967\,q^{-1}}{1-0{,}9363\,q^{-1}}\cdot u_2(t) - \\ - \frac{0{,}0967\,q^{-1}}{1-0{,}9363\,q^{-1}}\cdot \Delta u_2(t) + \Delta y_2 - \frac{0{,}0580\,q^{-1}}{1-0{,}9363\,q^{-1}}\cdot \Delta A_2(t) \quad (10)$$

The next notations are being made:

$$H_1(q) = \frac{q^{-1}}{1-0{,}8767\,q^{-1}} \;;\; H_2(q) = \frac{q^{-1}}{1-0{,}9363\,q^{-1}} \quad (11)$$

In these equations, the generalised variables have the following physical meanings:

$y_1(t)\rightarrow$ is equivalent to $\Delta L_1(t)$: variation of the measured level transduscer in the tank 1 around the values of steady states L_{10}=15 cm; $y_2(t)\rightarrow$ is equivalent to $\Delta L_2(t)$: variation of the level measured by the transduscer in tank 2 around of the steady state values L_{20}=15 cm; $u_2(t)\rightarrow$ is equivalent with the variation $\Delta U_P(t)$: control of the regulator; $u_1(t) \rightarrow$ is equivalent to $\Delta A_S(t)$ variation: continous disturbance by flow F_P; $\Delta y_1(t) \rightarrow$ is equivalent to fault transduscer L_1; $\Delta y_2(t)\rightarrow$ is equivalent to fault transduscer L_2; $\Delta u_1(t)\rightarrow$ fault broken pipe equivalent in the plant with additional flow ΔF_S; $\Delta u_2(t)\rightarrow$ pump fault to corresponding at normal regime $\Delta F_P = K_P\Delta U_P$; $\Delta A_2(t)\rightarrow$ the disturbance equivalent with the variation evacuation section of the liquid in tank 2.

It can be expressed $y_2(t)$ depending only by the command, disturbance and the faults $\Delta u_1(t)$, $\Delta u_2(t)$, $\Delta y_2(t)$, $\Delta A_2(t)$ bringing the relation (10) to form (11):

$$y_2(t) = -\frac{0{,}597q^{-2}}{1-1{,}813q^{-1}+0{,}82q^{-2}}\cdot u_1(t) + \frac{0{,}876q^{-1}-0{,}079q^{-2}}{1-1{,}813q^{-1}+0{,}82q^{-2}}\cdot u_2(t) + \\ + \Delta y_2(t) - \frac{0{,}597q^{-2}}{1-1{,}813q^{-1}+0{,}82q^{-2}}\cdot \Delta u_1(t) - \frac{0{,}876q^{-1}+0{,}815q^{-2}}{1-1{,}813q^{-1}+0{,}82q^{-2}}\cdot \Delta u_2(t) - \quad (12) \\ - \frac{0{,}058q^{-1}+0{,}05q^{-2}}{1-1{,}813q^{-1}+0{,}82q^{-2}}\cdot \Delta A_2(t)$$

The numerical values calculated of the coefficients according to (9) and (12) are:

K_{u21}=0,0937 ; K_{y1y2}=0,0580 ; K_{u1}=-10,307 ; K_{du22}=-0,0967
K_{du1}=-10,307 ; K_{u22}= 0,0967 ; K_{du21}=-0,0937

Under these conditions, according to the defined residuals, the equations (9) and (12) describing the real plant with the faults are defective:

$$y_{1r}(t) = -K_{u1}\cdot H_1(q)\cdot u_1(t) + K_{u21}\cdot H_1(q)\cdot u_2(t) + K_{du1}\cdot H_1(q)\cdot \Delta u_1(t) + \\ + K_{du21}\cdot H_1(q)\cdot \Delta u_2(t) + \Delta y_1(t) \quad (13)$$

$$y_{2r}(t) = K_{y1y2}\cdot H_2(q)\cdot y_1(t) + K_{du22}\cdot H_2(q)\cdot \Delta u_2(t) + K_{u22}\cdot H_2(q)\cdot u_2(t) - \\ - K_{y1y2}\cdot H_2(q)\cdot \Delta A_2(t) + \Delta y_2(t) \quad (14)$$

where $\Delta A_2(t)$ is the disturbance applicated through tank 2.

The mathematical model of the technological plant considered free of the faults resulting of (13) and (14), negating the corresponding terms of the faults $\Delta u_1(t)$, $\Delta u_2(t)$, $\Delta y_1(t)$, şi $\Delta y_2(t)$, results in the forms (15) and (16):

$$y_{1m}(t) = K_{u21} \cdot H_1(q) \cdot u_2(t) - K_{u1} \cdot H_1(q) \cdot u_1(t) \tag{15}$$

$$y_{2m}(t) = K_{y1y2} \cdot H_2(q) \cdot y_1(t) + K_{u22} \cdot H_2(q) \cdot u_2(t) \tag{16}$$

If the disturbances $u_1(t)$, $\Delta A_2(t)$ are unkenown , then these should not be included in the model, and will be detected as faults.

Under these conditions we shall obtained:

$$\begin{aligned} &y_{1m}(t) - y_{1r}(t) = K_{u21} \cdot H_1(q) \cdot u_2(t) - K_{u1} \cdot H_1(q) \cdot u_1(t) - K_{u21} \cdot H_1(q) \cdot u_2(t) - \\ &- K_{du1} \cdot H_1(q) \cdot \Delta u_1(t) - K_{du21} \cdot H_1(q) \cdot \Delta u_2(t) - \Delta y_1(t) = \\ &= K_{u1} \cdot H_1(q) \cdot u_1(t) - K_{du1} \cdot H_1(q) \cdot \Delta u_1(t) - K_{du21} \cdot H_1(q) \cdot \Delta u_2(t) - \Delta y_1(t) \end{aligned} \tag{17}$$

$$\begin{aligned} &y_{2m}(t) - y_{2r}(t) = -K_{y1y2} \cdot H_2(q) \cdot y_1(t) + K_{u22} \cdot H_2(q) \cdot u_2(t) + \\ &+ K_{y1y2} \cdot H_2(q) \cdot y_1(t) - K_{du22} \cdot H_2(q) \cdot \Delta u_2(t) - \\ &- K_{u22} \cdot H_2(q) \cdot u_2(t) + K_{y1y2} \cdot H_2(q) \cdot \Delta A_2(t) - \Delta y_2(t) = \\ &= -K_{du22} \cdot H_2(q) \cdot \Delta u_2(t) + K_{y1y2} \cdot H_2(q) \cdot \Delta A_2(t) - \Delta y_2(t) \end{aligned} \tag{18}$$

V. Design Of The Residual Generator

There are defined the residues **$r(q)=y_m(t)-y_r(t)$** end **$R(q)=W(q)*r(q)$** under the form [1, 2, 4]:

$$r(q) = \begin{bmatrix} r_1(q) \\ r_2(q) \end{bmatrix} \quad R(q) = \begin{bmatrix} R_1(q) \\ R_2(q) \end{bmatrix} \tag{19}$$

$$r_1(q) = [y_{1m}(t) - y_{1r}(t)] =$$

$$= \begin{bmatrix} -K_{u1} \cdot H_1(q) \cdot u_1(t) - K_{du1} \cdot H_1(q) \cdot \Delta u_1(t) - \\ -K_{du21} \cdot H_1(q) \cdot \Delta u_2(t) - \Delta y_1(t) \end{bmatrix} \tag{20}$$

$$r_2(q) = [y_{2m}(t) - y_{2r}(t)] = \begin{bmatrix} K_{du22} \cdot H_2(q) \cdot \Delta u_2(t) - \\ -K_{y1y2} \cdot H_2(q) \cdot \Delta A_2(t) + \Delta y_2(t) \end{bmatrix} \tag{21}$$

where r(q) is the residual vector, R(q) is the output of the residuals generator, W(q) is a weighting matrix that provides some properties of the residuals generator (isolation).

If the disturbances $u_1(t)$ and $\Delta A_2(t)$ are measurable, they will be introduced in the model and will appear in the expression of the residuals.

The following situations will be analyzed, situations in which the disturbances are unmeasured (they will not be introduced in the model and thus appear in the residuals expressions).

In (21) one will consider the matrix $W_1(q)=1$.

It is considered the definition of a residue **R(q) with a weighting matrix** W(q) to provide decoupling in relation to the faults.

The new residue **R(q)** will be under the form [1]:

$$R(q) = W(q)r(q) = \begin{bmatrix} Z_{11}(q) & 0 \\ 0 & Z_{22}(q) \end{bmatrix} \begin{bmatrix} \Delta u_2(t) \\ \Delta u_1(t) \end{bmatrix} \tag{22}$$

Considering the residuals $R_1(q)$ and $R_2(q)$ that depend on both faults $\Delta u_2(t)$ and $\Delta u_1(t)$ and both disturbances ΔA_1 and $u_1(t)$. From the relations (20) and (21) resulting the dependencies of the residues r_1 and r_2 (defined in table 1) and the faults $\Delta u_1(t)$, $\Delta u_2(t)$, $\Delta y_1(t)$, $\Delta y_2(t)$, the disturbances $\Delta A_2(t)$ and $u_1(t)$ presented in table 1.

TABLE I

Dependency residues – faults -disturbances

	---faults----				--disturbances-	
Residual/fault disturbance	Δu_1	Δu_2	Δy_1	Δy_2	u_1	ΔA_2
r_1	1	1	1	0	1	0
r_2	0	1	0	1	0	1

From the analysis of table 1 it results 3 separate columns, with the residues r_1 and r_2 defined by (20) and (21) and 3 cases can be detected:

Case 1: - detection of Δu_2 fault as modified simultaneously the residues r_1 and r_2 (combination 1 1);

Case 2: - detection of Δu_1 or Δy_1 fault or the u_1 disturbance (combination 1 0);

Case 3: - detection of Δy_2 fault or a disturbance ΔA_2 (combination 0 1).

It is noted that the two disturbances u_1 and ΔA_2 are known and are included in the model.

Then can be detected 3 of the 4 faults considered, namely: Δu_1 or Δy_1 influencing only the residue r_1, Δy_2 fault influencing only the residue r_2 and Δu_2 fault influencing simultaneously the residues r_1 and r_2 [1, 5, 7].

Under these conditions the residues r_1 and r_2 are defined as [1, 2].

$$\begin{bmatrix} r_1(t) \\ r_2(t) \end{bmatrix} = \begin{bmatrix} H_1(q) & 0 \\ 0 & H_2(q) \end{bmatrix} \begin{bmatrix} K_{du1} & K_{du21} & 1 & 0 & K_{u1} & 0 \\ 0 & K_{du22} & 0 & 1 & 0 & K_{y1y2} \end{bmatrix} U(t) \tag{23}$$

Where the transfer functions $H_1(q)$ and $H_2(q)$ have the values presented in (11).

Defining the combined input of the residual generator:

$$U(t) = [\Delta u_1(t) \quad \Delta u_2(t) \quad \Delta y_1(t) \quad \Delta y_2(t) \quad u_1(t) \quad \Delta A_2(t)] \tag{24}$$

then the form of the matrix equation from the residual generator is:

$$\begin{bmatrix} r_1(t) \\ r_2(t) \end{bmatrix} = \begin{bmatrix} H_{du1_1}(q) & H_{du21_1}(q) & H_{y1}(q) & 0 \\ 0 & H_{du22}(q) & 0 & H_{y2}(q) \end{bmatrix} \begin{bmatrix} \Delta u_2(t) \\ \Delta u_1(t) \\ \Delta y_1(t) \\ \Delta y_2(t) \end{bmatrix} \tag{25}$$

Where:

$$H_{du1}(q)=\frac{10{,}307\,q^{-1}}{1-0{,}8767\,q^{-1}};\quad H_{du22}(q)=\frac{0{,}0967\,q^{-1}}{1-0{,}9363\,q^{-1}}$$
$$H_{y1}(q)=\frac{q^{-1}}{1-0{,}8767\,q^{-1}};\quad H_{y2}(q)=\frac{q^{-1}}{1-0{,}9363\,q^{-1}} \qquad (26)$$
$$H_{u1}(q)=\frac{10{,}307\,q^{-1}}{1-0{,}8767\,q^{-1}}$$

It is defined the fault matrix $G_D(q)$ in the form:

$$G_D(q)=\begin{bmatrix} H_{du1}(q) & H_{du21}(q) & H_{y1}(q) & 0 & H_{u1}(q) & 0 \\ 0 & H_{du22}(q) & 0 & H_{y2}(q) & 0 & H_{A2}(q) \end{bmatrix} \qquad (27)$$

VI. Implementation of the Faults Detection and Isolation Programs

The design methods of the residuals generator developed in section 5 on the Quanser plant have been implemented on a control system of the real plant. For this there were used the facilities offered by Quanser acquisition and control system with acquisition card and the external connection mode transducers and actuators of the plant.

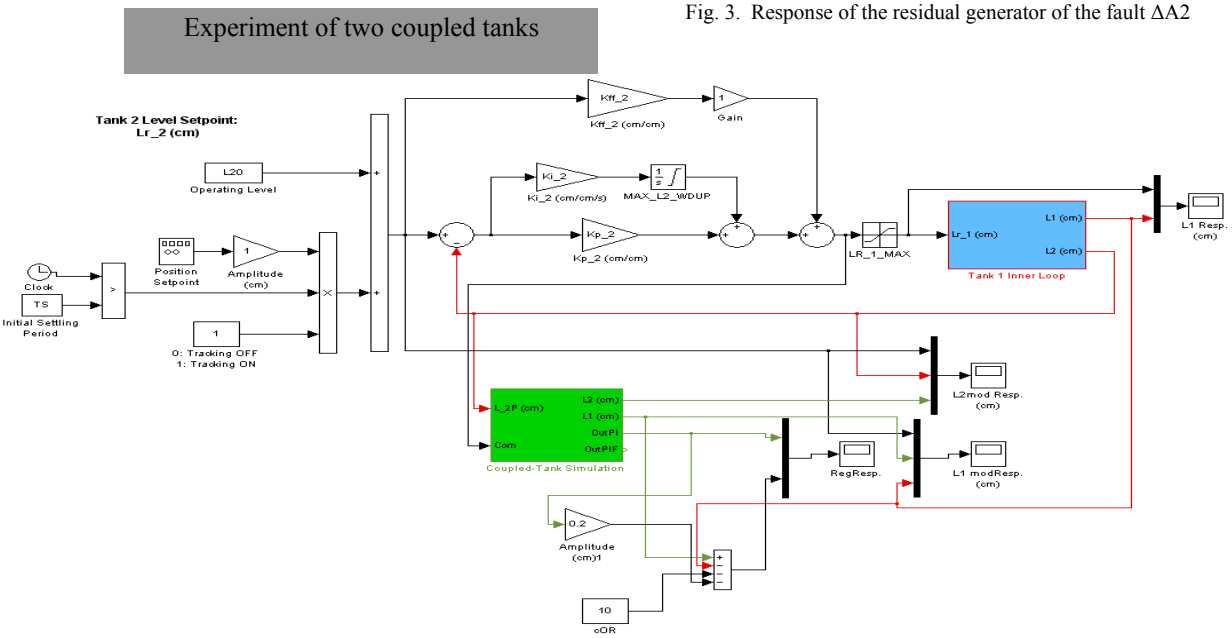


Fig. 2. Diagram of the residual generator

Depending the defect sought to be defective one can make simple configurations in fig. 2 in order to define the residues $r_1(t)$, $r_2(t)$, $R_1(t)$, $R_2(t)$ with the proper implementation of the weighting matrices as specified in section 5.

Quanser plant facility is controlled by a cascade control structure as the one showed in fig.2, aiming at adjusting the control level L_2 with a size prescribed 15 cm above the regular rectangular signal overlaps with a period of 200 sec.

Further on there are presented the experimental results only on the actual installation of the faults ΔA_2 and Δu_2 (figure 3).

In assessing the advantages of implementing the residuals generator, variations of the level L_2 in the plant (red) are shown in Fig. 4, which aims the prescribed size and the variations of L_1 in the plant (blue) when modifying the prescribed measure and the fault. One can note that it is not possible to control the differences between the variations made on command and the variations made by fault over the two variables (compared analysis of the two graphics 3 and 4).

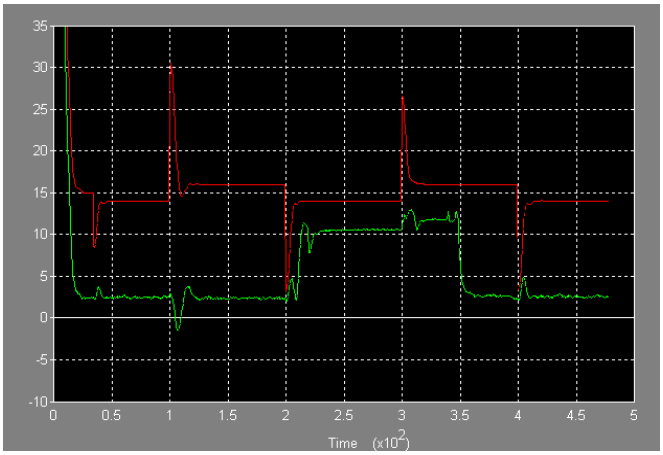


Fig. 3. Response of the residual generator of the fault ΔA2

Variations are shown in Fig. 3 of the level L_{1r} (green) and L_{1m} (red), produced both by the prescribed measure and also by defects. It is also noted that one cannot distinguish between changes caused by the faults and variations caused by the prescribed size.

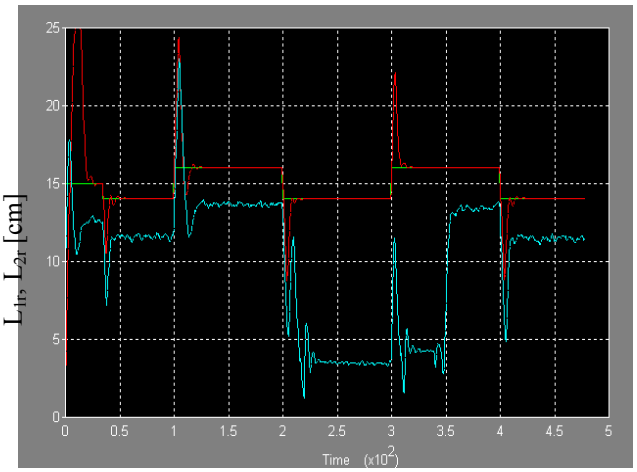


Fig. 4. Variation of the level L_{2r}

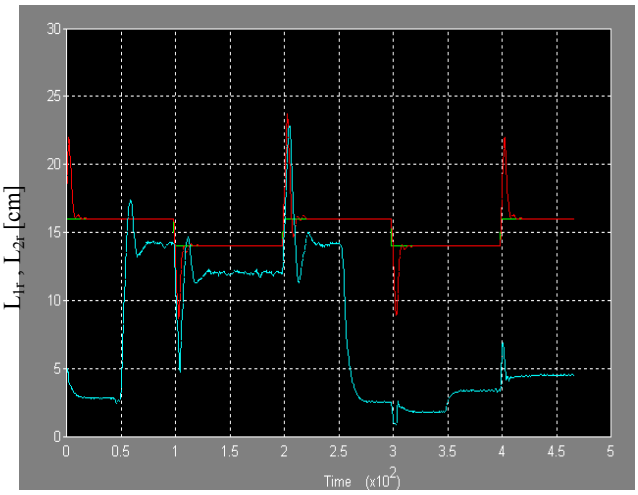


Fig. 5. Variation of the L_{1r} and L_{2r}

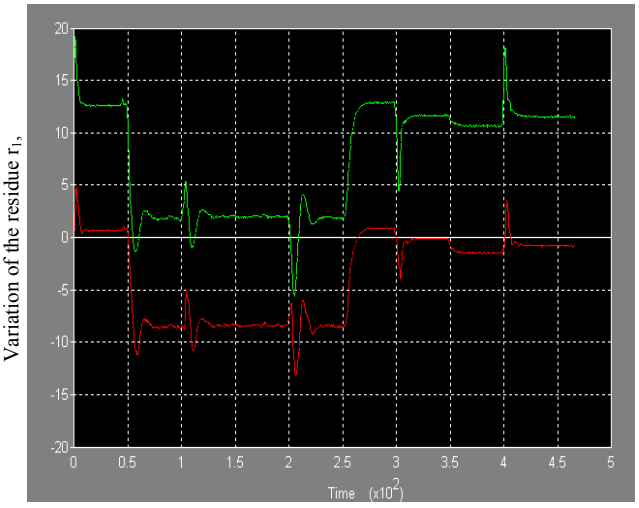


Fig. 6. Fault of the pump Δu_2

The response of the residual generator is shown in Figure 6, green is the residue R1 provided by the adaptive generator and the red is to residue R1. At time t = 50 sec a fault corresponding to the pump power flow of tank 2 was provoked, fault that disappeared at time t = 250 seconds, remaining only tank 1 input by the command measure. It is clearly seen in this case that the residue R1 is close to 0 in the absence of defect and values around -8 in the presence of the defect. There are observed in this case small variations caused by noise measurement transducers or the commands given by the control pump, which can be eliminated by introducing fault detection thresholds to eliminate false alarms. To compare and evaluate the advantages of using the adaptive residue generator in Figure 5 there are presented - in red - changes in the level L2, which seeks variations in the prescribed measures and L1r represented by blue that changes both when dealing with the prescribed measures and the defect variation.

VII. Conclusion

In this paper a method for faults detection and localization using residual vectors was presented; the proposed method was theoretically developed and experimentally verified as laboratory plant for control level using two coupled tanks "Quanser; it allowed detection and localization of two faults created in a real plant. To define residues, a mathematical model of the laboratory facility highlighting the size of order, disturbance and possible faults which may occur in the plant was developed, as can be seen in section 4.

A series of case studies was realized regarding the possibilities for detection of the faults using residuals generator for the system in discrete form.

References

[1] Asokan, D. Sivakumar, "Model based fault detection and diagnosis using structured residual approach in a multi-input multi-output system", *Serbian Journal of Electrical Engineering*, Vol. 4, No. 2, pp. 133-145, 2007.

[2] J. Gertler, "Fault Detection and Diagnosis in Engineering Systems", *Ed. Marcel Dekker*, 1998.

[3] Hong-Yue Zhang, "Fault Detection, Supervision and Safety of Technical Processes", *A Proceedings Volume from the 6th IFAC Symposium on Fault Detection*, Ed. Elsevier Science, 2006.

[4] K Käoppen-Seliger, E. Alcorta-Garca, P.M. Frank, "Fault Detection – Different Strategies for Modeling Applied to the Three Tank Benchmark – A Case Study", European Control Conference, Karlsruhe, Germany, 1999.

[5] M. Tertişco, B. Jora, D. Poescu, I. Russ, "Automatizări industriale continue", Ed. Didactică şi Pedagogică, 1991.

[6] V. Ramakrishnan, Y. Zhuang, S.Y. Hu., J.P. Chen, C.C Ko, B.M. Chen, K.C. Tan, "Development of a web-based control experiment for a coupled tank apparatus", American Control Conference, Proceedings of the 2000, Volume 6, Issue, pp. 4409 – 4413.

[7] J. Gertler, M. Staroswiecki, M. Shen, "Direct Design of Structured Residuals for Fault Diagnosis in Linear Systems", American Control Conference, Anchorage, Alaska, May 2002.

[8] J. Gertler, "Residual Generation in Model Based Fault Diagnosis", Control-Theory and Advanced Technology, Vol. 9, pp. 259-285, March 1993.

Towards Quantum Computer Graphics

Simona Caraiman

***Abstract*— In this paper we outline the recent development of a new paradigm in the field of quantum computing: quantum computer graphics. Our purpose is to demonstrate how fundamental computer graphics problems can be expressed using the quantum formalism and how corresponding quantum computer graphics algorithms can be formulated in order to exploit the immense potential of quantum information processing given by its remarkable properties: inherent parallelism of quantum superpositions, quantum interference and entanglement of quantum states.**

I. Introduction

The evolution of current technology allows us to predict that in the following twenty years, the basic element of binary information - the bit - could be implemented at subatomic scale. For example, the $-1/2$ particle spin could be used, or the vertical or horizontal particle polarization, or even the energy level of a hydrogen electron: all these offer various possibilities to encode 0 and 1 basis states. At this level, the laws of classical physics cease to apply, while the laws of quantum physics take place, even though the latter ones are not very intuitive. In order to achieve 10^{16} gate densities on a chip and frequencies perhaps higher than 10^{15} Hz - performances foreseen for 2020 in the case of a "normal" evolution - we have to adopt theoretical models fundamentally and explicitly based on the laws of quantum physics. Instead of eliminating the quantum effects, we will have to exploit them as they can offer by nature a still unimagined computing power.

In this paper we present the quantum algorithms that allow sketching the development of the quantum computer graphics field. We show how fundamental problems in rendering can be solved more efficiently using quantum computation. We also consider other important results with direct applications to these fields like quantum computational geometry and quantum model fitting.

II. Basic Concepts in Quantum Computing

The research in quantum informatics appeared as a consequence of Richard Feynman's ideas, winner of a Nobel Prize in physics, who in 1982 suggested that quantum phenomena can be used in conceiving quantum computers. The application of the principles of quantum physics in the computer area led to the concept of quantum computer, in which the data isn't stored in bits like in the conventional memory, but as a combined state of several systems with 2 qubit states.

S. Caraiman is with Faculty of Automatic Control and Computer Engineering, Technical University of Iasi, Romania sarustei@cs.tuiasi.ro

Just like a classical computer, a quantum computer is built out of three main parts: processor, memory and input/output. A quantum computer can be formally described by $M = (\mathcal{H}, O, T, \delta, \beta)$, where $\mathcal{H}$ - represents the states space (Hilbert space C^{2^n})of the quantum system, O - the set of unitary transformations, T - the set of measurement commands, δ - is an initialization operator and β - describes the initial measurement.

The quantum analogous of the classical bit is the qubit. A qubit is a quantum system whose states can be completely described by the superposition of two orthonormal basis states, labeled $|0\rangle$ and $|1\rangle$ (in a Hilbert space $\mathcal{H} = C^2$, $|0\rangle = (1 \quad 0)^T$, $|1\rangle = (0 \quad 1)^T$). Any state $|\Psi\rangle$ can be described by:

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle, \quad |\alpha|^2 + |\beta|^2 = 1, \tag{1}$$

where α and β are complex numbers. Thus, unlike the classical bit, the qubit can also be in a state different from $|0\rangle$ and $|1\rangle$: linear combinations of states can be formed, called superpositions (1). When measuring a qubit either the result 0 is obtained, with probability $|\alpha|^2$, or 1 with probability $|\beta|^2$. The sum of the probabilities must be 1, so the state of a qubit represents a unit vector in a complex bi-dimensional vector space.

A collection of n qubits is called a quantum register with dimension n. The general state of a n-qubit register is

$$|\Psi\rangle = \sum_{i=0}^{2^n-1} a_i|i\rangle, \tag{2}$$

where $a_i \in \mathcal{C}$, $\sum_{i=0}^{2^n-1} |a_i|^2 = 1$. This means that the state of a n-qubit register is represented by a complex unit vector in Hilbert space $\mathcal{H}_{2^n}$.

A. Quantum Gates

The quantum analogous of the classical NOT gate is labeled X and can be defined such that $X|0\rangle = |1\rangle$ and $X|1\rangle = |0\rangle$. The quantum NOT gate act similarly with its classical counterpart, although, unlike in the classical case, its action is linear: state $\alpha|0\rangle + \beta|1\rangle$ is transformed in a corresponding state $\beta|0\rangle + \alpha|1\rangle$. A convenient way of representing the action of the quantum NOT gate is in matrix form:

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}. \tag{3}$$

Hadamard gate, given by the following matrix form

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \tag{4}$$

is sometimes described as "square root of NOT" because when applying it to any of the $|0\rangle$ or $|1\rangle$ basis states produces an equal mixture of both of them:

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \quad and \quad H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle). \quad (5)$$

Controlled gates are quantum logical gates acting on more than one qubit. The notion of controlled gate allows the implementation of the $if - else$ constructs. Quantum controlled gates use a control qubit to determine whether a specific unitary action is applied to a target qubit.

The controlled-NOT operator (CNOT) is the prototypical multi-qubit gate. The first parameter of a CNOT gate is the control qubit. If this qubit is in state $|0\rangle$, the target qubit is left unchanged and if the control qubit is in state $|1\rangle$, the target qubit is flipped:

$$|00\rangle \to |00\rangle; \quad |01\rangle \to |01\rangle; \quad |10\rangle \to |11\rangle; \quad |11\rangle \to |10\rangle.$$

The CNOT operator is a generalization of the classical XOR, since its action can be summarized as $|x, y\rangle \to |x, x \oplus y\rangle$, where $\oplus$ is addition modulo two. The matrix representation of CNOT is:

$$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}. \quad (6)$$

There are several other multi-qubit gates, Nevertheless, the controlled-NOT gate and the single qubit gates represent the prototypes for any other quantum gate because of the following remarkable universality result: any multi-qubit gate can be built out of CNOT gates and single qubit gates. The proof of this statement represents the quantum analogous to the universality of the classical NAND gate.

B. Quantum Algorithms

According to the principles of quantum physics the computing power of a quantum machine is immense compared to the one of a classic computer due to three quantum resources that have no classical counterpart. *Quantum parallelism* harnesses the superposition principle and the linearity of quantum mechanics in order to compute a function simultaneously on arbitrarily many inputs. *Quantum interference* makes it possible for the logical paths of a computation to interfere in a constructive or destructive manner. As a result of interference, computational paths leading to desired results can reinforce one another, whereas other computational paths that would yield an undesired result cancel each other out. Finally, there exist multi-particle quantum states that cannot be described by an independent state for each particle. The correlations offered by these states cannot be reproduced classically and constitute an essential resource of quantum information processing called *entanglement*.

These remarkable properties of quantum systems allowed the formulation of optimal algorithms for two fundamental problems: integer factorization (Shor's algorithm [1]) and the search in an unstructured database (Grover's algorithm [2]). Thus, two main classes of quantum algorithms have better time complexity than their classical counterparts. The first one is based on the quantum Fourier transform [3] and includes remarkable algorithms for solving the factorization and discrete logarithm problems, while providing exponential speedup over the best known classical algorithms. The second class of algorithms is based on Grover's quantum search algorithm [2]. This class of algorithms provides a quadratic speedup with respect to the best classical algorithms. The importance of the quantum search algorithm is underlined by the intensive use of search techniques in classical algorithms, while, in many cases, a natural adaptation is possible in order to provide a faster quantum algorithm [4]. Special interest is given to the quantum counting algorithm which represents an inspired combination of quantum search with the Fourier transform and allows the estimation of the number of solutions to a search problem faster than possible on a classical computer [5].

The main source of computational power of quantum systems is represented by the inherent parallelism of quantum computation. Quantum parallelism is a fundamental feature for many quantum algorithms. In an simplified manner, this parallelism allows quantum computers to evaluate a function, $f(x)$, for many values of x simultaneously, by creating superposition states. Nevertheless, this parallelism isn't immediately useful because measuring a superposition state (containing all values $f(x)$) has the effect of collapsing the superposition in one of the basis states, thus only giving the value of the function for one single value for x. Quantum computing needs more than parallelism in order to be useful: the ability to extract information about more than one value of $f(x)$ from superposition states. This can be realized by combining quantum parallelism with the interference of quantum states and it is exploited by the known quantum algorithms that perform better than the classical counterparts.

These results determined not only the investigation of innovative applications that can be developed using these high performance computing systems, but also of ways to improve the performances over the classical case. Exploiting the properties of quantum systems and these fundamental results recently led to the emergence of innovative ideas in the field of computer graphics.

III. Quantum Computer Graphics Algorithms

A. Introduction

Many problems in classical computing can be reformulated to express the search of a unique element that satisfies a certain predefined condition. If there is no additional information about the search condition, the best classical algorithm is a brute-force search, meaning that the elements are sequentially tested against the search condition. For a list of N elements, this algorithm executes an average of $N/2$ comparisons. By exploiting the advantages of quantum parallelism and interference of quantum states, Grover formulated a quantum algorithm that can find the searched element in an unstructured database in only $O(\sqrt{N})$ steps [2].

Other recent results allow quantum algorithms to be used in solving search problems involving multiple solutions. These results include the semicloning protocol and the algorithm for the extraction of a semiclone state [6] and enlarge the spectrum of problems that can be efficiently solved using quantum parallelism.

Grover's results are only valid for the case in which the number of elements satisfying the search condition is $t = 1$. Nevertheless, in most practical applications t is unknown and the extraction of the entire set of t solutions is required. There are variations of Grover's algorithm that can randomly extract one of the solutions when t is unknown by applying $\lfloor \frac{\pi}{4}\sqrt{\frac{N}{t}} \rfloor$ iterations for $t < \frac{N}{2}$ [7]. Extracting the entire set of solutions requires the determination of number t prior to executing the $O(\sqrt{\frac{N}{t}})$ iterations of Grover's algorithm. The optimal quantum algorithm which determines the number t of solutions requires $O(\sqrt{(t+1)(N-t+1)})$ time (the approximate counting algorithm [5]) and dominates the complexity of Grover's algorithm for the case when t is unknown. Nevertheless, knowing t doesn't allow for the sequential extraction of the solutions because each time a state is measured, the solution superposition is destroyed. Thus the repeated application of Grover's algorithm would be needed in order to randomly extract the solutions and the number of searches applied for determining all t solutions is $O(t \log t)$. This would imply a total complexity of $O(\sqrt{tN} \log t)$. In [6] authors present an algorithm, the semicloning protocol, that eliminates the necessity to randomly extract the solutions, thus obtaining an optimal complexity, $O(t)$.

Another result with significant importance to quantum search is the quantum algorithm for finding the minimum/maximum value in an unstructured data set. This can be done in $O(\sqrt{N})$ steps as described in [8].

In the following we will show how these algorithms can be exploited in order to devise quantum solutions for several fundamental computer graphics problems.

B. Quantum Rendering

In computer graphics each object that needs to be visualized is decomposed in thousands or millions of polygons or other such surfaces. These surfaces are stored in a database which for most practical applications is very large, for example several million elements, depending on the scene complexity. In order to render such a scene, the visualization system applies a variety of rendering algorithms to each element of the database. These rendering algorithms have a common characteristic of executing several searches in the scene database in order to find which ones are visible. Taking into account the large number of objects in the database, this search process tends to represent the most significant source for latency. Thus, these search operations are primary candidates for optimization using Grover's quantum search algorithm or any of its variants. Using such an approach we show how quantum algorithms ca be devised for determining polygon visibility using the Z-Buffering algorithm, for global illumination models such as Ray-Tracing, radiosity and Photon Mapping and for the representation of models using levels of detail.

1) Polygon Visibility Problem: In order to render a 3D scene it is necessary to determine the visibility of the geometric primitives that form the objects of the scene. One of the simplest methods that can be used is the Z-Buffering algorithm [9]. Even though the Z-Buffering algorithm is not necessarily a search problem, a quantum approach for this algorithm can be built by exploiting a variant of Grover's algorithm: the quantum algorithm for searching the minimum value in a data set. The main idea behind this approach was suggested in [10] and we synthesize it in Algorithm 1:

For each pixel in the resulting image the quantum Z-Buffering algorithm uses a superposition of quantum states where each element represents one of the N polygons of the scene. In order to determine the polygons that intersect the current pixel we can apply the quantum search algorithm with multiple solutions. As discussed in the beginning of this section, this stage is executed in $O(\sqrt{dN})$, where d represents the number of polygons satisfying the search condition.

Next, the distance from the current pixel to each such polygon is computed in $O(1)$ time, exploiting the natural parallelism of the quantum computation. Out of these distances we have to find the shortest one and use the corresponding polygon to color the current pixel. This can be done in $O(\sqrt{d})$ time using the quantum min search algorithm.

Input : data base containing N polygons
Output: FB - color buffer of the resulting image

1 **foreach** *pixel* (x, y) *in the image* **do**
2 create a uniform superposition of states, $|\Psi_0\rangle$, where each element represents one of the N polygons in the scene, $\{P_1, P_2, ..., P_N\}$, $|\Psi_0\rangle = \frac{1}{\sqrt{N}}(|P_1\rangle + |P_2\rangle + ... + |P_N\rangle)$;
3 apply the approximate counting algorithm [5], where a solution state corresponds to polygons intersecting pixel (x, y);
4 apply algorithm [6] to extract the d solutions of the search problem in step 3 ;
5 create a uniform superposition of states, $|\Psi_1\rangle$, where each element represents one of the d polygons that intersect the current pixel, $\{P_1, P_2, ..., P_d\}$, $|\Psi_1\rangle = \frac{1}{\sqrt{d}}(|P_1\rangle + |P_2\rangle + ... + |P_d\rangle)$;
6 $|\Psi_2\rangle = determineZ(|\Psi_1\rangle)$;
7 $P_m = detMin(|\Psi_2\rangle)$ `// determine the minimum value using algorithm [8]`
8 $FB(x, y) = I_{P_m}(x, y)$;

Algorithm 1: Quantum algorithm for determining the visibility of geometric primitives in a 3D scene

The complexity of the quantum Z-Buffering algorithm is $O(P\sqrt{dN})$, while the classical algorithm needs $O(P(d + N))$ steps. Here P represents the total number of pixels, d

is the medium number of polygons intersecting one pixel and N is the number of polygons in the scene. The main advantages of the quantum variant of the algorithm are the fact that it has a much better scalability with respect to the number N of polygons than its classical counterpart and that it copes with arbitrarily shaped objects and it is not limited to simple polygons.

2) Global Illumination: In Computer Graphics, the rendering equation describes the flow of light energy in a 3D scene. Based on the physics of light, it provides theoretically perfect results, in contrast to the various rendering techniques, which approximate this ideal.

The physical basis for the rendering equation is the law of conservation of energy. At a particular position and direction, the outgoing light (L_o) is the sum of the emitted light (L_e) and the reflected light (L_r). The reflected light itself is the sum of the incoming light (L_i) from all directions, multiplied by the surface reflection and incoming angle [11]:

$$\begin{aligned} L_0(x,w) &= L_e(x,w) + \\ &+ \int_\Omega f_r(x,w',w) L_i(x,w')(w'\cdot n) dw', \end{aligned} \quad (7)$$

where:

$L_0(x,w)$ is light outward at a particular position x and in direction w,

$L_e(x,w)$ s light emitted from the same position and direction,

$\int_\Omega \cdots dw'$ is an infinitesimal sum over a hemisphere of inward directions,

$f_r(x,w',w)$ is the proportion of light reflected at the position x (from inward direction to outward direction),

$L_i(x,w')$ is light inward from the position x and direction w',

$(w'\cdot n)$ the attenuation of inward light due to incident angle.

Solving the rendering equation for any given scene represents the main challenge in realistic rendering. One approach for solving this equation is based on finite element methods, leading to the radiosity algorithm. Another approach using Monte Carlo methods has led to many different algorithms including ray tracing or photon mapping.

Ray Tracing is one of the most used rendering techniques [12]. One such algorithm determines the visibility of the surfaces by tracing a ray from the observer to the objects in the scene. The intersection between a ray and on object determines the color of the pixel. This process takes place for each pixel on the screen. Thus, for each pixel on the screen, the algorithm determines whether there is an intersection between the ray and any of the objects in the scene. If the ray intersects more than one object, only the closest one from the screen is displayed. Thus, the classical algorithm executes $O(N)$ operations per ray, because it determines the intersections for each of the N objects in the scene.

Because Ray Tracing is by nature a search algorithm, it is a good candidate for optimization using Grover's quantum search algorithm. In this way the ray tracing can be implemented by executing a quantum search for the intersections between rays and polygons, followed by a quantum search for the polygon closest to the screen.

For a quantum implementation of the Ray Tracing algorithm, a quantum state $|\Psi\rangle$ is created encoding all polygons in the scene in a uniform superposition:

$$|\Psi\rangle = \alpha(|00...01\rangle + |00...10\rangle + ...). \quad (8)$$

This superposition of states is used for each traced ray. It is also necessary to define a function f that would act on each states of the superposition like in the following:

$$f(x) = \begin{cases} 1, & x \; intersects \; the \; ray \\ 0, & otherwise \end{cases}. \quad (9)$$

Quantum parallelism can be employed in order to evaluate $f(x)$ for each element of the superposition (each polygon in the scene) in a single computational step. Next, the semi-cloning protocol can be used to determine all k intersecting the ray in $O(\sqrt{Nk})$ time. The polygon closest to the screen can now be determined using the quantum algorithm for determining the minimum in $O(\sqrt{k})$ steps. The total complexity of the quantum ray tracer becomes $O(\sqrt{Nk} + \sqrt{k})$ per ray. In most practical application $k \approx O(1)$ and $k \ll N$.

The main advantage of this quantum solution over the classical case is the optimal time complexity for general objects queries, with linear space complexity. In the classical case, polygons are used because computing the intersection is easier than in the case of generic objects.

Ray Tracing is an excellent method for simulating reflected and refracted light. Nevertheless, it approximates the diffuse light in a rudimentary and expensive way. The radiosity algorithm is a method that models very well diffuse light, but offers a poor representation of reflected light [13].

Radiosity computation is based on the principle of energy conservation [14], where the total energy emitted by a diffuse material is equal to the energy emitted naturally by the material plus the reflected energy:

$$Radiosity = emitted \; energy \; + \; reflected \; energy, \quad (10)$$

where, *Reflected energy = reflection index * total incident energy* coming from all other objects in the scene.

This can be written as the radiosity equation [13]:

$$B_i dA_i = E_i dA_i + \zeta_i \sum_j B_j F_{ji} dA_j \quad (11)$$

In these equations B_i represents the radiosity of each element, dA_i is the surface element, E_i represents the emitted energy, ζ_i is the reflection index of the material and F is the form factors matrix which determines the incident energy emitted by all other objects in the scene. This radiosity equation can be re-written in matrix form:

$$M \cdot B = E, \quad (12)$$

where M is an interaction matrix.

Thus, the shading of each pixel implies the following three steps:

1) Computing the matrix of form factors - this can be done using the Ray Tracing algorithm in $O(N^2)$ steps.
2) Determining the objects in the scene that contribute to the radiosity of the current pixel - this can be done using a ray tracer to determine the visibility of each object in the scene from the position of the current pixel ($O(N)$ steps).
3) Solving the radiosity equation - this can be done by applying Gauss method to solve the system and compute the radiosity vector B in $O(N)$ steps.

Stages (1) and (2) are operations similar to the Ray Tracing algorithm for which we have already seen the quantum variant. Thus, using a quantum ray tracer we can execute stage (1) in $O(N^{3/2})$ steps and step (1) in $O(\sqrt{N})$ steps. Still, step (3) cannot be optimized using a quantum computer. As a consequence, quantum radiosity can be executed in $O(N^{3/2})$ time, in contrast with $O(N^2)$ time needed in the classical case.

Photon mapping is a two-step global illumination algorithm developed by Henrik Jensen [15] as an efficient alternative to pure ray tracing Monte Carlo techniques.

Ray Tracing systems cannot generate correctly caustic phenomenon, cannot render indirect illumination produced by objects that reflect light and cannot implement diffuse inter-reflection (thus nor color bleeding). On the other hand, radiosity methods easily produce diffuse inter-reflections and indirect light, but cannot cope with specular reflection, have difficulties in processing transparency, need scene subdivisioning and are time consuming (a second pass is needed to produce reflection and refraction).

The idea behind Photon Mapping is to decouple the representation of the scene from its geometry and to store the illumination information in a global spatial data structure called *photon map*. The first step of the method builds the photon map by tracing photons leaving from a light source, while in the second step the scene is rendered using the information stored in the photon map.

A quantum implementation of Photon Mapping can benefit from the special properties of quantum computation under two aspects. On one hand, searching the intersections between a photon and a collection of N objects in the scene and determining the first intersection can be accelerated using quantum search algorithms just like in the case of the Ray Tracing algorithm. Thus, the first step of the quantum Photon Mapping has $O(\sqrt{N})$ time complexity, unlike $O(N)$ in the classical case.

On the other hand, unlike classical computers, quantum systems can implement Monte Carlo methods using authentic random values. In classical computing, pseudo-random values are used (in fact, fully deterministic numbers). The Monte Carlo techniques are used in Photon Mapping for two purposes [15]: photon emission by a light source and to probabilistically decide whether at the intersection with a surface the photon is absorbed, reflected or refracted, depending on the surface material (technique called the Russian Roulette).

Using a quantum computer as a random number generator is very simple and assumes applying the Hadamard transform to a quantum state that encodes the probability distribution. The effect of the Hadamard transform is to create uniform superposition of states and thus each state has the same probability to be measured:

$$H(|0\rangle|0\rangle\ldots|0\rangle) = \frac{1}{2^{m/2}}\sum_{i=0}^{2^m-1}|i\rangle \xrightarrow{measure} |i\rangle$$

$$with \quad probability \quad 2^{-m}$$

3) Scene Management: When rendering complex three-dimensional scenes, it is commonly the case that many objects are very small or distant. The size of many geometric features of these objects falls below the perception threshold or is smaller than a pixel on the screen. To better use the effort put into rendering such features, an object should be represented at multiple levels of detail (LODs). Simpler representation of an object can be used to improve the frame rates and memory utilization during interactive rendering. This technique was first described by Clark already in 1976 [16], and has been an active area of research ever since.

Level-of-detail algorithms can be essential for building very large worlds. They can be used to control the total amount of geometry in a scene. They can also help meet fixed performance goals under varying conditions (such as running on different machines). Generating levels of details addresses the problem of finding a series of progressive simplifications of a polygonal object, that have fewer primitives (polygons), but closely resemble the original object. The LOD method determines which vertices are necessary for representing the object in the given conditions. If certain vertices are not needed, they are removed from the polygonal mesh. To determine whether a vertex is needed or not, the method evaluates an error function $\epsilon(\nu)$ for each vertex ν. If the error associated with a vertex is smaller than a threshold, δ, then the vertex can be eliminated. In order to increase performance, all polygons can be encoded in a hierarchical tree structure that determines the active vertices at a certain moment. In this case, if the error is smaller than the threshold, the vertex can be further expanded to include inferior portions of the tree. Nevertheless, the process needs to be executed for the whole set of N active vertices and requires $O(N)$ steps.

In a quantum LOD method, a state $|\Psi\rangle$ can be used to encode the polygonal mesh with N vertices. Hadamard gates can be used to achieve an uniform superposition of states where each element represents a vertex. A function f can be defined such that

$$f(\nu) = \begin{cases} 1, & if\ \epsilon(\nu) < \delta \\ 0, & otherwise \end{cases} \tag{13}$$

Quantum parallelism can be exploited to evaluate function $f(\nu)$ for each element of the superposition in a single computational step. Quantum search can then be employed to find an unknown number of vertices with a sufficiently small error such that they can be discarded. This operation

can be executed in $O(\sqrt{N})$ steps, in contrast with the $O(N)$ steps needed in the classical case.

C. Other Algorithms for Quantum CG

There are many other algorithms commonly used in Computer Graphics. Several of them have already been defined in quantum terms. We mention here some Computational Geometry algorithms and the Random Sample Consensus algorithm.

Computational Geometry is concerned with the computational complexity of geometric problems that arise in a variety of disciplines: Computer Graphics, Computer Vision, Virtual Reality, Multi-Object Simulation and Visualization, Multi-Target Tracking. Many of the most fundamental problems in computational geometry involve multidimensional searches (search for those objects in space that satisfy a certain query criteria) and representation of spatial information (determination of the convex hull of a set of points, determination of object-object intersections). For example, virtually all of the computational bottlenecks in graphics and simulation applications involve some variant of spatial search: collision detection involves the identification of intersections among spatial objects; Ray Tracing involves the identification of the spatial object that is intersected first along a line of sight. As suggested in [17] Grover's algorithm can be used in order to accelerate some of the most important computational geometry algorithms: multidimensional search, determining the convex hull of a set of points, determining object-to-object intersections.

Another result with significant importance to the subject of the paper is the quantum RANdom Sample Consensus algorithm. RANSAC is an algorithm for robust fitting of models in the presence of many data outliers and has proved to be a very important result for both computer graphics and vision. The RANSAC algorithm has been applied to a wide range of problems: fundamental matrix estimation, trifocal tensor estimation, camera pose estimation, structure from motion and shape detection. The performance of the quantum variant of this algorithm is orders of magnitude faster than its classical variant due to the reformulation of the steps of the algorithm in the terms of search problems [4].

IV. Discussion

Theoretical and experimental research accomplished by now in the field of quantum computing are in an initial stage and still a lot of effort is needed for clarifying the theoretical and practical aspects concerning the information coding and processing based on quantum phenomena. Nevertheless, it has become an important area of research in computer science and the increased interest is justified by the existence of efficient quantum algorithms which can perform some calculations significantly faster than classical computers. Building a working quantum computer represents a very challenging task that requires spending a huge amount of resources. Developing efficient quantum algorithms for practical problems would therefore help in justifying these immense efforts. In this paper we have shown how existing quantum algorithms (such as Grover's algorithm and its variants) can be used to improve the performance of several fundamental computer graphics algorithms. The speedups of these quantum variants can also be used to increase the realism of the rendered images. Defining the quantum variants for these algorithms was possible by reformulating them in order to exploit the special properties of the superpositions of quantum states. In this way, we identified the steps of these algorithms that could be stated in the terms of a search problem and provided the variants of Grover's algorithm suited for solving these specific steps. The performances of the resulting quantum algorithms are orders of magnitude faster than the classical variants and represent a promising result for the investigation of more such applications of quantum computing to computer graphics an vision tasks.

References

[1] P. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *SFCS '94: Proceedings of the 35th Annual Symposium on Foundations of Computer Science*. IEEE Computer Society, 1994, pp. 124–134.

[2] L. Grover, "A fast quantum mechanical algorithm for database search," in *Proceedings of 28th ACM Annual STOC*, 1996, pp. 212–219.

[3] M. Nielsen and I. Chuang, *Quantum Computation and Quantum Information*, ser. Cambridge Series on Information and the Natural Sciences. Cambridge, UK: Cambridge University Press, 2000.

[4] S. Caraiman and V. Manta, "New applications of quantum algorithms to computer graphics: the quantum random sample consensus algorithm," in *CF '09: Proceedings of the 6th ACM conference on Computing frontiers*. ACM, 2009, pp. 81–88.

[5] G. Brassard, P. Hoyer, M. Mosca, and A. Tapp, "Quantum amplitude amplification and estimation," 2000, quant-ph/0005055.

[6] M. Lanzagorta and J.Uhlmann, "Hybrid quantum computing: semi-cloning for general database retrieval," in *Proceedings of the Quantum Information and Quantum Computation Conference, SPIE Defense and Security Symposium*, vol. 5815, 2005, pp. 78–86.

[7] M. Boyer, G. Brassard, P. Hoyer, and A. Tapp, "Tight bounds on quantum searching," in *Proceedings of Fourth Workshop on Physics and Computation*, 1996.

[8] C. Durr and P. Hoyer, "A quantum algorithm for finding the minimum," 1999, quant-ph/9607014.

[9] E. Catmull, "Computer display of curved surfaces," in *Proceedings of the IEEE Conference on Computer Graphics, Pattern Recognition and Data Structures*, 1975, pp. 11–17.

[10] M. Lanzagorta and J. Uhlmann, "Hybrid quantum-classical computing with applications to computer graphics," in *SIGGRAPH '05: ACM SIGGRAPH 2005 Courses*. New York, NY, USA: ACM, 2005, p. 2.

[11] P. Dutre, K. Bala, P. Bekaert, and P. Shirley, *Advanced Global Illumination*. AK Peters Ltd, 2006.

[12] A. Glassner, Ed., *An introduction to ray tracing*. London, UK: Academic Press Ltd., 1989.

[13] M. Cohen and J. Wallace, *Radiosity and Realistic Image Synthesis*. San Diego, CA, USA: Academic Press Professional, 1993.

[14] J. D. Foley, A. V. Dam, S. Feiner, and J. Hughes, *Computer Graphics: Principles and Practice*. Addison Wesley, 1996.

[15] H. Jensen, *Realistic image synthesis using photon mapping*. Natick, MA, USA: A. K. Peters, Ltd., 2001.

[16] J. Clark, "Hierarchical geometric models for visible surface algorithms," *Communications of the ACM*, vol. 19, no. 10, pp. 547–554, 1976.

[17] M. Lanzagorta and J. Uhlmann, "Quantum computational geometry," in *Proceedings of SPIE 2004: Quantum Information and Computation II*, E. Donkor, A. Pirich, and H. Brandt, Eds., 2004, pp. 332–339.

Modelling and Control of an Autonomous Energetic System Obtained Through Trigeneration

Sergiu Caraman, Marian Barbu, Viorel Minzu, Nicolae Badea, Emil Ceangă

Abstract— **The autonomous energetic system is for a residential building and it has as primary sources exclusively renewable resources (biomass and solar energy). For obtaining electrical energy, photovoltaic sources and a Stirling engine are used. They operate simultaneously in 1:3 ratio – electrical / thermal energy. Solar collectors and a pellet boiler are used to cover the necessary of thermal energy. The air conditioning in the summer regime is obtained using an adsorption plant. The energetic deficit is critic in the electrical subsystem where the load control is done through the adjusting of the Stirling engine power. The thermal power of this engine is a disturbance variable at the level of the thermal subsystem. The paper deals with the developing of a mathematical model of the whole system. The paper also proposes a control solution for the load regulation in the electrical and thermal subsystems and it presents the results obtained through numerical simulation.**

I. Introduction

Worldwide, the importance of renewable energy sources is that, till 2020, the total consumption of such energy should reach 24% of total consumption. In this context, Romania has proposed to meet the two targets: the first refers to 2010, when 33% of the rough electrical energy consumption be covered from renewable resources and the second refers to 2020, when 24% of total energy consumption be covered from renewable resources, accordingly to the world trend.

From a technical point of view, research into the production of energy from unconventional sources follows several directions: photovoltaic systems, thermo-solar systems, biogas, hydrogen, wind systems etc. These types of energies represent a viable alternative to the situation when oil deposits will be exhausted and, moreover, the energy is produced by clean processes, protecting in this way, the environment.

A very important application of using energy supplying produced from renewable resources is the so-called green house. Generally, CHP (cooling, heating and power) systems are used. They provide the necessary electricity and heat, thus ensuring the energy independence from centralized energy systems. The CHP systems require electricity and heat supplying systems with several sources, which implies an optimal real time operating strategy in terms of the cost of the supplied energy [1]. In the mentioned paper a dynamic simulator (TRNSYS) is presented. Using TRNSYS a micro-CHP system was simulated, proving that it can be a viable and economical option for powering a green house. The same simulations have shown the necessity that this system must operate under optimal conditions to ensure an energetic efficiency to the entire green house system.

In [2] a heating system installed at the Centre for Renewable Energy Sources, located in Central Greece, aiming to ensure the heating requirements for a specific block office of 60 m^2 area is presented. The system was analyzed over a period of six months and it resulted that the contribution of solar energy during this period covers 53% of the heat necessary of the building.

Cogeneration systems (thermal and electrical energy) provide heating and electrical power in a more efficient way than the separate production. In the paper [3] a m-CHP unit based on a Stirling engine fueled with pellets that provide both electricity and heat has been analyzed. The performances and the dynamic behavior of the Stirling engine powered with pellets were analyzed by a test bench. Based on this analysis, a model of the heating system that includes a thermal accumulator was achieved. The study was completed by a numerical simulation analysis of the sensitivities in order to quantify the influence of the key parameters on the annual average performances of the system in a residential building.

This paper aims to analyze an autonomous power system using renewable energy for a residential building located in the campus of "Dunarea de Jos" University of Galati. The system is designed to meet the thermal energy needs (heat in winter, cold in summer and domestic hot water) and electricity needs, with the two components: the operating of the household equipments and the own consumption of the supplying system with thermal and electrical energy.

The paper structure is as follows: in the second section the technological structure of the autonomous energetic system is presented; the third section shows the mathematical model, the fourth section presents the control

Sergiu Caraman, Marian Barbu, Viorel Minzu, Nicolae Badea and Emil Ceangă are with "Dunărea de Jos" University of Galati, Domnească 47, 800008, Galați, România. E-mail: Sergiu.Caraman@ugal.ro.

system; the results obtained through numerical simulation are shown in the fifth section and the last section is dedicated to the conclusions.

II. The Structure of the Autonomous Supplying System

Figure 1 presents the structure of the autonomous supplying system with electrical and thermal energy.

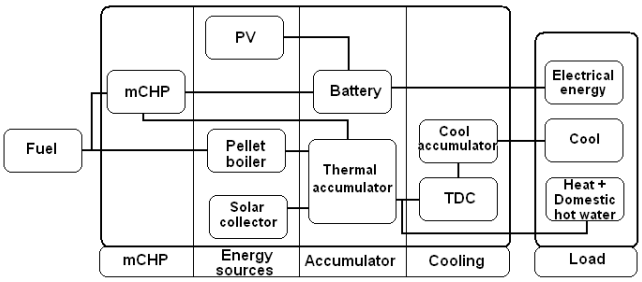


Fig. 1. The structure of the autonomous supplying system

It consists of two subsystems: electrical energy supplying subsystem and the one with thermal energy. For power supply two sources are provided, i.e.: a Stirling engine – m-CHP (it also provides thermal energy in an 1:3 ratio - electrical/thermal) and photovoltaic systems. The electrical energy provided by the Stirling engine and the photovoltaic systems is stored in a battery of 48V/ 800Ah. It aims to provide the necessary electricity in the peak loads.

Considering the thermal energy, outside the Stirling engine, which delivers a maximum power of 10 kW, there are two energy sources more: a pellet boiler that provides maximum 30 kW and a solar collector system that can deliver a maximum 10 kW thermal power. The thermal energy produced by the three sources is stored in a thermal accumulator (th. storage), in which the temperature of the thermal agent varies between 70 – 80 Celsius degrees. The heating system is supplied from this accumulator in the winter with thermal agent, the domestic hot water and the thermal source for the cooling system in the summer are also provided. It can be mentioned that the cooling is done using an adsorption system that needs a thermal energy of 30 kW.

III. The Mathematical Model of the Autonomous Supplying System

Figure 2 presents the overall structure of the whole system.

Electrical subsystem. The main equation of the electrical subsystem is the one that describes the dynamic accumulation in the electro-chemical source. It can be written as follows:

$$\frac{dW_B}{dt} = P_{Se}(t) + P_{PV}(t) - P_{al}(t) - P_{sl}(t) \quad (1)$$

where: W_B is the energy accumulated in battery [J], P_{Se} and P_{PV} are the powers produced by the Stirling engine and PV [W] (photovoltaic source) respectively, P_{al} and P_{sl} are the powers corresponding to the useful load and internal consumption from the energetic system respectively (the pump engines etc) [W]. The battery model links the voltage V to the accumulated energy at the current moment.

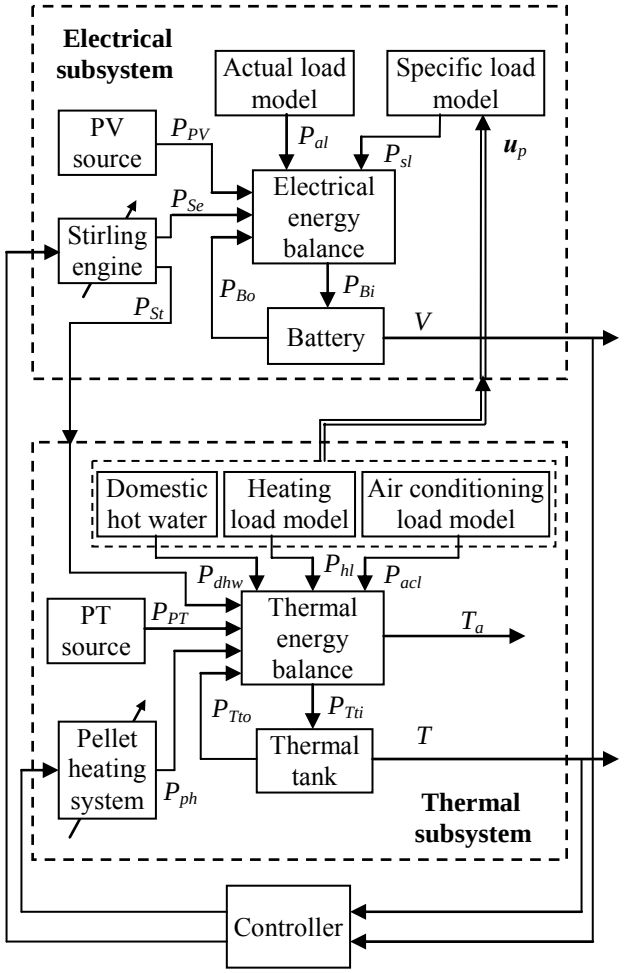


Fig. 2. Structure of the autonomous energetic system

A simplified model for the battery has been adopted:

$$V = V_m + \frac{V_M - V_m}{W_{BM} - W_{Bm}}(V - V_m) \quad (2)$$

where V_M and V_m are the voltage values at maximum energy, W_{BM}, and minimum energy, W_{Bm}, respectively, accumulated in battery [J]. A battery with the nominal voltage of 48 V has been chosen. In equation (2) the following limits of the voltage were considered: V_M=49.5 V and V_m=47 V. Admitting a battery capacity of 800Ah, it results W_{BM}= 138.24 MJ. The minimum value of the accumulated energy, to which the battery is considered

practically discharged is W_{Bm}=17.28MJ.

The variable $V(t)$ is regulated through the control of the electrical power delivered by the Stirling engine. Depending on the reference V^{sp} and the current value V of the voltage, the battery is controlled in charging or discharging regime and the power variables P_{Bi} and P_{Bo}, from these regimes, contribute to the balance of the produced power with the one required by load. The Stirling engine was modelled as a dynamic first order system, with a time constant of 60 sec.

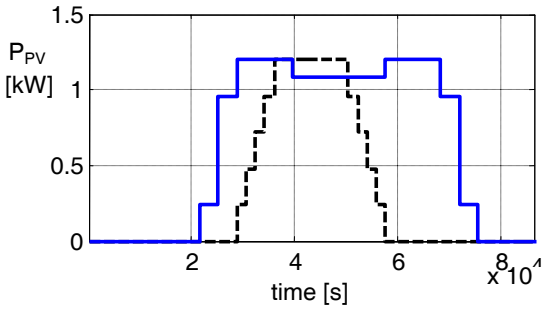


Fig. 3. Daily variation of the power variable $P_{PV}(t)$: in summer regime (solid line) and in winter regime (dash line)

The power produced by PV source has been determined based on the equations:

$$I = \int_{\lambda_1}^{\lambda_2} I_\lambda d\lambda \tag{3}$$

$$R = A\int_0^T I dt \tag{4}$$

where: I is the irradiance, I_λ - spectral irradiance, considered in the wavelength spectrum of interest, [λ_1, λ_2], R - radiation (measure of the optical power), A – surface receiving the light radiation (A = 12 m^2) and [0, T] – the considered time interval (24 hours). Using the usual data from literature, [4], and power conversion efficiency (the ratio between the maximum electrical power and the optical one), $\eta = 0.1$, the daily variations of the solar source power in summer and winter regimes were determined (Figure 3). The daily evolution of the power $P_{al}(t)$ has been adopted as a plausible default graphic of the useful load. Instead, the evolution of the power $Psl(t)$, representing the internal consumption of the plant is random. It is determined by the vector $u_p(t)$, whose components are discrete controls (0/1) given to the 15 motors of the plant (almost all belonging to the thermal subsystem).

Thermal subsystem. The mathematical model of the thermal subsystem has as a core the thermal balances at the level of the thermal accumulator and of the building. The thermal balance equation for the thermal accumulator is the following:

$$m_w c_w \frac{dT}{dt} = P_{St}(t) + P_{PT}(t) + P_{ph}(t) - P_{hl}(t) - P_{acl}(t) - P_{dhw}(t) \tag{5}$$

where m_w, c_w represent the mass and the specific heat of the water accumulated in the thermal accumulator, T – the water temperature in the thermal accumulator, P_{St} – the Stirling engine thermal power, P_{PT} - the solar collector power, P_{ph} – the power provided by the pellet boiler, P_{hl} – the power consumed in the building to cover losses through transmission and ventilation, P_{acl} - the power consumed by the air conditioning plant, P_{dhw} – the power consumed by the domestic hot water circuit. The equilibrium between the thermal power, provided by the sources (Stirling engine and pellet boiler), and the power consumed for building heating/conditioning and for the domestic water is achieved through the water temperature control in the thermal accumulator to a setpoint, T^{sp}. Whereas the thermal power of the Stirling engine is a random variable, resulted from the balance control of the load within the electrical subsystem, the temperature control is done by adjusting the power of the pellet boiler. This boiler was modelled as a dynamic first order system with a time constant of 100 sec. The second thermal balance equation is the following:

$$m_a c_a \frac{dT_a}{dt} = k_c(T - T_a) - P_{tv}(T_a, t_e, \boldsymbol{\xi}) \tag{6}$$

where: T_a is the temperature in the building, m_a, c_a – mass and specific heat of the air in the building, k_c – convection transfer coefficient and P_{tv} – power consumed in the house to cover losses through transmission and ventilation. The function $P_{tv}(T_a, t_e, \boldsymbol{\xi})$ depends on the outside temperature, t_e, and other variables that define the thermal regime of the building, $\boldsymbol{\xi}$. This function is calculated on the basis of the thermal balance of the building. The model of the thermal balance includes the relationships detailed for the calculus of the thermal flux components in heating regime (in winter) and in air conditioning using an adsorption plant (in summer). Within this balance the following values were considered: t_e = -17 ^{0}C and T_a = 20 ^{0}C, in winter regime, and t_e = 38 ^{0}C and T_a = 22 ^{0}C – in summer regime. Besides the thermal load corresponding to the heating/house conditioning, a significant contribution is given by the power variable $P_{dhw}(t)$ that is consumed in the domestic water circuit. The daily evolution of this variable has been adopted through a plausible graphic, presented in Figure 4.

The power produced by the solar collector is calculated by the relation:

$$P_{PT} = A\left[I_t \eta_0 - k_c(T_i - T_a)\right] \tag{7}$$

where I_t is the total solar irradiance [W/m^2], η_0 - the optical efficiency of the collector (η_0 = 0.9), T_i and T_a are the input temperature of the water in collector and the outside air temperature (variable during the day) respectively, A – the collector surface. On the basis of the daily evolutions of the solar irradiance and outside temperature, for the month of June, the daily variation of the solar collector power is illustrated in Figure 5.

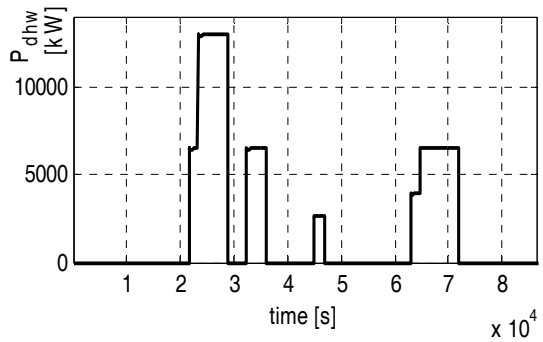


Fig. 4 Daily variation of the power variable $P_{dhw}(t)$

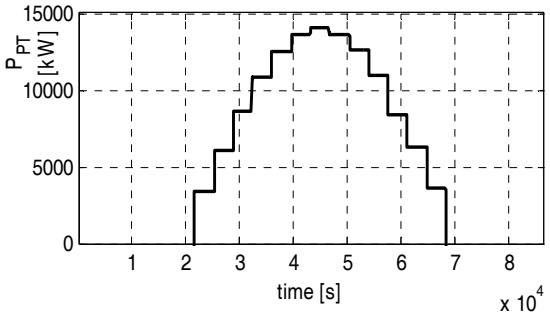


Fig. 5 Daily variation of the power variable $P_{PT}(t)$

Overall, the entire energetic autonomous plant is treated as a dynamic system heaving the following state vector:

$$\boldsymbol{x} = [W_B \quad P_{St} \quad T \quad T_a \quad P_{ph}]^T \tag{8}$$

in which the first two components correspond to the electrical subsystem and the other – to the thermal subsystem. The two subsystems are strongly inter-connected through the variables: $P_{st}(t)$, (being in a 3:1 ratio with $P_{se}(t)$), through the vector $u_p(t)$, that influences the power $P_{sl}(t)$ respectively.

IV. Control of the Autonomous Energetic System

Even the structure of the controlled system suggests the use of a multivariable controller, in this stage a simple solution as a decoupled loop has been adopted (Figure 6). The thermal process controller, R_T, adjusts the power variable $P_{ph}(t)$, and the battery voltage control is done by the adjusting of power variable $P_{se}(t)$, using the controller R_V and the anticipation controller R_A. The last mentioned controller gives a feed-forward control as a function of the measured variable $P_{al}(t)+P_{sl}(t)$. Considering that the elements controlled by the voltage and temperature controllers (the Stirling engine and the pellet boiler) have a steady-state characteristic of saturation type, antiwind-up systems have been provided for these controllers. An important difficulty in the achievement of the Stirling engine control consists in the fact that it accepts only discrete controls (0/1), within the range 0 – 1.6 kW. For higher controls, the control could be performed through modulation (the continuous adjustment of the power). For the implementation of the voltage controller, a conversion block of the control signal has been designed. It receives at the input the continuous signal from the controller R_V and gives at the output a signal in a joint representation: continuous signal, identical to the input one, when a power higher than 1.6 kW is required, or a PWM signal, with the average value equal to the input signal, for powers within the range 0 – 1.6 kW. Figure 7 illustrates the operating of the conversion block. The time modulated pulse period is 1000 sec. that means 17 minutes approximately, so that the request of the Stirling engine supplying mechanism is not great.

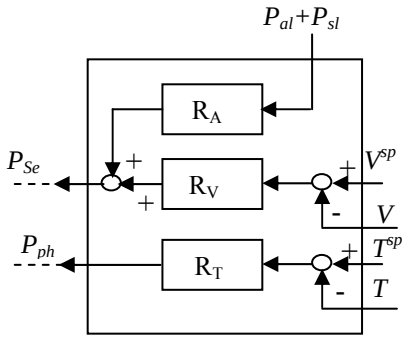


Fig. 6 Controller structure

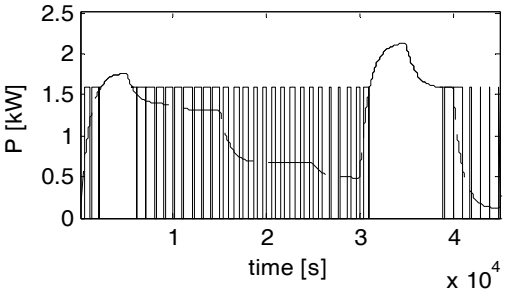


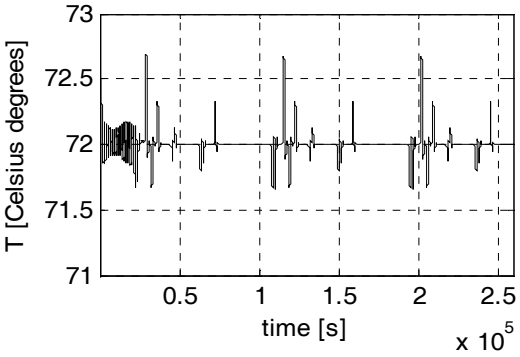
Fig. 7 Controls applied to the input (dash line) and output (solid line) of the PWM block

V. Results Obtained Through Simulation

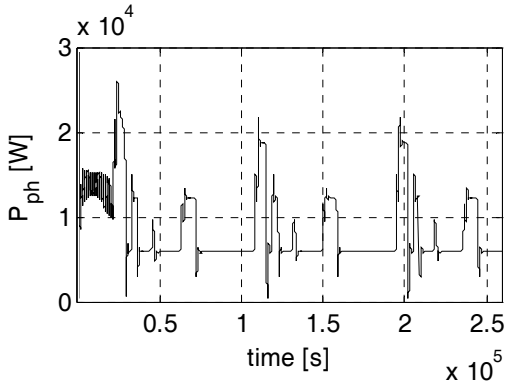
The objective followed through simulation was the preliminary validation of the autonomous energetic system feasibility, on the basis of the mathematical model

determined in the present paper. In the same time the performances of the control system were investigated. Considering the fact that the processes are slow, the simulation was performed on a 3 days horizon, aiming to obtain the permanent regime to the variations of the battery voltage and of the temperatures in the thermal accumulator and in the building.

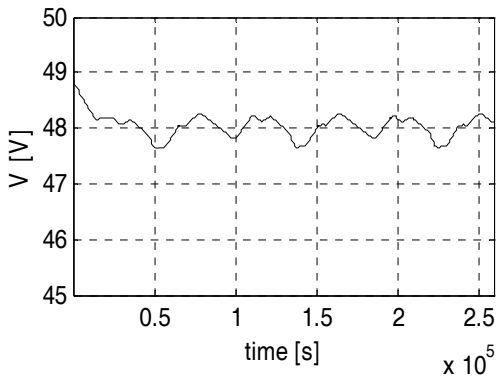
Figures 8, a and b present the evolutions of the temperature in the thermal accumulator, $T(t)$, and of the pellet boiler power, $P_{ph}(t)$, in winter regime. The setpoint for the temperature loop was chosen $T^{sp} = 72\ ^0\text{C}$. In Figures 8c,...,f are given the evolutions of the variables $V(t)$, $P_{al}(t)+ P_{sl}(t)$, $P_{st}(t)$, $P_{ph}(t)$, in summer regime. The controller of the electrical subsystem is able to maintain the voltage around the setpoint $V^{sp} = 48\ \text{V}$, even the consumed power $P_{al}(t)+ P_{sl}(t)$ has great variations, including the ones given by the variable term $P_{sl}(t)$, that is determined by the operating of the conditioning plant. The great consumption of the electrical energy in the summer regime imposes the use of two Stirling engines. The total thermal power of the two Stirling engines, in a 3:1 ratio with the electrical power, is shown in Figure 8e. In this graphic it can be seen the operating periods in PWM controlling regime of the two engines. Obviously, this regime influences the pellet boiler evolution too (Figure 8f).



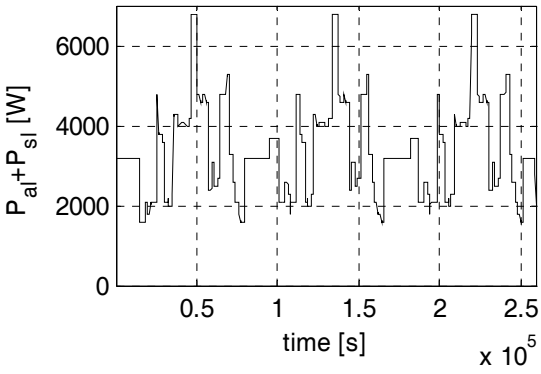
a



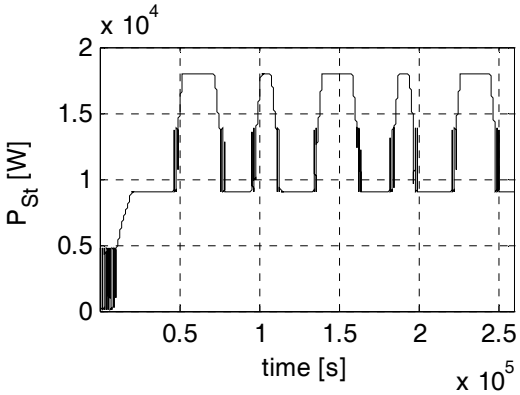
b



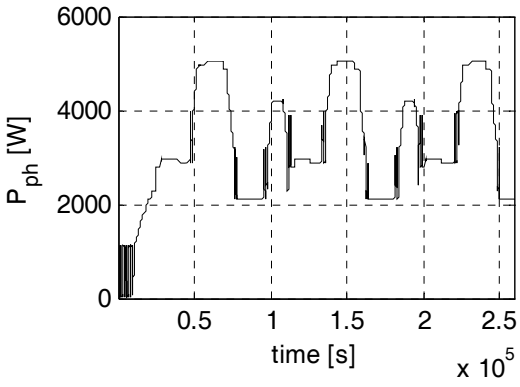
c



d



e



f

Fig. 8 Evolutions of the variables $T(t)$ and $P_{ph}(t)$ in winter regime (a and b) and $V(t)$, $P_{al}(t)+ P_{sl}(t)$, $P_{st}(t)$, $P_{ph}(t)$, in summer regime (c,d,e and f)

VI. CONCLUSIONS

The autonomous energetic systems that are based on renewable resources, of the type presented in the paper, represent a field recently approached in the literature, in connection to the concept of intelligent building. A mathematical model, which can be considered appropriate for the feasibility analysis of the autonomous energetic systems has been elaborated in the paper. The results obtained through numerical simulation have shown that the model can be useful for the investigation of advanced control solutions for these systems.

ACKNOWLEDGMENT

The authors would like to acknowledge to EEA Financial Mechanism for financing the research on "Integrated micro CCHP – Stirling Engine based on renewable energy sources for the isolated residential consumers from South-East region of Romania (m-CCHP-SE)", under the contract No. RO-0054/2009.

REFERENCES

[1] H. Cho, R. Luck, S.D. Eksioglu, L.M. Charma, "Cost-optimized real-time operation of CHP systems", *journal of Energy and Buildings*, Vol. 41, 2009, pp. 445-451.

[2] D. Chasapis, V. Drosou, I. Papamechael, *A. Aidonis,* R. Blanchard, "Monitoring and operational results of a hybrid solar-biomass heating system", *Journal of Renewable Energy,* No. 33, 2008, pp. 1759-1767.

[3] S. Thiers, B. Aoun, B.H. Peuportier, Experimental characterization, modelling and simulation of a wood pellet micro-combined heat and power unit used as a heat source for a residential building, B.H. Poor, *Journal of Energy and Buildings*, Vol. 42, 2010, pp. 896-903.

[4] L. Castaner, S. Silvestre, *Modelling Photovoltaic Systems Using PSpice*, John Willey&Sons, 2002.

The Influence of Chromatic and Luminance Noise in Real-Time Object Recognition Using Scale-Invariant Descriptors

Lucian Carata and Vasile Manta

Abstract— **It is largely accepted that algorithms like Scale Invariant Feature Transform (SIFT) or Speed-Up Robust Features (SURF) are highly robust, a quality that makes them practical in a large range of applications concerning object recognition, tracking or reconstruction. This paper sets to determine whether the observed instability of some of the interest points determined with such algorithms is a result of the algorithm's sensitivity to noise, performing a detailed analysis of the way stability is affected by camera chromatic and luminance noise. Those are the most common types of noise that appear in image acquisition systems, and are difficult to control due to their time-varying and non-deterministic nature. The analysis is performed on 4 sets of images ($\sim$ 200 images each), captured with commodity hardware and with no post-processing, assuring realistic conditions. The implications of the results are discussed, along with possible improvements that could increase interest point stability.**

I. INTRODUCTION

Algorithms like SIFT [1] or SURF [2] have been routinely used in object recognition, and are currently seen as a classic approach for solving the problem, especially when detection is necessary in the presence of occlusions. Due to their remarkable proprieties (like invariance to rotation, scale and robustness) there are a large number of researchers considering the adoption of such (or similar) algorithms in a broad range of applications, ranging from deformation measurement to 3D magnetic resonance image (MRI) classification.

While both of the algorithms use the cascade filtering approach - in which only the features that have the highest probability of remaining stable are retained at each filtering step, while computationally intensive stability checking is applied to fewer interest points - a typical real-time data acquisition and processing setup requires a compromise between high quality interest point filtering and fast image processing. As a consequence, virtually all the implementations (including the author's) suffer from the presence of unstable interest points in the final results.

In object recognition scenarios, this doesn't usually pose any problems: even if a number of initially detected interest points are not detected in subsequent images, there are sufficient remaining (stable) POI's to perform the recognition with a high degree of certainty. In similar terms, small changes in the POI's location relative to the detected feature are accepted, and interest point matching can be performed without difficulty.

L. Carata, V. Manta - Faculty of Automatic Control and Computer Engineering, "Gh. Asachi" Technical University of Iasi, Prof.dr.doc. Dimitrie Mangeron street, nr. 27, Romania. lucian.carata@yahoo.co.uk, vmanta@cs.tuiasi.ro

However, applications like real time object or shape tracking, deformation measurement or object modeling, that could consider using descriptors similar to the ones computed by SIFT or SURF, are more interested in POI stability (both in time and in position), for obtaining accurate results. This is the reason why such applications can benefit from a detailed descriptor stability analysis.

We have started our work from a simple and direct observation of the algorithm's[1] typical results: interest point instability exists even if the filmed scene, viewing perspective and environment conditions remain the same. Because the algorithm is deterministic, such behavior can only be attributed to time-varying noise in the acquired video stream.

A. Related work

While the problem of noise influence on the interest points has been considered in a number of publications, the results are somewhat hard to interpret, due to the lack of real world noise data, ambiguous testing conditions and implicit assumptions. In [3], a comprehensive evaluation and comparison of several algorithms and implementations is performed, regarding descriptor invariance to rotation, scale change, image noise and affine transformations. The noise test claims that noise (up to 20%) significantly reduces the number of points of interest detected, while keeping recognition rates between 100% and 95%. However, interest point stability is not considered, and the noise data is probably (no assumptions are explicitly stated in the article) Gaussian noise added to a image set in post-processing.

Luo, in [4], considers only the repeatability of interest points when the image is blurred, a condition for which the SURF and SIFT algorithms are designed to be robust. The results show high repeatability percentages when the Gaussian blur radius is smaller than 1.5. Similar results are obtained by Mikolajczyk [5] and Evans [6], with the latter also considering jpeg compression artifacts.

It should be noted that here "repeatability rate" refers to the percentage of points simultaneously present in two images. In this paper, we prefer the term "stability", to mean the continuous presence of the same interest point in an arbitrarily large image sequence. The two notions can be considered equivalent.

In our approach, we consider different sets of real noise-affected data (images), and investigate the possible correlation between interest point instability and feature regions that

[1]Our implementation considers SURF, but similar results are expected for SIFT

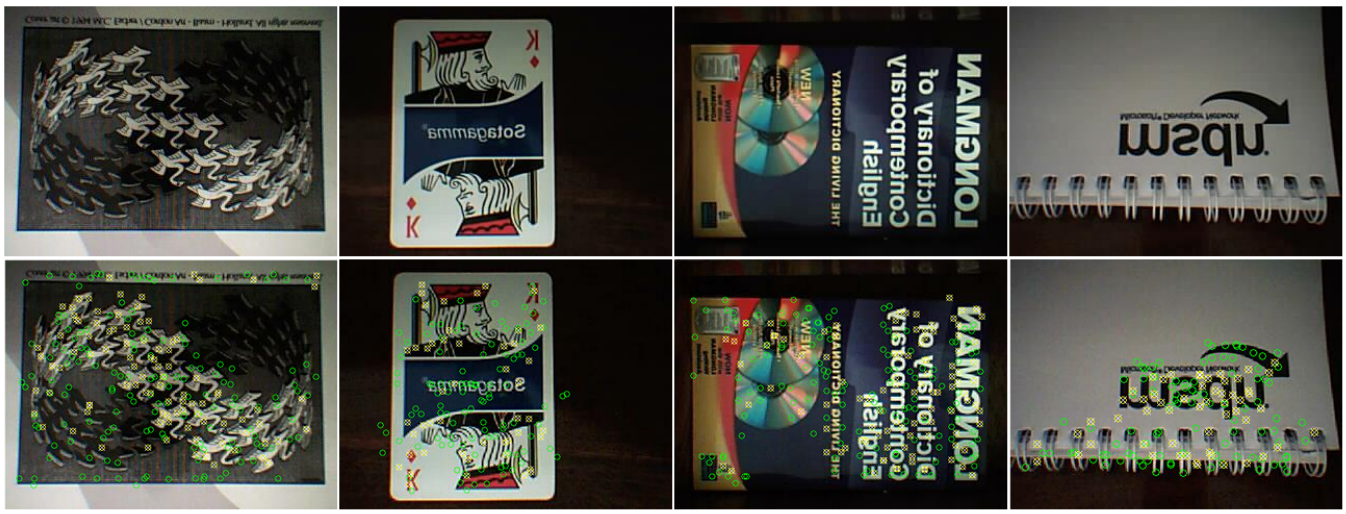


Fig. 1. The image datasets. In order: `gof`, `king`, `longman`, `notes`. The bottom row has POI data annotations. light markers - stable interest points; dark markers - unstable interest points

are affected by noise. We believe this method gives a better picture of actual descriptor behavior.

II. IMAGE DATA ANALYSIS AND THEORETICAL FRAMEWORK

In our data analysis, we make use of 4 image sets (`gof`, `king`, `longman` and `notes`), presented in Fig. 1. Each image set is composed of $\sim$ 200 images, captured with an usual webcam, at resolutions of 320×240.

In order to simplify the correlation of POI's between images and to obtain accurate results, the filmed scene is static (the target objects are not moved during the capturing). Also, every other environment variable (luminosity, environment reflections) is kept as stable as possible. The purpose of this is to ensure that all the variations in time between two images in the same set are due to camera noise (chromatic and luminance).

A. Measuring Noise

Given an interest point detected at scale σ_k, its location has been calculated by the algorithm considering the data from a rectangular window in it's vicinity. Let us formally define this region:

$$W_i(x, y) = w_{\sigma_k}(x, y) * I_i, \tag{1}$$

where $w_{\sigma_k}(x, y)$ represents the rectangular window function, centered on the k^{th} interest point and having the width and height equal to $3 \cdot (2\sigma_k + 1)$. I_i represents the image from which the interest point is computed.

Considering this, we can measure the cumulated noise between a region around a detected interest point in a reference image, and the same region in each of the other images from the set.

$$N_k = |W_r(x_k, y_k) - W_i(x_k, y_k)|, i = 1 \dots n, \tag{2}$$

where W_r is the region around the interest point from the random reference image and W_i is the same region in image i. While this relation does not provide absolute numerical noise measurements (because the reference image also contains noise), it calculates a measure that is related to the noise existent in the system:

$$W_r = W_r^* + e_1 \tag{3}$$

$$W_i = W_i^* + e_2, \tag{4}$$

where by convention W^* is the ideal image of the region (no noise), and e is the noise component. Considering that we do not modify the scene, we have $W_r^* = W_i^*$ and we can deduce

$$N_k = |e_1 - e_2| \tag{5}$$

which gives us the difference due to noise that is "observed" by the algorithm.

B. Interest Points

By analyzing all the frames from a given dataset, we determine the set of stable points:

$$S = \bigcap_{f=r}^{n} P_f, \tag{6}$$

where r is considered a random reference frame, n is the number of captured frames, and P_f is the set of all the interest points in frame f, computed by the algorithm. In a similar manner, the set of unstable points in frame f is given by:

$$U_f = P_f \setminus S \tag{7}$$

C. Interest Point Noise Measurements

The next step in processing the available data is to take the set of the determined stable POI's, and to compute an average noise (percentage) in the interest points region, for all the existing frames in a image set, for each of the color components of the frame (RGB). Then, we will do the same thing for unstable POI's, and compare the results.

TABLE I

INTEREST POINT STABILITY IN EACH OF THE DATASETS

gof	33.343%
king	31.4337%
longman	43.9981%
notes	35.9006%

Considering an intuitive result, we should find unstable points in areas that are more affected by noise. 100% noise means that a previously white pixel is black in a subsequent image. It is so that we take into account small variations in pixel lightness and color:

$$\overline{N_{R,G,B}} = \frac{1}{n-r}\sum_{f=r}^{n}\frac{1}{|S|}\sum_{k=1}^{|S|}\frac{100\cdot N_{k_{R,G,B}}}{255} \qquad (8)$$

III. EXPERIMENTAL RESULTS

First, we are interested in the number of interest points that remain stable throughout the frames in each set, starting with a random frame. We are starting with a random frame and averaging the outputs over different starting frames in order to eliminate particular starting frame influences on the final result. A typical result in such a case is presented in Fig. 2.

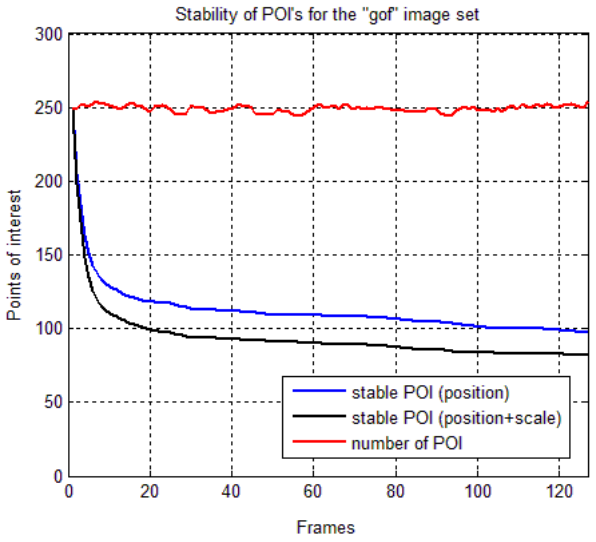


Fig. 2. POI stability iteratively considering dataset frames

We can observe that only approximately 34% of the interest points detected in the first frame are present in all the subsequent frames. The results for all the datasets can be consulted in Table I. Now, we will try to determine if this somewhat low percentage is caused by noise.

At a first look, Fig. 3 would suggest that noise directly affects interest point stability. Indeed, considering information from all the datasets, most of the results are very similar to those in Fig. 3: higher noise percentages are detected in unstable interest point regions. However, occasionally we

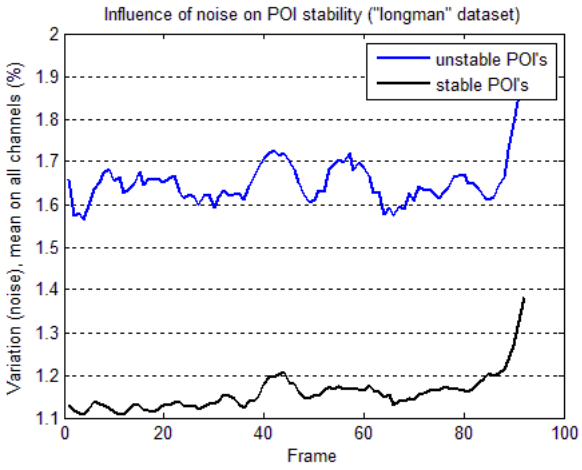


Fig. 3. Noise levels in interest point regions, longman dataset

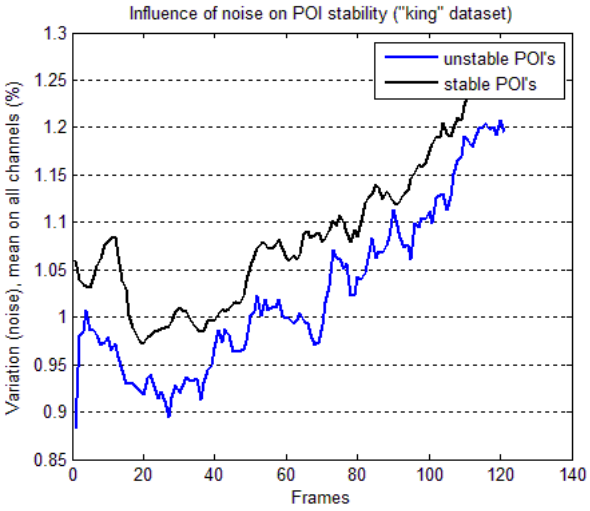


Fig. 4. Noise levels in interest point regions, king dataset

will get graphics such as those from Fig. 4. Those would appear to contradict our intuition regarding the expected results: for some of the sets, the noise from the regions surrounding stable interest points is actually larger than the noise for unstable POI's.

If noise is the only thing that varies between the images (and this is our case), such a result could have two major implications:

- The algorithm is not highly sensitive to noise, stable interest points can be determined even in noisy regions.
- There are other factors, besides noise, that make a region susceptible to POI instability. Because we have eliminated environment factors (that are kept fixed during the capture), those will have to be intrinsic to the image region.

The first thing we have looked for (regarding factors that could increase descriptor sensitivity to noise) is the contrast of the POI surrounding region. In order to test this hypothesis, we have calculated a contrast indicator for each

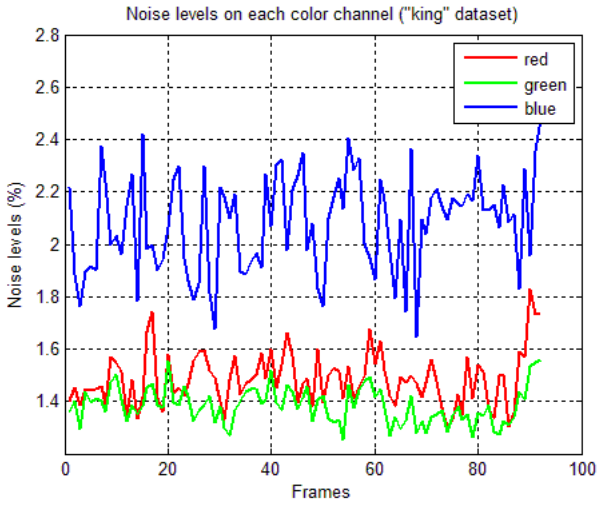


Fig. 5. Noise levels by color channel, `king` dataset

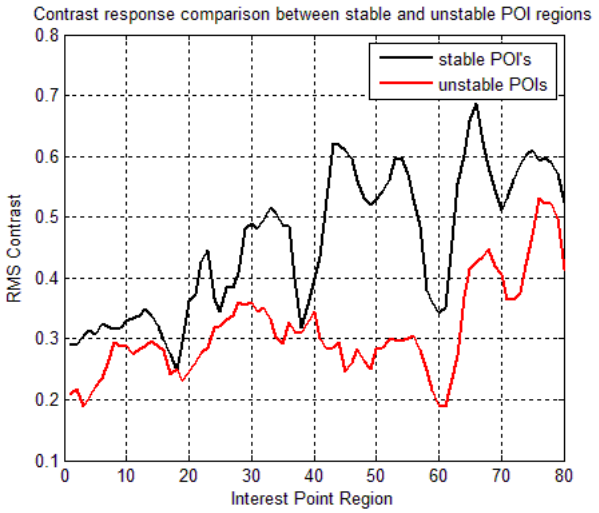


Fig. 6. Contrast levels for stable and unstable regions, `longman` dataset

of the regions in the vicinity of stable POI's, one for each region near unstable POI's, and performed a comparison between the two values. In our calculations, we have used Root Mean Squared (RMS) Contrast, that is defined as the standard deviation of the pixel intensities:

$$C_{RMS}(k) = \sqrt{\frac{1}{4\sigma_k^2}\sum_{i=0}^{2\sigma_k}\sum_{j=0}^{2\sigma_k}(W_{ij}(k) - \overline{W(k)})^2} \quad (9)$$

Intuitively, a region with low contrast (in which pixels do not vary much in intensity or color) is more susceptible to even small amounts of noise, that could create "artificial" local minimum or maximum values for the "Difference of Gaussians" (DoG) function:

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y), \quad (10)$$

where $G(x, y, \sigma)$ is the Gaussian kernel:

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (11)$$

While most of the POI's that are in regions with low contrast are eliminated in the SURF cascade filtering steps, some of them pass the filtering and appear in the final results. This also depends on the threshold value that is set as a parameter for the algorithm. So, observing a large number of unstable interest points might mean you should increase the thresholding value. However, the fact that region contrast plays an important role in POI stability can also be observed from our results, in Fig. 6.

It should be noted that for that particular result, there is a difference of almost 12% between the contrast of the stable interest point regions and the one for unstable POI's. Similar data is obtained for the other datasets, although the differences in contrast have sometimes lower values $(min = 3\%)$.

IV. CONCLUSIONS AND FURURE WORK

This paper has presented a detailed analysis of time-varying noise's (luminance and chromatic) influence on SURF descriptor's stability, considering various datasets. The results show that the algorithm can yield stable interest points even if noise levels are high. However, in some cases, other characteristics of the image might also influence SURF's sensitivity to noise. Contrast has been determined to be one of the significant ones.

A natural step toward improving algorithm output is therefore improving contrast for regions where high stability is desired. However, as Fig. 3 clearly shows, noise levels can not be considered completely irrelevant. If improving contrast is not possible, one could improve stability by lowering image noise levels. Because noise is usually not uniformly distributed among color channels, as can be clearly seen in Fig. 5, and SURF algorithms use monochromatic data, one could apply the processing by using only data from the color channels that are less affected by noise.

Future work should consider even more complex ways of removing chromatic and luminance noise, such as FFT filtering of the noisy frequences, keeping image detail intact (unlike the results obtained when applying a median filter).

REFERENCES

[1] D. G. Lowe, "Object recognition from local scale-invariant features," *Computer Vision, IEEE International Conference on*, vol. 2, pp. 1150–1157 vol.2, August 1999.

[2] H. Bay, T. Tuytelaars, and L. V. Gool, "Surf: Speeded-up robust features," in *ECCV*, 2006, pp. 404–417.

[3] J. Bauer, N. Sünderhauf, and P. Protzel, "Comparing several implementations of two recently published feature detectors."

[4] J. Luo and G. Oubong, "A comparison of sift, pca-sift and surf," *IJIP*, vol. 3, pp. 143 – 152, August 2009.

[5] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, October 2005.

[6] C. Evans, "Notes on the opensurf library," University of Bristol, Tech. Rep. CSTR-09-001, January 2009.

"Follow the Leader" Control for Multi-robot Formation with Hybrid Control Structure

Daniela Cristina Cernega and Razvan Solea, *Member, IEEE*

***Abstract*—The multi-robot formation control presented in this paper is based on solving the "follow the leader" problem based on a hybrid control structure. The two layers of the hybrid control structure are: the discrete control level and the continuous control level. The discrete event level is characterized through a discrete states set and a set of discrete events generating the state transitions. The discrete control level ensures the supervisory control. Each of the discrete states is characterized through a continuous dynamic (nonlinear). The trajectory tracking control problem at the continuous level is solved using the sliding mode control. The supervisor and the sliding mode controllers are presented, and also some results of the implementation of this control structure.**

I. Introduction

SINCE the genesis of robotics in the early 1960's, robots have played an increasingly large role in today's society [1]. Their applications have been as diverse as the scientifically inspired NASA Mars Rovers to domesticated robot vacuum cleaners. In fact, there are currently over 800 000 industrial robots in operation and over 600 000 household robots, mainly composed of lawn mowing and vacuum cleaning robots [2].

The multi-robots systems are an important robotics research field. Such systems are of interest for many reasons; tasks could be too complex for a simple robot to accomplish; using several simple robots can be easier, cheaper and more flexible than a single powerful robot [3], [4], [5], [6].

Research in robotic formations has focused on issues like formation generation [7], [8], maintenance of a formation shape [9], [10], controlling and changing formations [11], [12]. Generally, there are three broad approaches to the robotic formation problem available in the literature. They include the combined reactive behaviours [9], leader–follower strategies [10], [13], and virtual structures [14], [15]. A comprehensive review of robotic formation is given in [16].

In this study the hybrid leader-follower robot formation control is considered. The referenced robot is called *leader*, and the robot following it, is called *follower*. Thus, there are many pairs of leaders and followers and complex formations can be achieved by controlling relative positions of these pairs of robots respectively. This approach is characterized by simplicity, reliability and no need for global knowledge and computation.

The hybrid control structure consists of two levels: the discrete control level implementing the supervisory control, and the numerical control level using a sliding-mode controller to solve the trajectory tracking problem.

To control a multi-robot formation as a discrete event system means to follow a desired behaviour described through the imposed constraints. The desired behaviour is modeled as discrete event system for the entire formation and a supervisor is designed to achieve this behavior. The approach used to design the supervisor that proposed in [17], and in [18].

The discrete control level is coupled with the numerical control level and it detects the functioning situations. Each functioning situation is a discrete state. Each discrete state is characterized with a continuous model as shown in [19]. For each of the situations detected at the discrete event level the appropriate continuous model is selected together with the corresponding continuous controller. For each discrete state the references of the corresponding continuous controllers are also established.

The model for a hybrid automaton (HA), is defined with:

$$HA = \left(X, Q, \mu_1, \mu_2, \Sigma, \mu_3, Q_0, Q_f \right) \quad (1)$$

where:

- X is the vectorial space of the system state x, the continuous state vector of the system, denoted by $x = \left[x_1 \ldots x_n \right]^T \in X$, supposed to be continuos observable vectorial.
- Q the set of discrete states corresponding to all the possible phases, $Q = \{q_i, i=1\ldots m\}$; the hybrid state of the system is defined within the pair $(x, l) \in X \times Q$;
- μ_1 is the set of the m vectorial fields associated to each discrete phase;
- μ_2 is the set of the constraints associated to each discrete phase;
- Σ is the set of the events;
- μ_3 is the set of functions associated with the events;
- Q_0 is the set of the initial states and Q_f is the set of final sates.

In the next section the hybrid control system is presented.

Manuscript received April 30, 2010. This work was supported by CNCSIS-UEFISCSU, project PNII-IDEI 506/2008.

D. C. Cernega is with the Department of Control Systems and Industrial Informatics, Computer Science Faculty "Dunarea de Jos" University of Galati, Domneasca 47, 800008, Galati, Romania (corresponding author to provide phone/fax: +40-236-460182; e-mail: daniela.cernega@ugal.ro).

R. Solea is with the Department of Control Systems and Industrial Informatics, Computer Science Faculty "Dunarea de Jos" University of Galati, Domneasca 47, 800008, Romania (e-mail: razvan.solea@ugal.ro).

The discrete dynamic event system model for the multi-robot formation is obtained. The supervisor to ensure the desired behavior of the system is designed. The continuous models for each discrete state are also presented.

The wheeled mobile robot is a nonlinear system. The continuous control level is dedicated to the trajectory tracking control [20] - [24]. The trajectory tracking control problem is solved using the sliding mode control. In the Section 4 this problem is solved and the control laws together with the references for the discrete states are obtained. Section 5 is dedicated to the results obtained after implementing this hybrid structure.

II. The Hybrid Control Structure

The hybrid control structure proposed in this paper is shown in Fig.1.

To control the multi-robot formation as a discrete event system leads to a supervisor design. The discrete event model used is the automaton called *G*, defined as follows:

$$G = (Q, \Sigma, \delta, q_0, Q_m) \quad (2)$$

where Q is the set of the discrete states physically possible of the system, Σ is the set of all the events, δ is the transition function of the automaton, q_0 is the initial state and Q_m is the set of the marked states of the system. The events in the follow the leader-formation are from two distinctive categories: Σ_u is the set of the uncontrollable events and Σ_c is the controllable events set. The controllable events are subject of the control action and these events can be enabled and disabled at any time i.e. from any state. The uncontrollable events cannot be enabled or disabled by the control action.

Supervisory control for this discrete event system has the objective to ensure the desired behavior of the follower robot according to some constraints imposed.

The supervisor design for this problem is based on sonar data. The *sensitivity sphere* is a concept defined in order to establish the smallest distance equal to the length of the *follower* robot in order to avoid collision with the *leader* or with other obstacles. The sensitivity sphere is represented in Fig. 2.

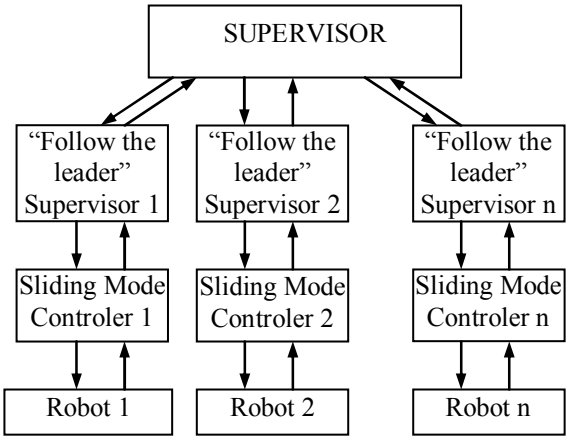


Fig. 1. Hybrid Control Structure

The analysis of the robot motion according to the defined sensitivity sphere, for this problem generates seven cases:

-*case 1*: the sonar 3 and 4 detect an object inside the sensitivity sphere and the robot will receive references to move ahead;

-*case2*: sonar 2 and 3 detect an object inside the sensitivity sphere and the supervisor generates references for the robot motion the constant distance *d* and the angle *α*;

- *case3*: sonar 4 and 5 detect an object inside the sensitivity sphere and the supervisor generates references for the robot motion the constant distance *d* and the angle -*α*;

- *case4*: sonar 1 and 2 detect an object inside the sensitivity sphere and the supervisor generates references for the robot motion the constant distance *d* and the angle 2*α*;

- *cas5*: sonar 5 and 6 detect an object inside the sensitivity sphere and the supervisor generates references for the robot motion the constant distance *d* and the angle -2*α*;

- *case6*: any sonar detects an object inside the sensitivity sphere closer then the minimum allowed distance and the supervisor generates references for the robot to stop;

- *case7*: no sonar pair detects an object and the supervisor generates references for the robot circular motion in order to search the leader.

These cases are generating the discrete states set, Q, of the automaton *G,* model of the process defined in (2).

The objective of the supervisory control for the *follower* is to track the *leader* when in the environment there are some unknown obstacles identified within the sensitivity sphere, or another leader appears inside the sensitivity zone. The discrete events generating discrete state transitions in this system are:

- $\Sigma_c = \{ e_{c0}, e_{c1}, e_{c2}, e_{c3}, e_{c4}\}$, where e_{c0} - the *start* command, e_{c1} - the distance established between the two robots is respected, e_{c2} – start the distance evaluation, e_{c3} – command the robot movement with speed references inside the established limits, e_{c4} - the sonar data are valid;

- $\Sigma_u = \{e_1, e_{u1}, e_{u2}, e_{u3}, e_{u4}\}$, where e_1 - end initialization, e_{u1} - an obstacle appeared in the interior of the sensitivity sphere, e_{u2} the leader is lost, e_{u3} reading errors from sonar detected, e_{u4} another leader appeared inside the sensitivity sphere.

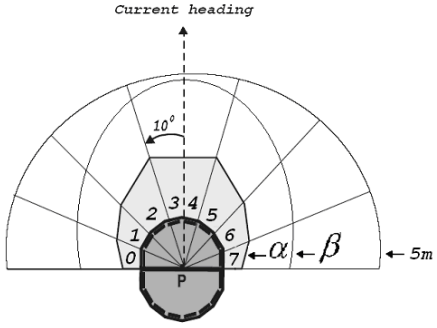


Fig. 2. The sensitivity sphere of robot

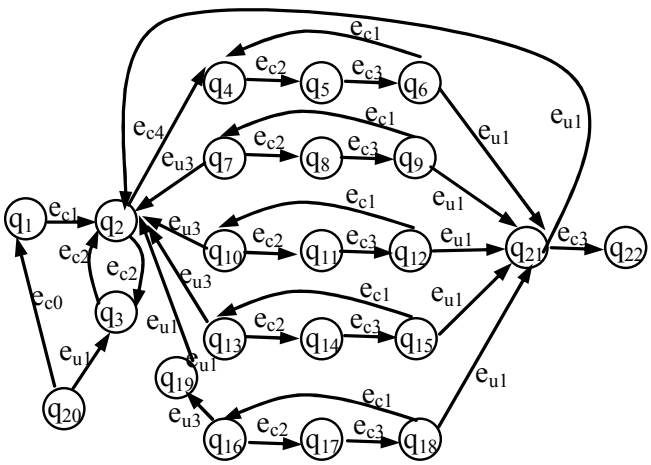


Fig. 3. The automaton *G*, model of the discrete event system.

The discrete state set, Q contains the states defined as follows: q_1 robot initialization; q_2 sonar reading; q_3 nearest limit verification for all the sonar; q_4 data analysis from sonar 3 and 4; q_5 trajectory tracking algorithm for *case 1*, q_6 movement according to *case1* references, q_7 data analysis from sonar 2 and 3, q_8 trajectory tracking algorithm for *case 2*, q_9 movement according to *case2* references; q_{10} data analysis from sonar 4 and 5; q_{11} trajectory tracking algorithm for *case 3*; q_{12} movement according to *case3* references; q_{13} data analysis from sonar 1 and 2, q_{14} trajectory tracking algorithm for *case 4*, q_{15} movement according to *case4* references, q_{16} data analysis from sonar 5 and 6, q_{17} trajectory tracking algorithm for *case 5*, q_{18} movement according to *case5* references, q_{19} 180 degrees rotation, q_{20} STOP, q_{21} corresponds to the situation when an obstacle appears inside the sensitivity sphere during the movement corresponding to one the states q_6, q_9, q_{12}, q_{15}, q_{18}; this state provides the supervisor the ability to avoid collisions, q_{22} is the state to be avoided with the supervisor control: the collision state.

The transition function, δ,of the automaton is represented in Figure 3.

III. Leader-Following Formation Models

Figure 4 is a leader-following control model where the formation pattern is specified by the separate distance *d* and the relative bearing ψ for two robots r_1 and r_2. The desired formation pattern can be defined as the desired separate distance d^d and the relative bearing ψ^d. The follower r_2 regulates the formation state errors of the separate distance and the relative bearing through its speed control signals $u_{r2} = [v_{xr2}\ \omega_{r2}]^T$.

$$\begin{bmatrix} \tilde{d} \\ \tilde{\psi} \end{bmatrix} = \begin{bmatrix} d^d \\ \psi^d \end{bmatrix} - \begin{bmatrix} d \\ \psi \end{bmatrix} \tag{3}$$

The relative distance between the leader and the follower robot is denoted as *d*, the separation bearing angle is ψ, and

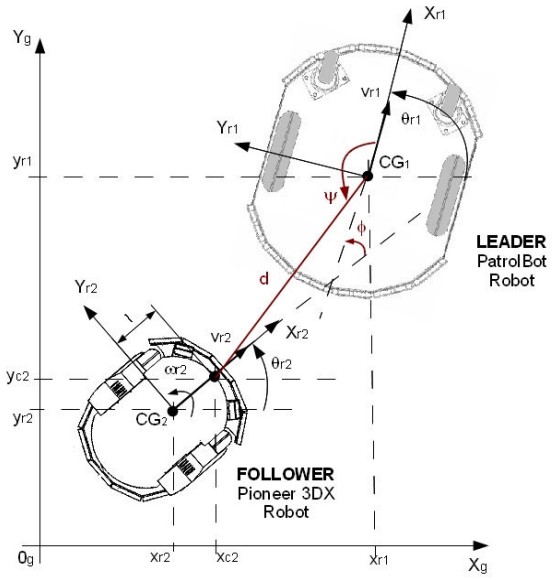


Fig. 4. Leader-following formation models.

they are given by:

$$d = \sqrt{(x_{r1} - x_{c2})^2 + (y_{r1} - y_{c2})^2} \tag{4}$$

$$\psi = \pi - [\theta_{r1} - \arctan 2(y_{r1} - y_{c2}, x_{r1} - x_{c2})] \tag{5}$$

where:

$$x_{c2} = x_{r2} + l \cdot \cos(\theta_{r2}), \quad y_{c2} = y_{r2} + l \cdot \sin(\theta_{r2})$$

The formation control can be investigated by modeling the formation state error as follows [9]:

$$\begin{bmatrix} \dot{\tilde{d}} \\ \dot{\tilde{\psi}} \end{bmatrix} = G \cdot u_{r2} + F \cdot u_{r1}, \quad \dot{\phi} = \omega_{r1} - \omega_{r2} \tag{6}$$

and

$$G = \begin{bmatrix} -\cos(\phi + \psi) & -l \cdot \sin(\phi + \psi) \\ \dfrac{\sin(\phi + \psi)}{d} & -\dfrac{l \cdot \cos(\phi + \psi)}{d} \end{bmatrix}, \quad F = \begin{bmatrix} \cos(\psi) & 0 \\ -\dfrac{\sin(\psi)}{d} & 1 \end{bmatrix}$$

where: $u_{ri} = [v_{ri}\ \omega_{ri}]^T$, $\phi = \theta_{r1} - \theta_{r2}$ and l is the distance between the robot position (x_{r2}, y_{r2}) and the robot hand position (x_{c2}, y_{c2}) as shown in Fig. 4.

IV. Sliding-Mode Controller Design

In a leader-follower configuration, with the leader's position given and once the follower's relative distance and angle with respect to the leader are known, the follower's position can be determined. To use the leader-following approach, it is assumed that the angular and linear velocities of the leader are known. In order to achieve and maintain the desired formation between the leader and follower, it is only need to control the follower's angular and linear velocities to achieve the relative distance and angle between them as specified. Therefore, the leader-following based mobile robot formation control can be considered as an extension of the tracking control problem of the nonholonomic mobile

robot.

A practical form of reaching the control law (proposed by Gao and Hung [25]) is defined as:

$$\dot{s}_i = -p_i \cdot |s_i|^{\alpha} \cdot \mathrm{sgn}(s_i), \quad 0 < \alpha < 1, \quad p_i > 0, \quad i = 1, 2. \quad (7)$$

This reaching law increases the reaching speed when the state is far away from the switching manifold, but reduces the rate when the state is near the manifold. The result is a fast reaching and low chattering reaching mode. Integrating (7) from $s_i = s_i(0)$ to $s_i = 0$ yields

$$t_i = \frac{1}{(1-\alpha) \cdot p_i} \cdot s_i(0) \cdot (1-\alpha) \quad i = 1, 2$$

A new design of sliding surface is proposed, such that distance between the leader and the follower robot, *d*, and the separation bearing angle, ψ, are internally coupled with each other in a sliding surface leading to convergence of both variables. For that purpose the following sliding surfaces is proposed:

$$s_1 = \dot{\tilde{d}} + k_d \cdot \tilde{d} \quad (8)$$

$$s_2 = \dot{\tilde{\psi}} + k_{\psi} \cdot \tilde{\psi} + k_0 \cdot \mathrm{sgn}(\tilde{\psi}) \cdot |\phi| \quad (9)$$

where k_0, k_d and k_{ψ} are positive constant parameters and $\tilde{d}, \tilde{\psi}, \phi$ are defined by (3).

If s_1 converge to zero, trivially $\tilde{d}$ converge to zero. If s_2 converge to zero, in steady-state it becomes $\dot{\tilde{\psi}} = -k_{\psi} \cdot \tilde{\psi} - k_0 \cdot \mathrm{sgn}(\tilde{\psi}) \cdot |\phi|$. Since $|\phi|$ is always bounded, the following relationship between $\tilde{\psi}$ and $\dot{\tilde{\psi}}$ holds: IF $\tilde{\psi} < 0 \Rightarrow \dot{\tilde{\psi}} > 0$ and IF $\tilde{\psi} > 0 \Rightarrow \dot{\tilde{\psi}} < 0$.

From the time derivative of (8) and (9) and using the reaching law defined in (7) yields:

$$\dot{s}_1 = \ddot{\tilde{d}} + k_d \cdot \dot{\tilde{d}} = -p_1 \cdot |s_1|^{\alpha} \cdot \mathrm{sgn}(s_1) \quad (10)$$

$$\dot{s}_2 = \dddot{\tilde{\psi}} + k_{\psi} \cdot \ddot{\tilde{\psi}} + k_0 \cdot \mathrm{sgn}(\tilde{\psi}) \cdot \mathrm{sgn}(\phi) \cdot \dot{\phi} = \\ = -p_2 \cdot |s_2|^{\alpha} \cdot \mathrm{sgn}(s_2) \quad (11)$$

After some mathematical manipulation, one can achieve:

$$\dot{v}_{c2} = \frac{p_1 \cdot |s_1|^{\alpha} \cdot \mathrm{sgn}(s_1) + k_d \cdot \dot{\tilde{d}} - C_1}{\cos(\phi + \psi)} \quad (12)$$

$$\dot{\omega}_{c2} = \frac{\left(p_2 \cdot |s_2|^{\alpha} \cdot \mathrm{sgn}(s_2) + k_{\psi} \cdot \ddot{\tilde{\psi}}\right) \cdot d - C_2}{l \cdot \cos(\phi + \psi)} \quad (13)$$

where

$$C_1 = l \cdot \dot{\omega}_{r2} \cdot \sin(\phi + \psi) - d \cdot (\dot{\phi} + \psi) \cdot (\psi + \omega_{r1}) - \\ - \dot{v}_{r1} \cdot \cos(\psi) - v_{r1} \cdot \dot{\phi} \cdot \sin(\psi)$$

$$C_2 = k_0 \cdot \mathrm{sgn}(\tilde{\psi} \cdot \phi) \cdot \dot{\phi} \cdot d - \dot{v}_{r2} \cdot \sin(\phi + \psi) + \\ + d \cdot (\dot{\phi} + \psi) \cdot (\psi + \omega_{r1}) - \dot{v}_{r1} \cdot \cos(\psi) - v_{r1} \cdot \dot{\phi} \cdot \sin(\psi)$$

The *sgn* functions in the sliding surface were replaced by *saturation* functions, to reduce the chattering phenomenon [26].

V. Simulation Results

In this section, some simulation results are presented to validate the proposed control law. To show the effectiveness of the proposed sliding mode control law numerically, experiments were carried out on the multi-robot formation control problem.

High-level control algorithms (including desired motion generation) are written in C++ and run with a sampling time of Ts = 100 ms on a embedded PC, which also provides a user interface with real-time visualization and a simulation environment.

All the simulations was made using the MobileSim. MobileSim is software for simulating MobileRobots' platforms and their environments, for debugging and experimentation with ARIA. The ARIA software can be used to control the mobile robots like Pionner, PatrolBot, PeopleBot, Seekur etc. ARIA (Advanced Robot Interface for Applications) it is an object-oriented Applications Programming Interface (API), written in C++ and intended for the creation of intelligent high-level client-side software.

Figure 5 shows a block diagram of the proposed sliding-mode controller.

Wheel velocity commands,

$$\omega_R = \frac{v_{c2} + L \cdot \omega_{c2}}{R}; \qquad \omega_L = \frac{v_{c2} - L \cdot \omega_{c2}}{R} \quad (14)$$

are sent to the power modules of the follower mobile robot, and encoder measures N_R and N_L are received in the robots pose estimator for odometric computations.

Two simulation experiments were carried out to evaluate the performance of the sliding mode controller presented in Section 4. The first simulation refers to the case of circular trajectory (v_{r1} = 0.4 [*m/s*] and w_{r1} = 0.1 [*rad/s*]).

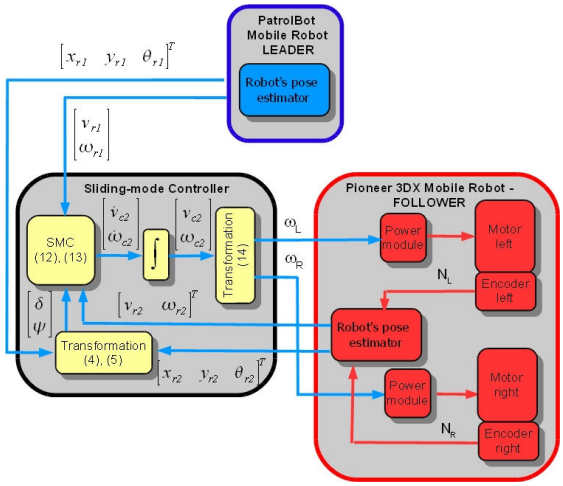


Fig. 5. Block diagram.

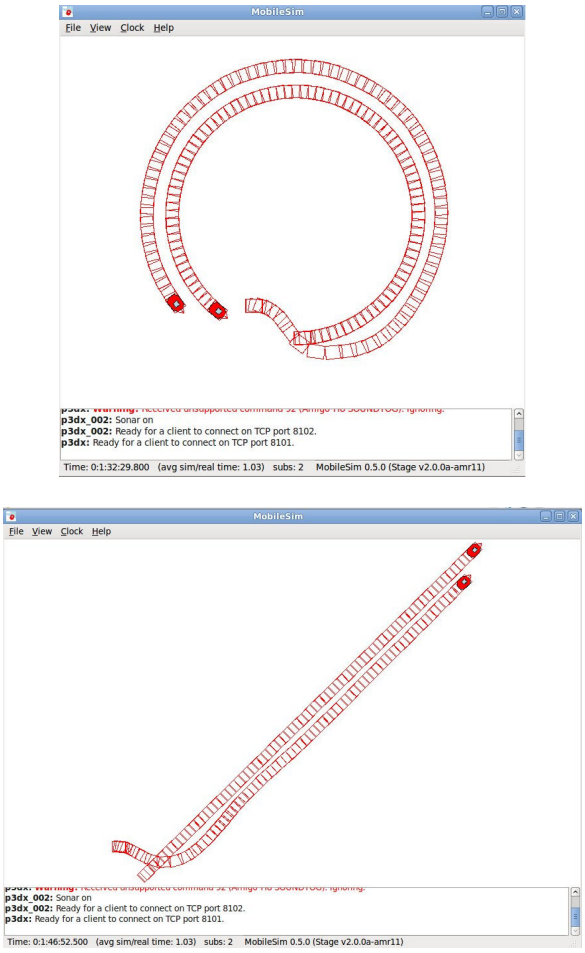


Fig 6. Simulation results using Aria and MobileSim software - Case I and Case II.

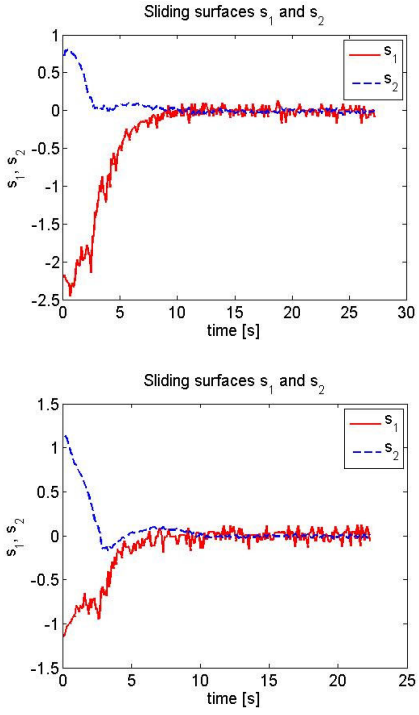


Fig. 7. Sliding surfaces s_1 and s_2 for case I and II.

The initial conditions of the leader and the follower are, $x_{r1}(0) = 0$, $y_{r1}(0) = 0$, $\theta_{r1}(0) = 0$, $x_{r2}(0) = -1.5$, $y_{r2}(0) = 1$, $\theta_{r1}(0) = 0$, $d^d = 1$ [m], $\psi^d = -135$[deg] (see Fig. 6).

In the second simulation the leader robot execute a linear trajectory but with a non-zero initial orientation ($\theta_{r1} = 45$ [deg]). The initial conditions of the leader and the follower in this second case are, $x_{r1}(0) = 0$, $y_{r1}(0) = 0$, $\theta_{r1}(0) = pi/4$, $x_{r2}(0) = -1$, $y_{r2}(0) = 1$, $\theta_{r2}(0) = 0$, $d^d = 1$ [m], $\psi^d = -135$ [deg] (see Fig. 6).

In Figs. 7 the sliding surfaces s_1 and s_2 asymptotically converge to zero.

The good performance for controlling the formation with the developed control law can be observed from Figs. 6 - 9.

The outputs of the formation system ($\tilde{d}$ and $\tilde{\psi}$) asymptotically converge to zero, as shown in Figs. 8 and 9.

VI. Conclusion

In this study a hybrid control structure to control a multi-robot formation is proposed. The hybrid control structure consists of two control levels: the discrete control level and the continuous control level. The discrete control level ensures the supervisory control and the continuous control level ensures the trajectory tracking control.

The desired formation, defined by two parameters (a distance and an orientation function) is allowed to vary in time. The effectiveness of the proposed designs has been validated via simulation experiments.

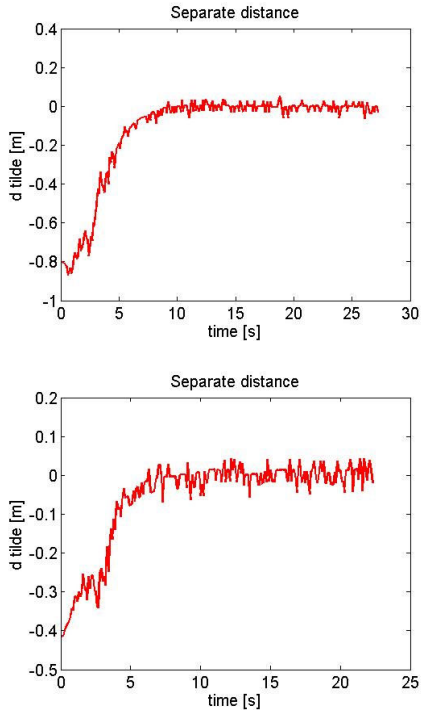


Fig. 8. Separate distance ($\tilde{d}$) for case I and II.

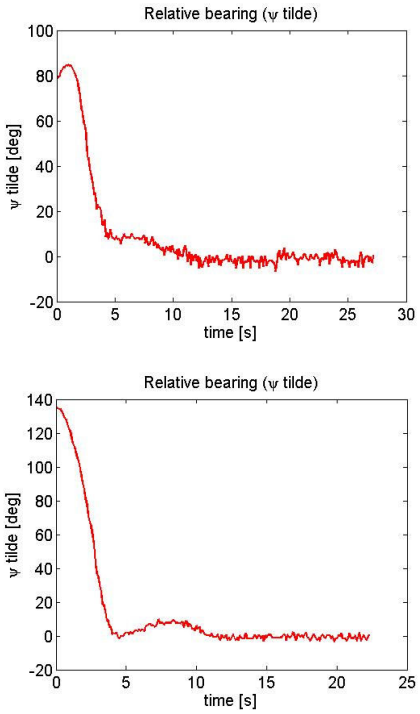


Fig. 9. Relative bearing ($\widetilde{\psi}$) for case I and II.

Simulation example is used to evaluate the sliding-mode algorithm and to show the application of the algorithm in practice. The controller is simply structured and easy to implement. From the simulation results, it is concluded that the proposed strategy achieves the effectiveness of desired performance.

Future research lines include the experimental validation of our control scheme and the extension of our results to skid-steering mobile robots. For the sake of simplicity in the present paper a single-leader, single follower formation has been considered. Future investigations will cover the more general case of multi leader, multi-follower formations.

ACKNOWLEDGMENT

This work was supported by CNCSIS-UEFISCSU, project PNII-IDEI 506/2008.

REFERENCES

[1] R. Murphy, "Introduction to AI Robotics". London: MIT Press, 2000.

[2] F. Lewis and G. Shuzhi, "Autonomous Mobile Robots: Sensing, Control, Decison Making and Applications", Boca Raton: CRC Press, 2006.

[3] R. M. Murray, "Recent research in cooperative control of multi-vehicle systems", *Journal of Dynamic Systems, Measurement and Control*, vol. 129(5), 2007, pp. 571-583.

[4] G. Klancar, D. Matko and S. Blazic, "Wheeled mobile robots control in a linear platoon", *Journal of Intelligent and Robotic Systems*, vol. 54(5), 2009, pp. 709-731.

[5] M. Mazo, A. Speranzon, K. Johansson and X. Hu, "Multi-robot tracking of a moving object using directional sensors", *IEEE International Conference on Robotics and Automation, ICRA '04*, vol. 2, 2004, pp. 1103-1108.

[6] M. M. Zavlanos and G. J. Pappas, "Dynamic assignment in distributed motion planning with local coordination", *IEEE Transaction on Robotics*, vol. 24(1), 2008, pp. 232-242.

[7] T. Arai, H. Ogata, and T. Suzuki, "Collision avoidance among multiple robots using virtual impedance", *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Tsukuba, Japan, 1989, pp. 479-485.

[8] H. Yamaguchia and T. Arai, "Distributed and autonomous control method for generating shape of multiple mobile robot group", *Proc. IEEE Int. Conf. on Robotics and Automation*, San Diego, CA, USA, 1994, pp. 800-807.

[9] T. Balch and R. Arkin, "Behavior-based formation control for multi-robotic teams", *IEEE Transaction on Robotics and Automation*, vol. 4, 1998, pp. 926-934.

[10] J. P. Desai, J. P. Ostrawski, and V. Kumar, "Modelling and control of formation of nonholonomic mobile robots", *IEEE Transaction on Robotics and Automation*, vol. 17, 2001, pp. 905-908.

[11] A. K. Das, R. Fierro, V. Kumar, J. P. Ostrowski, J. Spletzer, and C. J. Taylor, "A vision based formation control framework", IEEE Transaction on Robotics and Automation, vol. 18 (5), 2002, pp. 813-825.

[12] A. D. Nguyen, Q. P. Ha, S. Huang, and H. Trinh, "Observer-based decentralised approach to robotic formation control", *Proc. of the 2004 Australian Conference on Robotics and Automation*, Canberra, Australia, 2004, pp. 1-8.

[13] J. P. Desai, J. P. Ostrawski, and V. Kumar, "Control of changes in formation for a team of mobile robots", *Proceedings of the IEEE International Conference on Robotics and Automation*, Detroit, MI, 1999, pp. 1556-1561.

[14] J. Jongusuk and T. Mita, "Tracking control of multiple mobile robots: A case study of inter - robot collision free problem", *Proc. of the IEEE International Conference on Robotics and Automation*, Seoul, Korea, 2001, pp. 2885-2890.

[15] M. A. Lewis and K. H. Tan, "High precision formation control of mobile robot using virtual structures", *Autonomous Robots*, vol. 4, 1997, pp. 387-403.

[16] B. Erkin, S. Onur, and S. Erol, "A review: Pattern formation and adaptation in multirobot systems", Robotics Institute-Carnegie Mellon University, Pittsburgh CMU-RI-TR-03-43, 2003.

[17] P.J. Ramadge and W.M. Wonham, "Supervisory control of a class of discrete event processes". *SIAM J. of Control Optim.*, vol. 25 (1), 1987. pp. 206-230.

[18] M.H. Queiroz, J.E. Cury and W.M. Wonham, "Multitasking supervisory control of discrete-event systems". *In Discrete Event Dynamic Systems*, Kluwer Academic Publishers, vol. 15(4), 2005.

[19] P.J. Antsaklis and A. Nerode, "Hybrid control systems: An introductory discussion to the special issue", *IEEE Trans. on AC*, vol. 43(4), 1998, pp.457-460.

[20] D. Chwa, "Sliding-mode tracking control of nonholonomic wheeled mobile robots in polar coordinates", *IEEE Transactions on Control*. vol. 12(4), 2004, pp. 637–644.

[21] J.-M. Yang and J.-H Kim, "Sliding mode control for trajectory tracking of nonholonomic wheeled mobile robots", *IEEE Transactions on Robotics and Automation,* vol. 15(3), 1999, pp. 578–587.

[22] T. Floquet, J.P. Barbot and W. Perruquetti, "Higher-order sliding mode stabilization for a class of nonholonomic perturbed systems", *Automatica.* vol. 39(6), 2003, pp. 1077–1083.

[23] R. Solea and D. Cernega, "Sliding mode control for trajectory tracking problem - performance evaluation", In *Lecture Notes in Computer Science, Springer*, vol. 5769, 2009, pp. 865–874.

[24] R. Solea and U. Nunes, "Trajectory planning and sliding-mode control based trajectory-tracking for cybercars", *Aided Engineering, IOS Press*, vol. 14(1), 2007, pp. 33–47.

[25] W. Gao and J.C. Hung, "Variable structure control of nonlinear systems: A new approach", *IEEE Transactions on Industrial Electronics*, vol. 40(1), 1993, pp. 45–55.

[26] J. J. E. Slotine and W. Li, *Applied nonlinear control*, Prentice-Hall, 1991.

A Software Tool for image analysis of Enea's Tokamak based on a database system

M. Chinnici, S. Cuomo, S. Migliori, A. Quintiliani

***Abstract*— Imaging analysis is used for analyzing Enea-Frascati Tokamak Upgrade (FTU) - plasma. The overall data are organized in an open source portable software system based on a suitable database customized for the FTU. A software tool is developed in order to correlate the images with plasma signals and retrieve image information. Some Scientific Computing kernels of the software provides a comprehensive set of reference standard algorithms and graphical tools for image processing and analysis.**

I. Introduction

THE advent of the internet and extensive digital image libraries have entailed the development of rapid and efficient computer-based image searching and browsing techniques. The term content-based image retrieval refers to automatic recovery of images from a database based on a set of graphic features that qualify the images and that are, loosely speaking, similar to the characteristics of a given query image. This paper concerns the management through a database (the open source MySQL) and the application of retrieval techniques and analysis of growing database of plasma images coming from Frascati Tokamak Upgrade (FTU). Indeed, the huge amount of data produced requires the adoption of new technological solutions for data images handling. Data from FTU are acquired as movies or single frames. The wide number of information available by visual inspection of the data is often a limitation to real time experimental investigation. It appears that the images from the cameras can help to reconstruct some plasma phenomena and real time analysis could be useful for machine operation. On the other hand, due to the large number of data recorded, visual inspection of the movies is time consuming and often insufficient to correlate the images to other experimental findings. For a profitable use of the signals it is desirable to build specific tools for automatic processing of the images. The problems increase in complexity and cost in terms of hardware and software resources. In this research field the growing demand for computational resources has been a persistent goal of computational scientists. The challenge is
to solve these problems with a computational power that could be achieved inexpensively by collecting "distributed" resources. The main goal is to provide both efficient and low cost computing environment by sharing the computational
tools. This paper illustrates the acquisition system and the procedures developed for the processing and analysis of FTU images using ENEA-GRID technologies and middleware. In details, data are stored in a suitable database customized for the FTU, to be adopted in mathematical and statistical models in order to correlate the images with plasma signals and retrieve image information.

This contribution is organized as follows. In section 2 we introduce the scenario of FTU. In section 3 we present the "FTUsoftware" and the software implementation using the ENEA-technologies; in section 4 we give the image analysis instruments in order to analyse our data and we discuss the results of several retrieval experiments using the "FTUsoftware". Finally, conclusions are drawn in Section 5.

II. Scenario of Enea-Tokamak

Frascati Tokamak Upgrade (FTU) is a compact, high magnetic field tokamak experiment. The objectives of the machine are studying plasma transport, plasma heating and current drive in presence of strong additional Radio Frequency (RF) heating, and studying plasma profile by means of pellet injection. Tokamak producing 20-25 shots (plasma discharge) for days and each shots being 1,5s long and producing presently about 30-40 MB of data for 1400 channels. Data images from FTU are acquired as movies or single frames (in RGB-Red, Green, Blue) in real time; technology adopted as the storage architecture is AFS (Andrew File System) - AFS is a client/server architecture that allows to access files and directories residing on geographically distributed machines like a unique virtual machine, under the afs filesystem. In standard operating mode there is an experiment every 20–25 min. The FTU machine—from a control and data acquisition point of view —is composed of four main subsystems: the machine itself (torus, load assembly, cryostat, cooling system); the power supply (two fly-wheel generators, feeding the toroidal magnet and the poloidal windings respectively); RF facility and the diagnostic devices installed on the machine. The RF subsystem is made up of a lower hybrid system (LH: six gyrotrons operating at 8 GHz, total coupled power 2.5 MW), Ion-Bernstein Waves (IBW: three klystrons at 433 MHz, total coupled power 0.4 MW) and an electron cyclotron resonant heating system (ECRH: four gyrotrons at 140 GHz, total power 1.6 MW). The diagnostic devices —generally referred to as 'diagnostics'—include detectors covering the whole range of the electromagnetic spectrum, particle density control devices like DCN laser, or experiments like multi-pellet injection. In this paper, we focus our attention on images of plasma which are observed by wide 12 angle video-cameras placed inside the viewing ports and close to the plasma edge. The cameras monitor the status of the vacuum vessel and toroidal limiter and give useful data on the occurrence of several events such as arching and flying debris or even major damages such as the detachment of limiter plates. In details, the ports that we considered are:

[1]Marta Chinnici is with ENEA-UTICT-PRA, Casaccia Research Center, S. Maria di Galeria (Roma), Italy (corrispondig author e-mail: martachinnici@gmail.com).

Salvatore Cuomo is with University of Naples Federico II - Department of Mathematics and Applications "R. Caccioppoli", Napoli, Italy

Silvio Migliori is with ENEA-UTICT, Enea Sede, Lungotevere Thaon di Revel n. 76 - 00196 Roma, Italy.

Andrea Qintiliani is with ENEA-UTICT-PRA, Casaccia Research Center, S. Maria di Galeria (Roma), Italy.

Port_3_hor, Port_5_hor, Port_8_hor (where the suffix “hor” indicates horizontal video-cameras).

III. FTUsoftware on ENEA-GRID Middleware

In modern tokamaks visible and infrared video-cameras are becoming more and more important to monitor plasma evolution during fusion experiments. In the last years video-cameras have been extensively used in magnetic confinement fusion experiments for both the understanding of the physics and the safety of the operation. The images can be used not only to monitor the evolution of a plasma discharge but also to evaluate specific parameters, from the determination of impurity radiation to the distribution of power loads on the plasma facing components. Data analysis is normally performed off-line, due to the high amount of information to be processed, making the data acquired by the camera quantitatively useful only for post pulse evaluations. The main difficulty in using visible or infrared images for plasma feedback control is the fact that real-time image processing is challenging and heavy in terms of processing time, especially when complex tasks are required [3].

In order to manage, classify and process the images coming from FTU, we developed a software (“FTUsoftware”) that organizes the images in a database and allows to assess the information within the images. FTUsoftware is implemented by following the software engineering methodology. The main goal is to provide an integrated set of numerical and statistical tools and methods in a framework that is used to structure, plan and control the process of FTU data analysis. The software is developed in C++ language and the graphic interface is based on Open Qt libraries. The scientific computing kernel of the software is carried out on Problem Solving Environments Matlab that provides a comprehensive set of reference-standard algorithms and graphical tools for image processing and analysis [1].

The software is implemented by following the E/R Software Engineering model. The main entities of FTUsoftware are: `Ports`, `Cams`, `Folders` and `Images`. The overall data of the system are stored in a DataBase (named tokamak) that is built in MySql. The Fig.1 reports the Class Diagram.

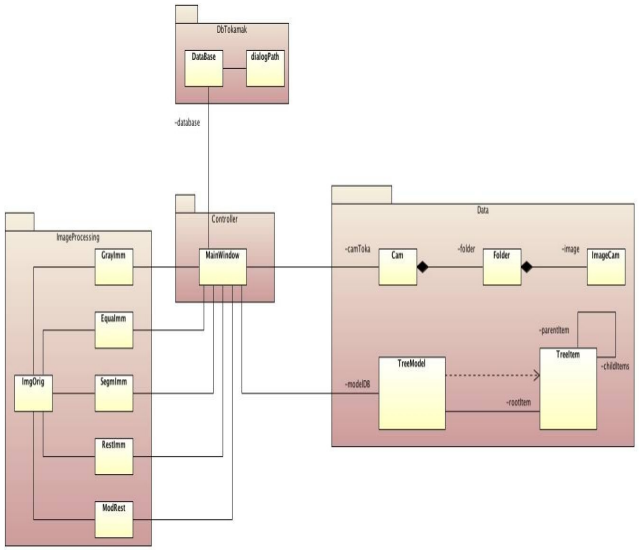


Figure 1: Scheme of “Tokamak” DataBase

In this paper we present a platform-based architecture for developing portable software in order to run it on any machine and any operating system without restrictions [6].

At the beginning it is necessary to set up the Middleware and install the appropriate libraries required to run the executable. **The first step** for accessing to images DataBase consists in inserting the IP numerical (e.g. 192.168.1.1) or string (e.g. “localhost”) data. Apart from the IP, the access requires to insert the name of the database, user and password; only authorized users can access to the platform. In a **second step** the system checks the existence of the file “config.conf” that gives us the information about the path of the images (Fig. 2).

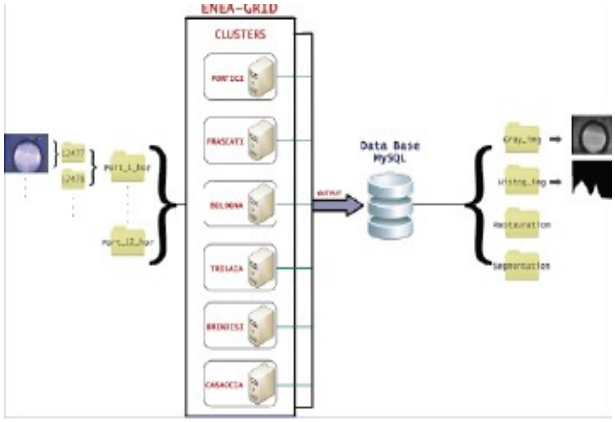


Figure 2: Sketch of System Pattern

The presence of the file “Config.conf” guarantees that the FTU images DataBase exists, thus it not necessary to find the images inside the server. On the contrary, if we do not find this file, it is obligatory to insert the path of the server path or machine’s folders.

Step three. We examine the path; in fact, the path must be structured in a way as showed in Fig. 3.

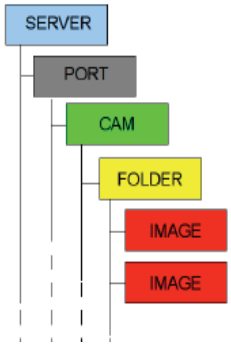


Figure 3: Path of the source data

Step four. FTUsoftware creates folders of our data and if the server has been correctly detected then it fills the folders recursively. Afterwards, FTUsoftware writes the "`config.conf`" file and the "`treeDb.txt`" file; the latter arranges the images of Database in order to create dynamical trees. After these phases, we may enter in the main page (Fig. 5) where we can operate on images. Fig 5 Operating window The main page is divided in boxes; the first box on the left includes two subfolders - Directory Server and Database. The first one contains the tree of the folders (inside the server) from which we pick up one or more images for analysis. The box has been set in order to show the last image under analysis, even if more images have been utilized. The second sub-folder – Database- shows the tree of the Data Base. At this point, the selected image is displayed in the box "Original Image".

When processed (e.g. gray-scale, restoration, segmentation)the image appears in the corresponding box (e.g. "Gray Scale" etc.) - Fig. 4.

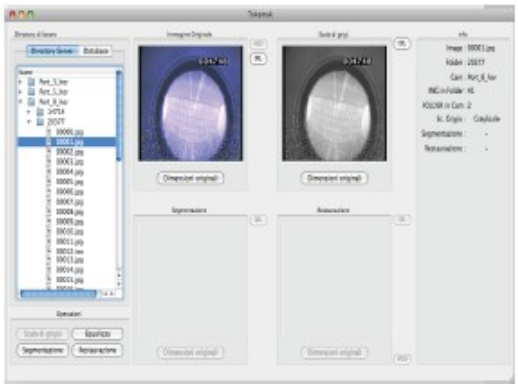


Figure 4: Gray Scale Image of a selected FTU Software Image

It is possible to press others utility key such as the "Original Dimension" in order to analyze source images Fig. 5.

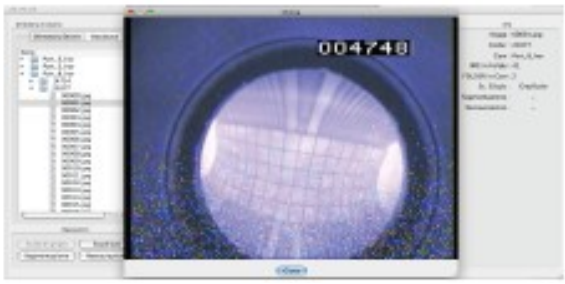


Figure 5: Original dimension of image

It is possible to insert in or delete from the database an image by pressing "Add" or "Del" keys. Obviously, in order to avoid error, the "Add" key will be active only if the image is not stored in the database, likewise the "Del" key will be active only if the image is already in the database.

In Fig. 4-5, the first box on the right (Fig. 6) gives us the information about the selected image.

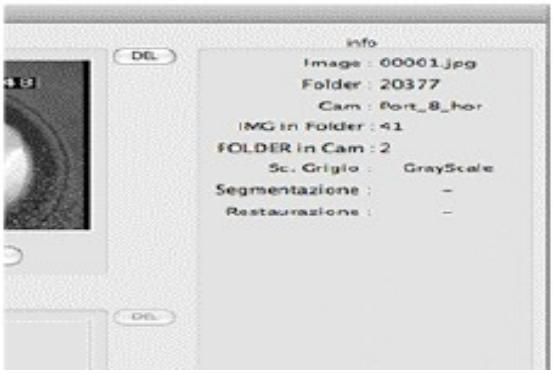


Figure 6: Windows of selected image informations

IV. Image Analysis

By using FTUsoftware we can extract some information about FTU experiments that are very interesting in the application [2]. For examples we can equalize the histogram with:

$$s_k = T(r_k) = \sum_{j=0}^{k} p_r(r_j) = \sum_{j=0}^{k} \frac{n_j}{n}, \quad 0 \le r_k \le 1 \ \text{ e } \ k = 0,1,\ldots,L-1 \qquad (1)$$

where r_k is a fixed gray level and p_r is an uniform probability distribution, n the total number of elements and n_j is the number of elements at r_j level. The results is showed in Fig. 7.

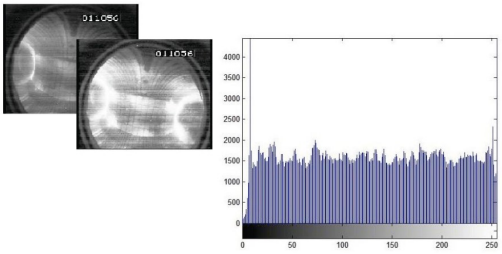


Figure 7: Example of Histogram equalization

Others image processing operation are implemented such as Image Restoration or Image Segmentation. By using average harmonic filter

$$\hat{f}(x,y) = \frac{mn}{\sum_{(s,t)\in S_{xy}} \frac{1}{g(s,t)}} \quad (2)$$

where S_{xy} is a fix rectangular region of m x n dimension and g(x,y) is the related part of original image f(x,y).

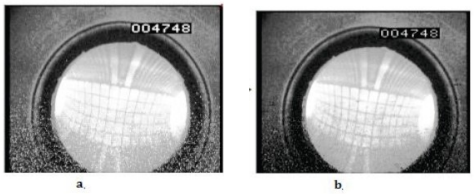


Figure 8: Example of image's Restauration

The result of a Edge Detection Filters and Region Growing is showed in Fig. 9 e Fig.10.

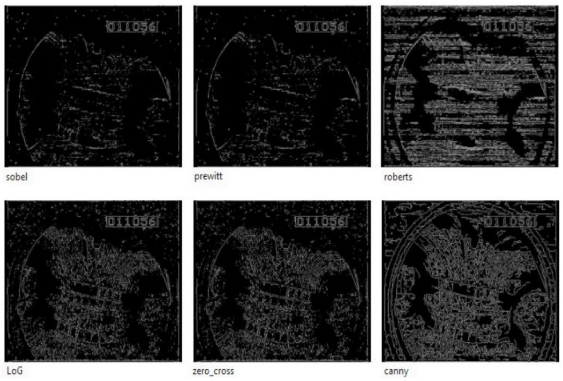


Figure 9: Example of Edge Detection Filter

All figures show a possible use of image analysis tools for monitoring and analysing physical experiment. Particularly in Fig.10 can be identified clearly the behaviour of the plasma.

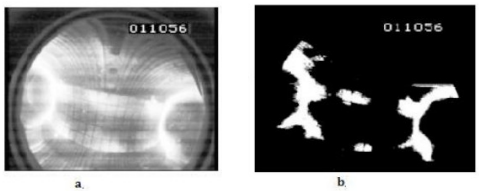


Figure 10: Example of Region Growing

V. FTU-SOFTWARE STATISTICS

The software is used in ENEA in order to analyze the data of FTU experiment. In this section we report some statistics given by the image analysis the experimental data selected are in Table I.

Table 1: Selected Ftu data

Folder	Number of images	Disk usage
Port_3_hor	16.971.745	17GB
Port_5_hor	29.400.484	29GB
Port_8_hor	682.04	667MB
TOTAL	47.054.275	46GB

Table 2: Histogram analysis

Folder	**NG**	**G**	**Eq**
Port_3_hor	78%	10%	12%
Port_5_hor	35%	35%	30%
Port_8_hor	8%	41%	51%

Case 1. Histogram analysis. FTUsoftware automatically detect the images to be equalized by the following heuristic algorithm:

1. *Compute the maximum M of histogram function h(x) for x in [T1, T2] threshold bounds.*
2. *if M is*
   $M>k$ or $M>=h_1$ and $M<h_2$
   with k, h_1, h_2 experimental parameters equalize the image
3. *if $M<K/4$ delete image*
4. *otherwise image is good.*

Table II report the images for which we are not able to analyze (NG), the images with good gray level distribution (G) and those to be equalized (Eq). We set K=4000, h_1 =2000 and h_1 =3000.

Case 2. Image Restoration. In this subsection only few numerical experiments on image restoration are carried out. We compute Signal to Noise ratio (SNR):

$$SNR_{ms} = \frac{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left[\hat{f}(x,y)\right]^2}{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left[\hat{f}(x,y) - f(x,y)\right]^2} \quad (3)$$

between selected gray scale f(x,y) and median filtering restored images.

$\hat{f}(x,y)^*$	$f(x,y)^{**}$	SNR_{ms}	$\hat{f}(x,y)^*$	$f(x,y)^{**}$	SNR_{ms}
gray_00001	rest_00001	0.0382	gray_00020	rest_00020	3.8312
gray_00002	rest_00002	0.2340	gray_00021	rest_00021	3.9769
gray_00003	rest_00003	0.5435	gray_00022	rest_00022	4.1054
gray_00004	rest_00004	0.5248	gray_00023	rest_00023	4.0319
gray_00005	rest_00005	1.0218	gray_00024	rest_00024	4.0457
gray_00006	rest_00006	1.2112	gray_00025	rest_00025	4.9981
gray_00007	rest_00007	1.0020	gray_00026	rest_00026	4.2386
gray_00008	rest_00008	1.1327	gray_00027	rest_00027	6.3468
gray_00009	rest_00009	1.1845	gray_00028	rest_00028	5.8252
gray_00010	rest_00010	2.8529	gray_00029	rest_00029	5.7690
gray_00011	rest_00011	2.6735	gray_00030	rest_00030	7.0154
gray_00012	rest_00012	3.1391	gray_00031	rest_00031	6.3867
gray_00013	rest_00013	2.5010	gray_00032	rest_00032	2.3753

Table 3: Table of SNR results

The above table is loaded into a special database that will be used by operators to compare the noise level of images acquired by FTU-Tokamak.

VI. Conclusions

This paper presents, within the ENEA CRESCO Project, a software facility "FTUsoftware" aimed at analysing Enea Tokamak images. The target of the work is to use FTUsoftware in order to automate the processing of Tokamak images; indeed, we ported a number of matlab applications which analyse and elaborate the images coming from the Tokamak. In details, the applications allow image quality improvement (noise reduction, contrast enhancement, distortions correction), automatic classification by pattern recognition algorithms and brightness analysis, used to detect images with a characteristic feature (quite recurrent in the plasma) in the brightness distribution.

The system is integrated on ENEA-Grid oriented in order to build a specific tool for processing and analyzing the images. Data are stored with MySQL in a suitable database customize to FTU, on mathematical and statistical models both based in order to correlate the images with plasma signals and to search the image information retrieval."FTUsoftware" is integrated in ENEA-GRID infrastructure and it is capable to link Numeric Mathematics with Statistical tools. FTUsoftware is an efficient, extremely robust and portable software; it allows to draw out and to detect/analyse the images within a database.

Acknowledgment

This work is parts of the CRESCO Project funded by the ENEA (www.cresco.enea.it) [5].

References

[1] Gonzalez , Digital Image Processing Using Matlab.

[2] S. Cuomo, La Trasformata di Wavelet Parallela 2D, di un'immagine digitale, CPS-CNR Tech. Rep. TR-2001 - 04, Febbraio 2001

[3] M. Chinnici, S. Migliori, R. De Angelis, S. Borioni, S. Pierattini, Image analysis of a nuclear plasma: Frascati Tokamak Upgrade using IDL and ENEA-GRID technologies – EScience Conference, Nsaples 2008

[4] http://www.cresco.enea.it/LA1/cresco_ sp12_graf3d/

[5] http://tangentsoft.net/mysql++/

An image moment based approach for visual predictive control

C. Copot, A. Burlacu, *Member, IEEE* and C. Lazar, *Member, IEEE*

***Abstract*— In this paper, a model predictive controller based on image moments derived from points is proposed for controlling a manipulator robot. An image-based visual control architecture for a 6 degree of freedom manipulator robot with an eye-in-hand camera configuration is considered. The predictive control strategy was implemented and validated. A comparison with classical proportional visual control architecture was considered, revealing better performances.**

I. INTRODUCTION

IMAGE based visual servoing (IBVS) approach uses visual information acquired from a sensor to control a robot from an initial to a desired configuration. Visual features used in IBVS to compute the control action are defined in image. The relation between the image plane and the Cartesian space is done using a nonlinear operator named usually interaction matrix. Taking into account that the main advantage of image moments is that it allows the decoupling of the interaction matrix and thus it allows the decoupling of the camera screw velocity components, image moments derived from point features are used as visual features to control a manipulator robot. Considering the complexity of visual servoing tasks, in order to design image based controllers, advanced control techniques are needed.

Visual predictive control algorithms have already been reported in literature. In the last years, the stability problem for the receding horizon control grows an important issue for many researchers. Recently, in [1] the IBVS tasks are solved using a predictive approach. To predict the values of the point features over a prediction horizon, two types of model are considered. The first one used a nonlinear global model for the image prediction, while the second one used a local model based on interaction matrix to predict the evolution of the point features regarding the camera velocity.

Based on a linear approximation around the working points, a multivariable controller was used to control a fast visual servo system of a two links arm [2]. The robot is modelled taking into account the flexibility in the link and the dynamics of the velocity controlled actuators. The proposed model was used to design a GPC controller, and also an advanced controller in the frequency domain (H_∞ controller). Indeed in [3], Hashimoto assumes that the robot dynamic is modelled as a perfect Cartesian motion device and uses point coordinates, in order to control a 6 d.o.f robot with a LQ state feedback controller. For a high speed visual servoing of a manipulator, an ARIMAX multivariable model which allows to implement a GPC controller is described in [4], and the robot is modelled as a Virtual Cartesian Motion Device (VCMD) [5].

Another model of a visual servo system is given in [6] and it employs camera position controller with a robust disturbance observer in the joint space. In this way each joint axis is decoupled and the inner loop can be expressed in the frequency region below the cut-off frequency of the robust disturbance observers as a diagonal transfer matrix. In [7] the IBVS structures based on nonlinear model predictive control are used for controlling manipulators with catadioptric cameras. Also, for a 3D visual servoing task, a nonlinear predictive approach is presented in [8]. Until now, all the visual predictive control approaches reported in literature do not use image moment prediction to control the robot movements.

In this paper, a visual predictive control strategy is developed based on a new approach that uses image moments for the first time. Using [9] and [10], an image-based predictive control architecture is developed and a proper plant model for computing future prediction of image moments derived from point features is proposed. Taking into account the discrete model of an eye-in-hand configuration (for a manipulator robot) and using the relation between the time derivative of visual feature and the camera velocity through interaction matrix, one step ahead prediction model of the visual servoing system is obtained. Based on a recursive technique of the one step ahead prediction model, new predictors over a horizon prediction is computed. Using the proposed predictor, an image based predictive controller was designed to control a six d.o.f. manipulator robot. The image moments based predictive control algorithm was tested and validated using a Matlab implementation. A comparison with a proportional controller was conducted and experimental results were revealed.

The structure of this paper is given as follows: in Section II is presented the analytical method of computing image moments and interaction matrix related to it. Section III is devoted to the predictive control algorithm, the experimental results are presented in Section IV and the conclusions are detailed in Section V.

Cosmin Copot is with the Department of Automated Control and Applied Informatics, "Gheorghe Asachi" Technical University of Iasi, Romania. (corresponding author, e-mail: ccopot@ac.tuiasi.ro).

Adrian Burlacu is with the Department of Automated Control and Applied Informatics, "Gheorghe Asachi" Technical University of Iasi, Romania. (e-mail: aburlacu@ac.tuiasi.ro).

Corneliu Lazar is with the Department of Automated Control and Applied Informatics, "Gheorghe Asachi" Technical University of Iasi, Romania. (e-mail: clazar@ac.tuiasi.ro).

II. Interaction Matrix of Image Moments

In general, moment functions are used in statistics and in mechanics. The first application of moments in image analysis was developed by Hu [11] in '60s. In computer vision, moments are commonly used for pattern recognition applications, being recently introduced in visual servoing application by Chaumette [12]. It is assumed that an image function $V(x,y)$ is represented as a density distribution function of a 2D random variable and an object in the image V is described by non-zero pixel values of coordinates (x,y) with $V(x,y)=1$. The object consists of a discrete set of n points and, in this case, image moments m_{ij} of order $(i+j)$ are defined as:

$$m_{ij}=\sum_{k=1}^{n}x_k^i y_k^j \tag{1}$$

and the centered moments μ_{ij} of order $(i+j)$ are given by:

$$\mu_{ij}=\sum_{k=1}^{n}(x_k-x_g)^i(y_k-y_g)^j\ , \tag{2}$$

where $x_g=m_{10}/a$, $y_g=m_{01}/a$ represent the coordinates of gravity center, and $a=m_{00}=n$ is the object area.

Let $v_c=[v,\omega]$ be the camera velocity in an eye-in-hand configuration. The vector v_c is composed from $v=[v_x,v_y,v_z]$ and $\omega=[\omega_x,\omega_y,\omega_z]$ the linear, respectively angular components of the camera velocity.

The central moments of order two $\mu_{20},\mu_{02},\mu_{11}$ are known as inertia moments and, as in [12], they are used to compute the orientation angle α of an object:

$$\alpha=\frac{1}{2}\arctan\left(\frac{2\mu_{11}}{\mu_{20}-\mu_{02}}\right) \tag{3}$$

which is used to control the angular velocity ω_z.

It is well known that the centered moments are invariant to 2D translational motion. In general, image moments that are invariant to 2D translations, 2D rotation and to scale have a polynomial form, more details can be found in [13]. In [14], Tahri develops a new type of image moments that will be used as visual features to control the angular velocities ω_x and ω_y:

$$\tau=\frac{I_{n1}}{I_{n3}},\quad \xi=\frac{I_{n2}}{I_{n3}}, \tag{4}$$

where:

$$\begin{aligned} I_{n1}&=(\mu_{50}+2\mu_{32}+\mu_{14})^2+(\mu_{05}+2\mu_{23}+\mu_{41})^2\\ I_{n2}&=(\mu_{50}-2\mu_{32}-3\mu_{14})^2+(\mu_{05}-2\mu_{23}-3\mu_{41})^2\\ I_{n3}&=(\mu_{50}-10\mu_{32}+5\mu_{14})^2+(\mu_{05}-10\mu_{23}+5\mu_{41})^2 \end{aligned} \tag{5}$$

Considering a visual servoing application for grasping a fixed object using a 6 d.o.f. robot manipulator and taking into account the disadvantage of classical point features, a set of image moments $f_m=(x_n,y_n,a_n,\tau,\xi,\alpha)$ are used to derive the image-based control law. The first three components of moment based features vector f_m are generally used to control the linear velocities v_x,v_y and v_z:

$$x_n=a_n x_g,\quad y_n=a_n y_g,\quad a_n=Z^*\sqrt{\frac{a^*}{a}}, \tag{6}$$

where Z^* represents the desired depth between the desired configuration and the camera, and a^* is the desired object area. For a discrete object, the area a represents the number of points which can not be used as visual feature and, thus, in [14] Tahri proposes to replace the area value by:

$$a=\mu_{20}+\mu_{02}\,. \tag{7}$$

To apply the image moments in visual servoing applications, first, the interaction matrix related to image moments must be computed. In the following, the algorithm for computing the interaction matrix is presented.

Considering $\mathbf{x}=(x,y)$ a point feature, the interaction matrix related to it is computed using:

$$L_{\mathbf{x}}=\begin{bmatrix} -\frac{1}{Z} & 0 & \frac{x}{Z} & \frac{xy}{\delta} & -\frac{\delta+y^2}{\delta} & y\\ 0 & -\frac{1}{Z} & \frac{y}{Z} & \frac{\delta+y^2}{\delta} & -\frac{xy}{\delta} & -x \end{bmatrix}, \tag{8}$$

where Z represents the point depth and δ the focal length. Assuming the hypothesis that the object is planar, its equation expressed in the camera frame is given by:

$$\frac{1}{Z}=Ax+By+C\ , \tag{9}$$

where Z is the depth of any object point [12]. Taking into account that the object is static, the time variation of a visual features $\mathbf{x}$ can be computed with respect to camera velocity v_c:

$$\dot{\mathbf{x}}=L_{\mathbf{x}}v_c\ , \tag{10}$$

where $L_{\mathbf{x}}$ is the interaction matrix related to $\mathbf{x}$ computed using (8).

Combining (9) and (8) together with (10), the velocity of any object point (x_k,y_k) is given by:

$$\begin{cases} \dot{x}_k = -(Ax_k + By_k + C)v_x + x_k(Ax_k + By_k + C)v_z \\ \quad + x_k y_k \omega_x - (1 + x_k{}^2)\omega_y + y_k \omega_z \\ \dot{y}_k = -(Ax_k + By_k + C)v_y + y_k(Ax_k + By_k + C)v_z \\ \quad + (1 + y_k{}^2)\omega_x - x_k y_k \omega_y - x_k \omega_z \end{cases} \quad (11)$$

Using (11) together with (2) and after tedious development, the interaction matrix of the centered moments μ_{ij} is obtained [14]:

$$L_{\mu_{ij}} = [\mu_{v_x} \quad \mu_{v_y} \quad \mu_{v_z} \quad \mu_{\omega_x} \quad \mu_{\omega_y} \quad \mu_{\omega_z}], \quad (12)$$

where:

$$\begin{cases} \mu_{v_x} = -iA\mu_{ij} - iB\mu_{i-1,j+1};\ \mu_{v_y} = -jA\mu_{i+1,j-1} - jB\mu_{ij} \\ \mu_{v_z} = -A\mu_{wy} + B\mu_{wx} + (i+j)C\mu_{ij} \\ \mu_{\omega_x} = (i+j)\mu_{i,j+1} + ix_g\mu_{i-1,j+1} + (i+2j)y_g\mu_{ij} \\ \quad - in_{11}\mu_{i-1,j} - jn_{02}\mu_{i,j-1} \\ \mu_{\omega_y} = -(i+j)\mu_{i+1,j} - (2i+j)x_g\mu_{i,j} - jy_g\mu_{i+1,j-1} \\ \quad + in_{20}\mu_{i-1,j} + jn_{11}\mu_{i,j-1} \\ \mu_{\omega_z} = i\mu_{i-1,j+1} - j\mu_{i+1,j-1};\quad n_{ij} = \mu_{ij}/a \end{cases} \quad (13)$$

Having the image center coordinates (u_0, v_0) and the intrinsic camera parameter (p_x, p_y) the point feature **x** can be expressed in pixels using:

$$f_p = (u_i, v_i) \text{ with } u_i = \frac{x_i}{z_i} p_x + u_0, \quad v_i = \frac{y_i}{z_i} p_y + v_0. \quad (14)$$

For a set of discrete image moments $f_m = (x_n, y_n, a_n, \tau, \xi, \alpha)$ derived from point features, the interaction matrix is given by:

$$L_{f_m} = \begin{bmatrix} -1 & 0 & 0 & a_n e_{11} & -a_n(1+e_{12}) & y_n \\ 0 & -1 & 0 & a_n(1+e_{21}) & -a_n e_{11} & -x_n \\ 0 & 0 & -1 & -e_{31} & e_{32} & 0 \\ 0 & 0 & 0 & \tau_{\omega x} & \tau_{\omega y} & 0 \\ 0 & 0 & 0 & \xi_{\omega x} & \xi_{\omega y} & 0 \\ 0 & 0 & 0 & \alpha_{\omega x} & \alpha_{\omega y} & -1 \end{bmatrix}. \quad (15)$$

The parameters from (15), $e_{11}, e_{12}, e_{21}, e_{31}, e_{32}$ are related to linear velocities, while $\tau_{\omega x}, \tau_{\omega y}, \xi_{\omega x}, \xi_{\omega y}, \alpha_{\omega x}, \alpha_{\omega y}$ are related to angular velocities, and their analytical form can be found in [14].

III. Predictive Control

A. Predictor Model

An image based architecture that uses image moments to control a 6 d.o.f. manipulator robot is presented in Fig. 1. As set point in the control structure, a set of image moments f_m^* computed for the desired configuration of the point features is considered. The image moments are computed from the classical point features f_p^* using transformation $\mathbb{M}$ that is based on equations (1)-(7). The predictive controller for IBVS is developed to control a manipulator with an eye-in-hand configuration in order to grasp a fixed object.

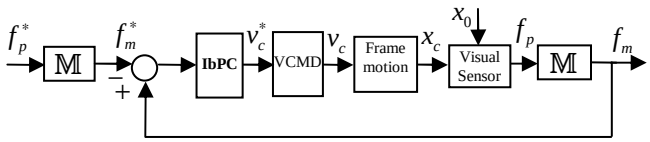


Fig. 1 Image based visual servoing architecture

The manipulator can be modelled as a Virtual Cartesian Motion Device (VCMD), which has as input the reference of camera velocity v_c^* and as output the camera velocity v_c. If it is considered that the manipulator robot is a decoupled linear diagonal plant, then, the camera velocity can be computed in the camera coordinates using:

$$v_c(s) = G(s)v_c^*(s). \quad (16)$$

One way is to assume that each joint axis is decoupled under cut-off frequency [6]. Assuming that the velocity controller is designed as a diagonal matrix $\mathbf{k}_v = diag\{k_v, \ldots, k_v\}$, the following linear discrete model of the VCMD system is obtained:

$$G(z) = (1 - z^{-1})\mathbb{Z}(G(s)/s); \quad G(s) = \frac{k_v}{s + k_v} I_6, \quad (17)$$

where k_v is a proportional gain controller and $\mathbb{Z}$ represents the *z* transform.

The camera velocity is used in "Frame Motion" to compute the new position and orientation of the camera, Fig. 2. It is assumed that the camera starting pose is known. The output of "Frame motion" represents the homogeneous transformation $T_c^b(k)$ at current discrete time *k*.

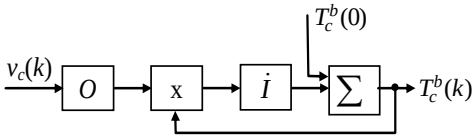


Fig. 2. Frame motion block

The camera velocity v_c is processed by an operator *O* that gives a homogeneous matrix. After a multiplication and an integral action, the result is added with the starting camera pose $T_c^b(0)$ and, thus, $T_c^b(k)$ is obtained [9]. Assuming that, the starting object features position related to the robot base x_0 are known and using the camera pose x_c stored in $T_c^b(k)$, the current features coordinates are obtained.

Using the point features f_p and the transformation $\mathbb{M}$, the new image moments f_m derived from current point features are computed. The one-step-ahead predictor model for a visual servoing system is presented in Fig. 3.

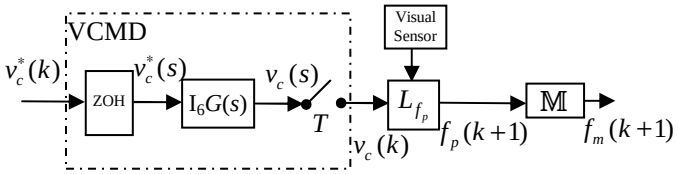


Fig 3. One-step-ahead predictor

Using the derivative approximation of the features vector f_p from (10), the following equation is obtained:

$$\frac{f_p(k+1)-f_p(k)}{T}=L_{f_p}(k)v_c(k)\,. \tag{18}$$

Substituting (16) in (18) and taking into account the discrete model of the VCMD, one-step ahead prediction of the classical point features f_p is obtained:

$$f_p(k+1)=f_p(k)+TL_{f_p}(k)G(z^{-1})v_c^*(k)\,. \tag{19}$$

Next it will be presented how the h_p predictors are obtained using a recursive of the one-step ahead prediction model:

$$\begin{aligned}
&f_p(k+2)=f_p(k+1)+TL_{f_p}(k+1)G(z^{-1})v_c^*(k+1)\\
&\text{.....}\\
&f_p(k+i)=f_p(k+i-1)+TL_{f_p}(k+i-1)G(z^{-1})v_c^*(k+i-1)\\
&\text{.....}\\
&f_p(k+h_p)=f_p(k+h_p-1)+\\
&\qquad TL_{f_p}(k+h_p-1)G(z^{-1})v_c^*(k+h_p-1)
\end{aligned} \tag{20}$$

As it can be observed in (20), for the computation of a general predictor $f_p(k+i)$ it is necessary to know the visual features at moment $(k+i-1)$, the previous command $v_c^*(k+i-1)$ and the interaction matrix $L_{f_p}(k+i-1)$ related to previous features. Using the point features $f_p(k+i)$ and the transformation $\mathbb{M}$, the predicted image moments derived from points are obtained:

$$f_m(k+i)=\mathbb{M}(f_p(k+i))\,. \tag{21}$$

B. Control Algorithm

The principle of an image-based control architecture is to minimize the error $e(t)$ defined by:

$$e(t)=f_m(t)-f_m^*\,, \tag{22}$$

where $f_m(t)$ is the current set of visual features, while f_m^* represents the desired configuration of visual features.

The image moment control error expressed in Cartesian space is defined by:

$$e_c(k+i)=L_{f_m}^{+}\left(f_m^*-f_m(k+i)\right),\ i=\overline{1,h_p}\,, \tag{23}$$

where $L_{f_m}^{+}=\left(L_{f_m}^T L_{f_m}\right)^{-1}L_{f_m}^T$ is a pseudo-inverse of the interaction matrix L_{f_m}. For the experiments, the following approximation of the interaction matrix is used:

$$\widehat{LL}=1/2\left(L_{f_m}+L_{f_m}^*\right), \tag{24}$$

which represents an average between L_{f_m} and $L_{f_m}^*$, ensuring better performances. The features vector f_m related to the predicted image moments represents the input of a minimization criterion defined as a cost function which has a quadratic form of input features vector and image error expressed in Cartesian space:

$$J=\frac{1}{2}\sum_{i=1}^{h_p}e_c^T(k+i)Qe_c(k+i)+\frac{1}{2}\sum_{i=0}^{h_c-1}v_c^{*T}(k+i)Rv_c^*(k+i)\,, \tag{25}$$

where h_c is the control horizon, Q and R are positive matrices designed to weight the features error e_c and the camera velocity. Generally the weighting matrices are equal with the identity matrix. In order to ensure an exponential decrease it is proposed to compute the weighting matrices Q and R using:

$$Q=e^{h_p-i}I_6\,,\ R=e^{h_c-i-1}I_6\,. \tag{26}$$

By minimizing the cost function (25), v_c^*, which represents the reference for the VCMD, is obtained. Based on the horizon prediction h_p, the cost function computes the velocity vectors from moment $(k+1)$ to $(k+h_p)$. The cost function is minimized using the Matlab function *fmincon*. In order to guarantee a valid solution of the image based visual servoing predictive strategy, it is necessary to introduce the following constrain due to the limits of the image expressed in pixels:

$$\left(u_i(k),v_i(k)\right)\in\left[u_{\min},u_{\max};v_{\min},v_{\max}\right]. \tag{27}$$

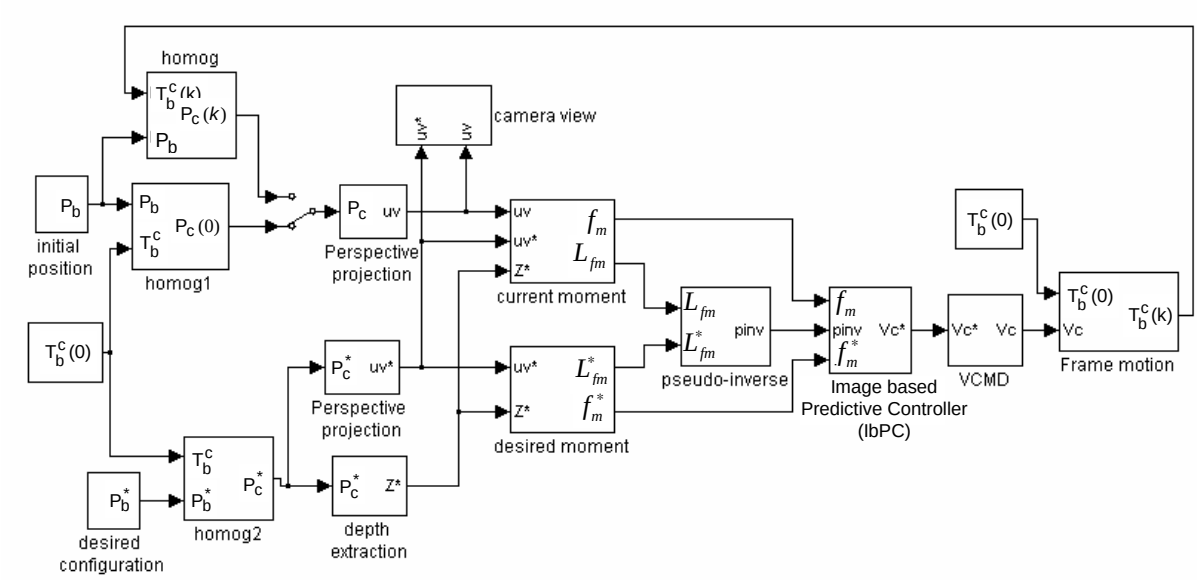


Fig. 4 Visual control architecture

IV. Experimental Results

In this section, the experimental results of the image moments predictive strategy are presented. The testing and validation of image moments based approach for visual predictive control was done by using the simulator from Fig. 4. This block diagram has as input two sets of points in Cartesian space, points that describe a planar object in the desired P_b^* and initial P_b pose towards the camera frame.

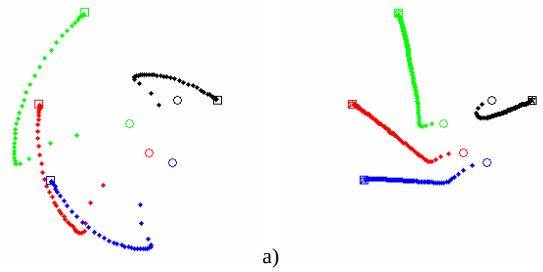
Knowing the camera position related to the robot frame $T_b^c(0)$, the desired points related to the robot P_b^* and, using a homogeneous transformation implemented with 'homog2' block, the desired points position P_c^* related to the camera frame can be detected. In a similar mode, the initial and the current position of the points are transposed from the robot frame to the camera frame using a homogeneous transformation [15]. Using the 'Perspective projection' blocks from Fig. 4, the points given in Cartesian space are transposed to the image plane. The visualization of the initial, current and desired 2D points during the simulation is available by using 'Camera view' block. The image moments selected to control a manipulator robot, $f_m = (x_n, y_n, a_n, \tau, \xi, \alpha)$, are derived from points and were computed using 'current moment' and 'desired moment' blocks. Image moments are calculated based on (3), (4) and (6). In the same time, the interaction matrices L_{f_m} and $L_{f_m}^*$ are calculated with (15) and represent the input of the 'pseudo-inverse' block. The 'IbPC' block implements the minimization of (25) and has as output the reference v_c^* of the VCMD block.

To evaluate the performances of the image based predictive approach presented in Fig. 3, the structure depicted in Fig. 4 was implemented in Matlab. The planar object is consisting of 4 points given in Cartesian space. The task configuration of the servoing system depicted in Fig. 5a and 5b was considered to run the experiments, where circles illustrate the initial position, while the desired configuration of the points is marked with squares. An image-based predictive controller was designed to minimize the cost function (25) at each iteration *k*. To compute the prediction of image moments derived from point features, equation (21) is implemented. Considering the parameters for the IbPC having the values $h_p = 3$ and $h_c = 1$, the following experimental results were obtained. Fig. 5a presents the point features trajectory when the tuned image moments based predictive control is considered. The evolution of the linear and angular components of the camera velocity vector, when using the predictive approach, are illustrated in Fig. 6.

To implement an image-based proportional controller, the following control law was used:

$$v_c = -\lambda \widehat{LL}(f_m - f_m^*) \tag{28}$$

where λ is a positive gain tuning which was set to 0.125. Fig. 5b illustrates the trajectory of point features when a proportional controller is used.



a) b)

Fig. 5 Task configuration: a)point features trajectory using IbP controller; b)point features trajectory using a proportional controller

If a proportional controller is used to control a manipulator robot from an initial configuration to a desired one Fig. 5b, the camera velocity is depicted in Fig. 7. Analyzing the results from Fig. 6 and 7 an improvement of the camera velocity can be observed if the predictive control algorithm is used to control the robot. Experimental results prove the validity of the proposed predictive strategy and show a greater convergence for the predictive approach compared to the classical proportional control law.

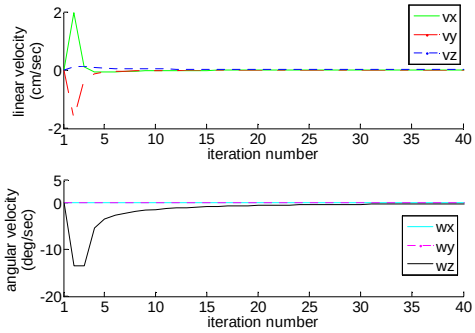


Fig. 6 Camera velocity: the linear and angular camera velocity using predictive controller

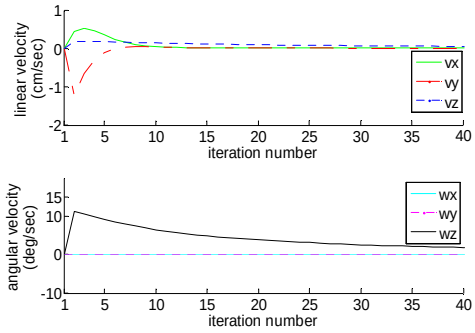
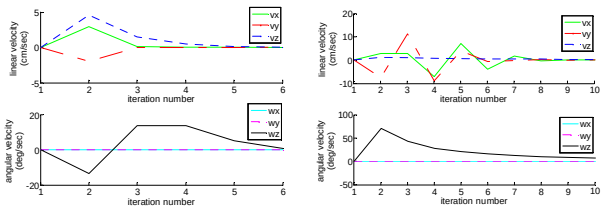


Fig. 7 Camera velocity: the linear and angular camera velocity using a proportional controller

Another experiment was considered in order to analyze the robustness of the predictive control algorithm. This experiment shows that using an image-based predictive controller derived from point features, it is possible to reach the desired configuration in less than 10 iterations (Fig. 8a), while in case of a classical proportional controller, if the tuning parameter is increased, the linear component of the camera velocity has an oscillatory behavior (Fig. 8b).



a) b)

Fig. 8 Camera velocity: a) camera velocity components when an IbPC is used b) the linear and angular camera velocity using a proportional controller

V. Conclusion

In this paper, a new approach for visual predictive control is presented. This new predictive strategy uses image moments derived from point features to control the end-effector of a manipulator robot towards complition of a visual servoing task. A visual servoing simulation architecture was used for testing and validating the image moments predictive strategy using a planar object consisting of 4 Cartesian points. The comparison of the new image moments predictive approach and a classical proportional controller is presented. As the experimental results revealed, the image moments visual predictive control improves the camera velocity in comparison with a classical proportional controller, by reducing the number of iterations needed to reach the desired configuration of the visual features.

References

[1] G. Allibert, E. Courtial and F. Chaumette, *Visual servoing via nonlinear predictive control*, Springer-Verlag, Berlin 2010, pp. 375-393.

[2] L. Cuvillon, L. Edouard, J. Gangloff and M. de Mathelin, "GPC versus H∞ Control for Fast Visual Servoing of a Medical Manipulator Including Flexibilities", *Proc. IEEE Int. Conf. on Robotics and Automation*, Barcelona, 2005, pp. 4044-4049

[3] K. Hashimoto, T. Ebine and H. Kimura, "Visual servoing with hand-eye manipulator-optimal control approach", *IEEE Trans. on Robotics and Automation*, vol. 12, 1996, pp. 766-774.

[4] J. Gangloff and M. de Mathelin, "High Speed Visual Servoing of a 6 DOF Manipulator Using Multivariable Predictive Control", *Advances Robotics*, vol. 21, 2003, pp. 993-1021.

[5] Hutchinson S., Hager C., Corke P., "A tutorial on Visual Servo Control" *IEEE Trans. On Robotics and Automation*, vol. 12, 1996, pp. 651-670.

[6] H. Fujimoto, "Visual Servoing of 6 Dof Manipulator by Multirate Control with Depth Identification", *Proc. of 42nd IEEE Conference on Decision and Control*, Hawaii, December 2003, pp. 5408- 5413.

[7] G. Allibert, E. Courtial and Y. Toure, "Visual Predictive Control for Manipulators with Catadioptric Camera", *IEEE Int. Conf. on Robotics and Automation*, Pasadena, 2008, pp. 510-515.

[8] T. Murao, T. Yamada and M. Fujita, "Predictive visual feedback control with eye-in-hand system via stabilizing receding horizon approach", *Proc. of 45th IEEE CDC*, December 2006, pp. 1756-1763.

[9] C. Lazar and A. Burlacu, "Predictive control strategy for image based visual servoing of robot manipulators", *Proc. of 9th International Conference on Automation and Information*, Bucharest 2008, pp 18, 91-97.

[10] C. Lazar and A. Burlacu, "Visual Servoing of Robot Manipulators using Model-based Predictive Control", *Proc. of 7th IEEE International Conference on Industrial Informatics*, 24-26 June, Cardiff, pp. 690-695.

[11] M.K. Hu, "Visual pattern recognition by moment invariants", *IRE. Trans. Inf. Theory*, vol. 8, no. 1, 1962, pp. 179–187.

[12] F. Chaumette, "Image moments: a general and useful set of features for visual servoing", *IEEE Trans. on Robotics*, vol. 20, no. 4, August 2004, pp. 713-723.

[13] R. Mukundan and K.R. Ramakrishnan, *Moment Functions in Image Analysis Theory and Application*. World Scientific Publishing Co.Pte.Ltd, 1998.

[14] O. Tahri and F. Chaumette, "Point-based and region-based image moments for visual servoing of planar objects", *IEEE Trans. on Robotics*, vol. 21, no. 6, December 2005, pp. 1116-1127.

[15] C. Copot, A. Burlacu and C. Lazar, "Visual Control Architecture of Servoing Systems Based on Image Moments", *12th International Conference on Optimization of Electrical and Electronic Equipment*, Brasov 2010.

Absolute stability conditions for some scalar nonlinear time-delay systems with monotone increasing nonlinearity

Daniela Danciu

***Abstract*— This paper deals with the analysis of the absolute stability for a class of scalar nonlinear time-delay systems having monotone increasing nonlinearities. The approach is based on frequency domain inequalities of Popov type for time-delay systems in the critical case of the transfer function with a simple zero pole. In order to solve the *minmax* problem which arises from the frequency domain inequality, the analytical analysis was completed by numerical computations using MATLAB software package. We have obtained a *time-delay dependent absolute stability condition*.**

I. STATE OF THE ART AND PROBLEM STATEMENT

A. Absolute stability and frequency domain inequalities

The property of absolute stability had been motivated by Letov [1], [2] by the rather poor information about nonlinearity and, as Răsvan remarks in [3], "in a more contemporary statement this is nothing else but robust stability with respect to some kind of nonlinear function uncertainty".

Absolute stability refers to the global asymptotic stability of the zero equilibrium of the nonlinear system

$$\dot{x}(t) = Ax - b\varphi(c^*x) \tag{1}$$

having sector restricted nonlinearities of the form (see Fig. 2)

$$0 \le \underline{\varphi} \le \frac{\varphi(\sigma)}{\sigma} \le \overline{\varphi} \le +\infty \ , \ \varphi(0) = 0 \tag{2}$$

the property of the equilibrium being valid for all the linear and nonlinear functions verifying (1).

For the absolute stability problem, the system under the analysis (1) might be written as a "negative" feedback connection (Fig. 1) of the linear block L described by

$$\begin{aligned} \dot{x}(t) &= Ax + b\mu_1 \\ \sigma_1 &= c^*x \end{aligned} \tag{3}$$

with the nonlinear block, subject to (2) and described by

$$\sigma_2 = \varphi(\mu_2) \tag{4}$$

the interconnection rules being

$$\mu_2 = \sigma_1 \ , \ \mu_1 = -\sigma_2. \tag{5}$$

This work has been supported by the Research Project CNCSIS ID-95 of the Romanian Council for University Research.

Daniela Danciu is with Department of Automatic Control, University of Craiova, A.I. Cuza, 13, Craiova, RO-200585, Romania. {daniela}@automation.ucv.ro

The above remark will be useful in the next sections. In the sequel we shall use the following notations regarding the signals in Fig. 1: $\mu := \mu_1 = -\sigma_2$ and $\sigma := \sigma_1 = \mu_2$.

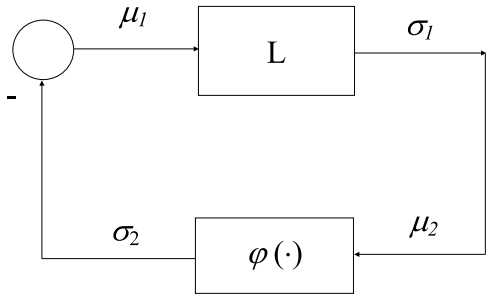


Fig. 1. Absolute stability feedback structure.

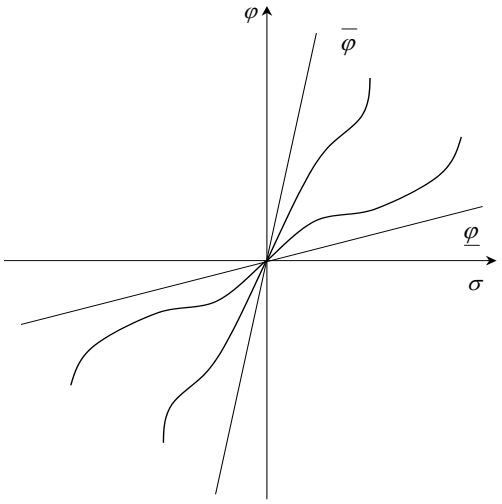


Fig. 2. Sector restricted nonlinearities

Concerning the system with the structure in Fig. 1 we recall here the so-called Aizerman problem. Let L be a linear controlled system. If $\varphi(\sigma)$ is a linear function $\varphi(\sigma) = h\sigma$, any stability criterion would give a sector $h \in (\underline{\varphi}_H, \overline{\varphi}_H)$ called the Hurwitz sector which, corresponding to the necessary and sufficiently stability conditions, is thus maximal. If, on the other hand, one considers nonlinear functions verifying (2), then only sufficient global stability conditions can be obtained (generally speaking) and the resulting maximal sector will be, as a rule, more narrow than the Hurwitz sector. *The comparison of the two sectors is the Aizerman problem*: the closer they are, the less is the "degree of conservatism"

obtained via the available sufficient conditions of absolute stability (among which the frequency domain inequalities together with the Liapunov functions and the Linear Matrix Inequalities associated to them are the less conservative).

According to Lefschetz [4], the development of the absolute stability theory has known two periods: the pre-Popov period and Popov period that started after 1960. Regarding the second period, this is marked by the frequency domain inequalities used for analyzing the stability of nonlinear systems. These were introduced by the Romanian scientist V. M. Popov in his pioneering paper [5] and became known worldwide especially after his seminal paper [6].

For the goal of the paper we shall introduce here the absolute stability result based on frequency domain inequality of Popov-type for time-delay systems in the critical case of a zero root (a straightforward extension of the Theorem 6.1 in [7]).

Theorem 1: Consider the control system described by

$$\begin{gathered}\dot{x}(t)=Ax(t)+\sum_1^r B_k x(t-\tau_k)+b_0\xi(t)+\sum_1^r b_k\xi(t-\tau_k),\\ \dot{\xi}(t)=-\varphi(\sigma(t)),\\ \sigma(t)=c_0^{'}x(t)+\sum_1^r c_k^{'}x(t-\tau_k)+\gamma_0\xi(t)+\sum_1^r\gamma_k\xi(t-\tau_k)\end{gathered} \tag{6}$$

where $\varphi(\sigma)$ is a continuous nonlinearity verifying the conditions:

$$0<\frac{\varphi(\sigma)}{\sigma}<k\leq+\infty\ ,\ \varphi(0)=0. \tag{7}$$

One supposes that the characteristic equation

$$\det\left(sI-A-\sum_1^r B_k e^{-s\tau_k}\right)=0 \tag{8}$$

has all its roots within $\mathbb{C}^-$, and

$$\gamma_0+\sum_1^r\gamma_i-\left(c_0^{'}+\sum_1^r c_j^{'}\right)\left(A+\sum_1^r B_k\right)^{-1}\left(b_0+\sum_1^r b_l\right)>0. \tag{9}$$

If there exists $q\geq 0$ finite such that

$$\frac{1}{k}+\Re e\,(1+\imath\omega q)H(\imath\omega)\geq 0, \tag{10}$$

where

$$\begin{gathered}H(s)=\tfrac{1}{s}\left[\gamma_0+\sum_1^r\gamma_i e^{-s\tau_i}+\left(c_0^{'}+\sum_1^r c_j^{'}e^{-s\tau_j}\right)\cdot\right.\\ \left.\left(sI-A-\sum_1^r B_k e^{-s\tau_k}\right)^{-1}\left(b_0+\sum_1^r b_l e^{-s\tau_l}\right)\right],\end{gathered} \tag{11}$$

then the system (6) is asymptotic stable for every nonlinearity within the class above defined.

Remark 1: The Theorem 1 ensures absolute stability for all nonlinearities which verify (7), yielding the absolute stability sector $(0,k)$ which can be compared with the Hurwitz sector in the Aizerman problem; the fulfilment of the frequency domain inequality (10) will give the "dimension" of the absolute stability sector. The inequality (9) ensures the limit stability property of the linear block in the critical case considered and, is similarly to the condition (12) in Theorem 2 from the subsection *Limit stability property* which will follow. The equation (11) is nothing else that the transfer function of the system (6).

B. The limit stability property

In connection with absolute stability and Aizerman problem is the limit stability property - introduced by Aizerman and Gantmakher [8]. Consider again the systems having the structure in Fig. 1 and suppose that the nonlinear function is known with some uncertainty. As already said, we call absolute stability a robust version of the stability property, i.e. the stability of the zero equilibrium for all nonlinear and linear function within the sector $(0,k)$ and belonging to a certain class of functions. For ensuring the absolute stability property, a necessary and minimal condition would be exponential stability for a single linear function of the sector. In particular, if L defines an exponentially stable linear system, the property (called by V. M. Popov [9] minimal stability) holds for $\varphi(\sigma)\equiv 0$. But if L is in a critical case i.e. it has the spectrum in $\mathbb{C}^-$ as well as on $\imath\mathbb{R}$ a necessary (and minimal) requirement would be exponential stability for $\varphi(\sigma)=\varepsilon\sigma$ with $0<\varepsilon<\varepsilon_0$ and $\varepsilon_0>0$ arbitrarily small. This is called limit stability and is in fact a property of the linear block L; the rigorous definition would be as follows

Definition 1: Let L be a linear dynamical block described by some input/output operator - a convolution (in the time domain) or a transfer function (in the complex domain) - connecting the input μ to the output σ. System L is said to have the limit stability property if it is output stabilizable by the output feedback $\mu=-\varepsilon\sigma$ with $0<\varepsilon<\varepsilon_0$ and $\varepsilon_0>0$ arbitrarily small.

There are known necessary and sufficient conditions for limit stability for linear block L associated to a strictly proper rational transfer function - see [8]. They were extended to "the delay case", i.e. to linear blocks whose transfer functions are strictly proper meromorphic functions defined by a ratio of quasi-polynomials; for the sake of the completeness, we shall give in the sequel the main result for this case [7].

Theorem 2: Consider a linear block with the transfer function $H(s)=N(s)/D(s)$ where $N(s)$ and $D(s)$ are quasi-polynomials, $H(s)$ being strictly proper in the sense that the degree of the principal term of $D(s)$ is larger than that of $N(s)$ and assume $D(s)$ to have at most a finite number of roots on $\imath\mathbb{R}$, the other roots being in $\mathbb{C}^-$. For the limit stability of this system it is necessary and sufficient that the multiplicity of each pole should be at most 2 and the following conditions hold for it

a) for a simple zero pole

$$\Re es\,H(s)\,|_{s=0}>0 \tag{12}$$

b) for a simple non-zero pole, if the Laurent expansion is considered

$$\begin{aligned}\Re e\, H(\imath\omega) &= \frac{e_{-1}}{\omega-\omega_0}+d_0-e_1(\omega-\omega_0)-\\ &-d_2(\omega-\omega_0)^2+\dots\\ \Im m\, H(\imath\omega) &= \frac{-d_{-1}}{\omega-\omega_0}+e_0+d_1(\omega-\omega_0)-\\ &-e_2(\omega-\omega_0)^2+\dots\end{aligned} \tag{13}$$

and the sequence: d_{-1}; $e_{-1}e_0$; $-d_1$; $-e_1e_2$; d_3; ... is associated, either $d_{-1}\neq 0$ or $e_{-1}\neq 0$ and the first non-zero term of the above sequence should be strictly positive;

c) for a double zero root, if the Laurent expansion is considered

$$\begin{aligned}\Re e\, H(\imath\omega) &= \frac{-d_{-2}}{\omega^2}+d_0-d_2\omega^2+\dots\\ \Im m\, H(\imath\omega) &= \frac{-d_{-1}}{\omega}+d_1\omega-d_3\omega^2+\dots\end{aligned} \tag{14}$$

and the sequence $-d_{-1}$; d_1; $-d_3$; ... is associated, then *i*) $d_{-2}>0$ and *ii*) the first non-zero term of the sequence should be strictly negative;

d) for a double non-zero root, if the Laurent expansion is considered

$$\begin{aligned}\Re e\, H(\imath\omega) &= \frac{-d_{-2}}{(\omega-\omega_0)^2}-\frac{e_{-1}}{\omega-\omega_0}+d_0-\\ &-e_1(\omega-\omega_0)-\dots\\ \Im m\, H(\imath\omega) &= \frac{-e_{-2}}{(\omega-\omega_0)^2}-\frac{d_{-1}}{\omega-\omega_0}+e_0+\\ &+d_1(\omega-\omega_0)-\dots\end{aligned} \tag{15}$$

and the sequence d_{-1}; $-e_0^2$; $-d_1$; $-e_2^2$; ... is associated, then *i*) $d_{-2}>0$, $e_{-2}=0$ and *ii*) the first non-zero element of the sequence should be strictly positive.

Within the theorem the notion "principal term" is used in the sense of Pontryagin: if

$$h(z,w)=\sum_{m,n}a_{mn}z^m w^n$$

is a polynomial in two variables, $a_{rs}z^r w^s$ is the principal term of the polynomial if $a_{rs}\neq 0$ and for any other term with $a_{mn}\neq 0$ one of the following is possible: i) $r>m$, $s>n$; ii) $r=m$, $s>n$; iii) $r>m$, $s=n$. Note that any quasi-polynomial may be given the form $h(z,e^z)$ with $h(z,w)$ a polynomial in two variables.

C. The mathematical model and the problem statement

Consider the class of scalar nonlinear time-delay systems described by

$$\dot{x}(t)=a-bx(t-\tau)\psi(x(t-\tau))\ ,\ a>0\ ,\ b>0 \tag{16}$$

where $\psi(\cdot)$ is a monotone increasing nonlinearity, $\tau>0$ is a constant time-delay and the initial condition $x(\theta)=\varphi(\theta)$, for $\theta\in[-\tau,0]$, where $\varphi\in\mathscr{C}(-\tau,0;\mathbb{R})$.

The differential equation (16) may model, for instance, the fluid dynamics in high-performance networks. We refer here to the mathematical model proposed in [10] for describing the rate control algorithm in order to avoid the congestion in communication networks. Considering the case when a collection of flows uses a single resource and shares the same gain parameter K, the model of [10] reads as

$$\dot{y}(t)=K[w-y(t-\tau)p(y(t-\tau))]\ ,\ K>0\ ,\ w>0 \tag{17}$$

where $\tau>0$, assumed constant, represents the round-trip time and $p(\cdot)$ can be viewed as "the probability a packet produces a congestion indication signal", thus being "positive, continuous, strictly increasing function of y and bounded above by unity" [10].

We shall turn now to the mathematical model (16). Let $\bar{x}$ be the unique equilibrium of (16), thus verifying $a=b\psi(\bar{x})\bar{x}$. Using a change of the coordinates, $\xi=x-\bar{x}$, one can shift the equilibrium point $\bar{x}$ to the origin so that the system (16) can be written into the form:

$$\dot{\xi}=-b\phi(\xi(t-\tau)) \tag{18}$$

where the nonlinear function

$$\phi(\xi)=\psi(\bar{x}+\xi)(\bar{x}+\xi)-\psi(\bar{x})\bar{x}. \tag{19}$$

verifies the conditions

$$\phi(0)=0\ ,\ \frac{\phi(\xi)}{\xi}>0. \tag{20}$$

Instead (18), we shall consider - without loss of the generality - the nonlinear system

$$\dot{\xi}=-\phi(\xi(t-\tau)) \tag{21}$$

with $\tau>0$ and $\xi(\theta)=\chi(\theta)$, for $\theta\in[-\tau,0]$, where $\chi\in\mathscr{C}(-\tau,0;\mathbb{R})$.

The aim of this paper is to obtain conditions for absolute stability of the class of systems described by (21), where the nonlinearity $\phi(\cdot)$ is a monotone increasing function verifying (20).

II. THE ABSOLUTE STABILITY RESULT

A. The frequency domain condition

Following the absolute stability approach which is described in section I, the system (21) can be written as a negative feedback connection of the linear block

$$\begin{aligned}\dot{\xi}(t)&=\mu(t)\\ \sigma(t)&=\xi(t-\tau)\end{aligned} \tag{22}$$

with a nonlinear one, the interconnection rule being

$$\mu(t)=-\phi(\sigma(t)). \tag{23}$$

On the other hand, the system (21) is of the form (6) with $A=b_0=c_0=\gamma_0=0$ and $B_k=b_k=c_k=0$ for $k=\overline{1,r}$,

$\gamma_1 := \gamma = 1$. In both cases the transfer function of the linear block results

$$H(s) = \frac{e^{-\tau s}}{s}, \tag{24}$$

and one remarks the system is in the critical case a) of the Theorem 2: a simple zero pole. It can be easily seen that the linear block has the limit stability property since

$$\Re es\, H(s)\,|_{s=0} = 1 > 0 \tag{25}$$

or, equivalent, it is verified the condition (9).

The frequency domain inequality (10) is written in our case as

$$\frac{1}{k} + \Re e\,(1+\imath\omega\theta)\frac{e^{-\imath\omega\tau}}{\imath\omega} = \frac{1}{k} + \left(\theta\cos\omega\tau - \frac{\sin\omega\tau}{\omega}\right) > 0 \tag{26}$$

and, dividing by $\tau > 0$ one obtains

$$\frac{1}{k_0} + \theta_0\cos\lambda - \frac{\sin\lambda}{\lambda} > 0\ ,\ \forall\lambda > 0 \tag{27}$$

where we denoted $\lambda := \omega\tau$, $\theta_0 := \frac{\theta}{\tau}$ and $k_0 := k\tau$. The inequality (27) can be written as a *minimax* problem

$$\max_{\theta_0\geq 0}\min_{\lambda\geq 0}\left(\theta_0\cos\lambda - \frac{\sin\lambda}{\lambda}\right) > -\frac{1}{k_0}. \tag{28}$$

Let

$$f(\lambda) = \theta_0\cos\lambda - \frac{\sin\lambda}{\lambda}\ ,\ \forall\lambda > 0 \tag{29}$$

be the function under evaluation. On can see that $f(0) = \theta_0 - 1$ and the condition $f(0) > -\frac{1}{k_0}$ will give $\theta_0 > 1 - \frac{1}{k_0}$. For $\lambda_d \to \infty$ and $\lambda_d = n\pi$ the most unfavorable case is $\cos\lambda_d = -1$ and $f(\lambda_d) = -\theta_0 > -\frac{1}{k_0}$ gives $\theta_0 < \frac{1}{k_0}$. We have obtained the general boundary conditions:

$$1 - \frac{1}{k_0} < \theta_0 < \frac{1}{k_0} \tag{30}$$

and one can see that the alternate sign of the function impose as necessary the condition $k_0 < \infty$. Also, one remarks the inequality (30) shows that the larger θ_0 is the smaller is k_0 and as a consequence the narrow is the absolute stability sector.

A summary analysis shows that the interval $(0,\pi)$ is interesting for the variation of function f. First of all, one observes that the second term in (29) is the function $\mathrm{sinc}(\lambda) = \sin\lambda/\lambda$ whose principal lobe has positive values on $(0,\pi/2)$ and due to the minus sign it has a unfavorable influence for our *minimax* problem. On the other hand, for $\lambda \in (\pi/2,\pi)$ both terms of the function have negative values. Evaluating $f(\pi/2) = -2/\pi > -1/k_0$, $\forall\theta_0$ we obtain an estimation for k_0:

$$1 - \frac{1}{k_0} < 1 - \frac{2}{\pi} < \frac{2}{\pi} < -\frac{1}{k_0}. \tag{31}$$

B. The analysis of the function on intervals

Concerning the *minimax* problem, the general analysis of the function and its derivative

$$f'(\lambda) = -\theta_0\sin\lambda - \frac{\lambda\cos\lambda - \sin\lambda}{\lambda^2} \tag{32}$$

leads to the following remarks:

- $\lambda = n\pi$: $f(n\pi) = (-1)^n\theta_0$ which shows again that θ_0 cannot be too large since for $n = (2p+1), p = 0,1,\ldots$ it would give a more restrictive condition for the absolute stability sector: $k_0 < \frac{1}{\theta_0}$.
- $\lambda = n\frac{\pi}{2}$: $f((2p+1)\pi + \frac{\pi}{2}) > 0$ and it is not important for the problem; $f(2p\pi + \frac{\pi}{2}) = -\frac{1}{2p\pi+\frac{\pi}{2}} < 0$ and the smallest value (having the maximum modulus) is $f(\frac{\pi}{2}) = -\frac{2}{\pi}$.

B1. Consider now the interval $\lambda \in (0,\pi)$.
a) $\lim_{\lambda\to 0_+} f'(\lambda) = 0$ which means $\lambda = 0$ is an extremum. Making use of Taylor expansion we obtain $f'(\lambda) = (\frac{1}{3} - \theta_0)\lambda + o(\lambda^2)$ and we deduce that on $(0,\pi)$: i) if $0 \leq \theta_0 < \frac{1}{3}$ then $\lambda = 0$ is a minimum and ii) if $\theta_0 > \frac{1}{3}$ then $\lambda = 0$ is a maximum.
b) Evaluating the derivative in $\lambda = \frac{\pi}{2}$ we obtain an other point of interest for our analysis on θ_0: $\frac{4}{\pi^2}$. For $\theta_0 = \frac{4}{\pi^2}$, $f(\frac{\pi}{2}) = -\frac{2}{\pi}$ is a minimum on $(0,\pi)$ and f is strictly increasing for $\theta_0 < \frac{4}{\pi^2}$ and decreasing otherwise.

We thus have obtained the following intervals of interest for θ_0 when $\lambda \in (0,\pi)$:

- $\theta_0 \in (0,\frac{1}{3})$: $f(0) = \theta_0 - 1 < \frac{2}{3}$ is a minimum on $(0,\pi)$.
- $\theta_0 \in (\frac{1}{3},\frac{4}{\pi^2})$: $f(0)$ is a maximum on $(0,\pi)$ and there exists a minimum in $(0,\frac{\pi}{2})$, then the function rises; $f(\pi) = -\theta_0 < 0$; $f'(\pi) = \frac{1}{\pi} > 0$, $\forall\theta_0$.
- $\theta_0 \in (\frac{4}{\pi^2},\infty)$: the minimum on $(0,\pi)$ is within the interval $(\frac{\pi}{2},\pi)$ and $f(\overline{\lambda}_0) < -\frac{2}{\pi}$.

B2. The analysis on the intervals $(2p\pi,(2p+1)\pi)$, $p = 1,2,\ldots$ gives the following conclusions:

- $\theta_0 < \frac{1}{(2p\pi+\frac{\pi}{2})^2} < \frac{1}{3}$: f has a negative minimum within the interval $(2p\pi, 2p\pi + \frac{\pi}{2})$.
- $\theta_0 > \frac{1}{(2p\pi+\frac{\pi}{2})^2}$: f has a negative minimum within the interval $(2p\pi + \frac{\pi}{2}, (2p+1)\pi)$.

B3. On the intervals $((2p+1)\pi,(2p+2)\pi)$, $p = 1,2,\ldots$ the extremum points are positive maxima and, are not of interest for our analysis.

We conclude that the intervals of interest for checking the minima have the general form

$$\lambda \in (2p\pi,(2p+1)\pi),\quad p = 0,1,\ldots. \tag{33}$$

Let $\overline{\lambda}_p$ be the solution of $f'(\lambda) = 0$ on such an interval; it verifies also

$$\tan\lambda_p = \frac{\lambda_p}{1 - \theta_0\lambda_p^2} \tag{34}$$

and one can compute the minimum on the interval

$$f(\lambda_p) = \left[\theta_0(1 - \theta_0\lambda_p^2)\right]\frac{\sin\lambda}{\lambda} < 0 \tag{35}$$

Since on such intervals $\text{sinc}(\lambda) > 0$, it results that for $p \geq 1$ this minimum is negative no matter where it is placed: either within the first or second half of the interval.

We can estimate now that the larger $\theta_0 > 0$ is the smaller the negative minimum will be. This means that θ_0 cannot be increased too much since the *minimax* problem requires the maximization of the minima with respect to θ_0.

C. The analysis of the local minima on intervals

The sequence of the local minima is defined by the equation

$$(1-\theta_0\lambda^2)\sin\lambda - \lambda\cos\lambda = 0 \tag{36}$$

with the solution defined by (34) and the value of a local minimum

$$f(\lambda_p) = \left(\theta_0 + \frac{1}{\theta_0\lambda_p^2 - 1}\right)\cos\lambda_p. \tag{37}$$

The analysis on each interval of interest for θ_0, namely those we have found in section *B* (B1 and B2), is simple but tediously. It reveals that the function of the minima is the same in all cases and has the general form

$$g(x) = -\frac{1+\theta_0(\theta_0 x - 1)}{\sqrt{(\theta_0 x - 1)^2 + x}} \tag{38}$$

where $x = \lambda_p^2 \geq 0$. The derivative of the "minima" function

$$g'(x) = -\frac{(3\theta_0 - 1) - \theta_0^2 x}{2[(\theta_0 x - 1)^2 + x]^{\frac{3}{2}}} \tag{39}$$

gives the extremum point at

$$x = \frac{(3\theta_0 - 1)}{\theta_0^2} \geq 0, \tag{40}$$

which means $\theta_0 \geq \frac{1}{3}$.

One has thus to solve

$$\max_{\theta_0 \geq \frac{1}{3}} \left\{ -\frac{1+\theta_0(\theta_0\lambda_0^2 - 1)}{\sqrt{(\theta_0\lambda_0^2 - 1)^2 + \lambda_0^2}} \right\} \tag{41}$$

where λ_0 is the zero of (36) for a choice of θ_0. One can use for this purpose the mathematical software packages.

The significant local minima of the function f, computed by using MATLAB software for $\theta_0 \geq \frac{1}{3}$ are given in Table I. Fig. 3 and Fig. 4 (the zoom in on $(0,\pi)$) show the graphical representations of $f(\lambda)$ for both cases $\theta_0 \in [\frac{1}{3}, \frac{4}{\pi^2}]$ and $\theta_0 > \frac{4}{\pi^2}$. One can observe that, as we previously have estimated, the values of minima decrease when θ_0 increases. On the other hand, it can be seen that for any of these functions the smallest minimum is within the interval $(0,\pi)$. In order to solve (41) we have thus to find the maximum of these absolute minima on $(0,\pi)$. The curve of the absolute minima of $f(\lambda)$ with respect to θ_0 is illustrated in Fig. 5; also, the significant values are in Table I. We conclude that for the case $\theta_0 \in (\frac{1}{3}, \infty)$, the minimum we search for is $\overline{f}_{min} = -0.64968$ obtained for $\theta_0 = 0.38$.

Fig. 6 shows the curves of $f(\lambda)$ for $\theta_0 \in [0, \frac{1}{3})$ (the negative value of θ_0 is taken from curiosity). These confirm that the absolute minimum is in this case $\lambda = 0 \in [0,\pi)$. But for this case we have already obtained that $f(0) = g(0) = \theta_0 - 1 \in [-1, -\frac{2}{3})$, thus the maximum "absolute minimum" for $\theta_0 \in [0, \frac{1}{3})$ is less than -0.64968, the value of the minimum for $\theta_0 \in (\frac{1}{3}, \infty)$.

We conclude that the solution of the *minimax* problem (28) is

$$-0.64968 \geq -\frac{1}{k_0} \tag{42}$$

and thus we have determined the sector of absolute stability

$$k < \frac{1.5392}{\tau}. \tag{43}$$

The result of our analysis reads as follow

Consider the time-delay nonlinear system (21) with $\phi(\cdot)$ a continuous monotone increasing nonlinearity. The system is asymptotic stable for all nonlinear (and linear) functions verifying (20) and which are within the sector $(0, \frac{1.5392}{\tau})$.

Remark 2: Obviously, the absolute stability sector (43) is narrowed by increasing the time delay. Regarding the mathematical model (16) for the congestion problem in communication networks this time-delay dependent absolute stability condition means that the class of the nonlinear functions $\psi(\cdot)$ of the type (7) is diminished by increasing the round-trip time τ.

III. CONCLUSIONS

This paper deals with the analysis of the absolute stability for a class of scalar nonlinear time-delay systems having monotone increasing nonlinearities. These systems can be encountered, for instance, as congestion problems in high-performance communication networks.

We have used an approach based on frequency domain inequalities of Popov type for time-delay systems in the critical case of the transfer function with a simple zero pole. In order to solve the *minmax* problem which arises from the frequency domain inequality, the analytical analysis was completed by numerical computations using MATLAB software package.

We have obtained a time-delay dependent absolute stability condition i.e. the class of the nonlinear functions, and therefore of the nonlinear systems under consideration, can be limited by large time-delays (for instance, the round-trip time in the case of the congestion problems in high-performance communication networks).

REFERENCES

[1] A. M. Letov, "Stability of nonlinear control systems: state of the art," in *Proc. 2nd All-Union Conference on Control Theory*, URSS Academic Publishing House, Moscow, vol.1 1955, (in Russian).

[2] A. M. Letov and A. I. Lurie, "Recent research and open problems in nonlinear system stability," in *Proc. of the URSS Academy Conference on Scientific Problems of Industrial Automation*, URSS Academic Publishing House, Moscow, vol.1 1957, (in Russian).

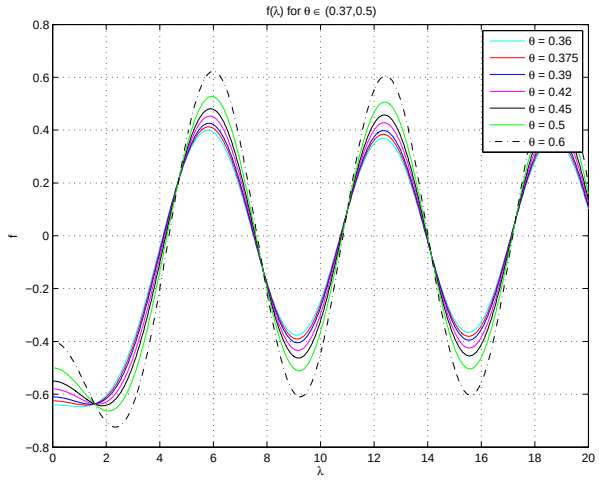


Fig. 3. Function f for $\theta \in (\frac{1}{3}, \frac{1}{2})$.

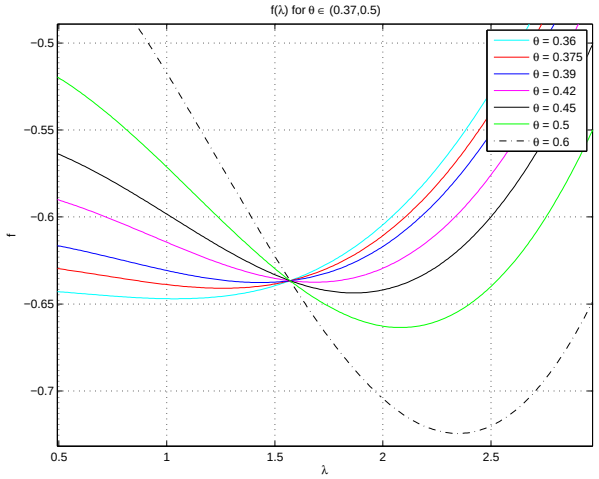


Fig. 4. Zoom in: function f for $\lambda \in (0, \pi)$ and $\theta \in (\frac{1}{3}, \frac{1}{2})$.

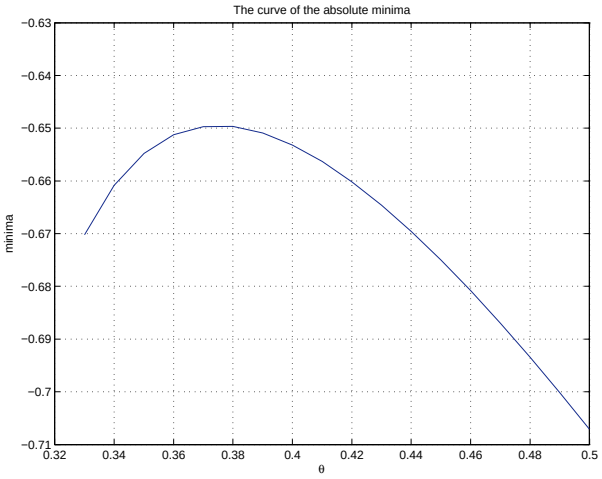


Fig. 5. The absolute minima of $f(\lambda)$ for $\theta \in (\frac{1}{3}, \frac{1}{2})$.

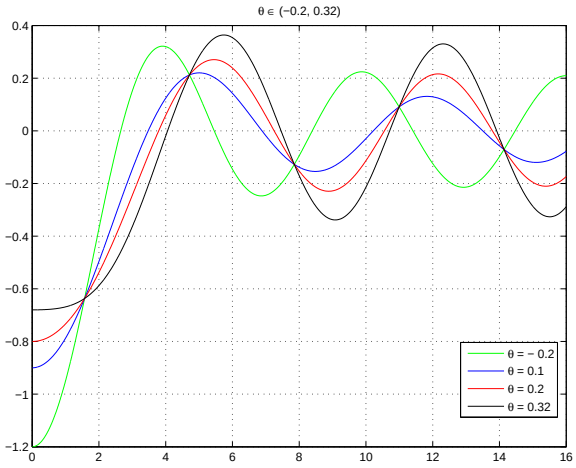


Fig. 6. Function f for $\theta \in (-0.2, \frac{1}{3},)$.

TABLE I

THE MAXIMUM MINIMA OF $f(\lambda)$ FOR $\theta \in (\frac{1}{3}, \frac{4}{\pi^2})$

θ_0	$f(\lambda_0)$
0.33	-0.6702
0.34	-0.6608
0.35	-0.6548
0.36	-0.6513
0.37	-0.6497
0.38	-0.64968
0.39	-0.6510
0.40	-0.6532
0.41	-0.6563
0.42	-0.6602

[3] V. Răsvan, "Popov theories and qualitative behavior of dynamic and control systems," *European Journal of Control*, vol. 8, no. 3, pp. 190–199, 2002.

[4] S. Lefschetz, *Stability of nonlinear control systems*. New York: Academic Press, 1965.

[5] V. M. Popov, "Stability criteria for nonlinear control systems based on laplace transform," *St. Cerc Energetic*, vol. IX, no. 1, pp. 119–136, 1959.

[6] ——, "On the absolute stability of nonlinear control systems," *Avtom. i telemekhanika*, vol. 22, no. 8, pp. 961–979, 1961.

[7] V. Răsvan, *Absolute stability of time lag control systems*, 1st ed. Bucharest: Editura Academiei, 1975, (in Romanian; Russian revised edition by Nauka, Moscow, 1983).

[8] M. A. Aizerman and F. R. Gantmakher, *Absolute stability of regulator systems*. Moscow: USSR Academy Publ. House, 1963, (in Russian).

[9] V. M. Popov, *Hyperstability of Control Systems*, 1st ed. Berlin-Heidelberg-New York: Springer Verlag, 1973.

[10] F. Kelly, "Mathematical modelling of the internet," in *Mathematics Unlimited - 2001 and Beyond*, B. Engquist and W. Schmid, Eds. Berlin: Springer-Verlag, 2001, pp. 685–702.

Simulation of a Monitoring Agent Based System

Ancuța O. Dobîrcău, Corina D. Nemeş, Silviu Folea, Honoriu Vălean, Adina Morariu,

Technical University of Cluj-Napoca, Automation Department

***Abstract*— The paper describes a monitoring agent-based system and simulation of this kind of system. In the first part of the paper a brief introduction in multi-agent systems is presented. In the second part a short presentation of agent technology is made. The third part describes the hardware and software architecture. The implementation section of fourth part describes the application and its functionalities. The fifth part presents experimental results and the last one deals with the conclusions. The agents are able to communicate in order to allocate the hardware resources via Wi-Fi. In this way the system implements virtual redundancy, e.q. that if a part of the system fails, other parts will assume its tasks. Virtual redundancy provides a low cost reliable configuration. Another novelty of the system design is Wi-Fi with ultra-low power consumption, in a very small package that is running on a battery for years and offers a platform for sensor measurements using the Internet network infrastructure.**

I. INTRODUCTION

Most of the complex real world problems are solved using distributed environments as in [1]. Multi-agent systems (MAS) can model complex systems and introduce the possibility for the agents to have common or conflicting goals. These agents may interact with each other both indirectly (by acting on the environment) or directly (via communication and negotiation). Agents may decide to cooperate for mutual benefit or may compete to serve their own interests. They are being used in an increasingly wide variety of applications, ranging from comparatively small systems for personal assistance to open, complex, mission critical systems for industrial applications [2].

Industry is a very important field of application for multi-agent systems because they are where the first multi-agent system techniques were experimented with and demonstrated their initial potential. One of the key components of multi-agent systems is communication. In fact, agents need to be able to communicate with users, with system resources, and with each other if they need to cooperate, collaborate, and negotiate and so on. In particular, agents interact with each other by using some special communication languages; called agent communication languages that rely on speech act theory as in [3] and that provide a separation between the communicative acts and the content language. The first agent communication language with a broad uptake was KQML [4]. KQML was developed in the early 1990s as part of the US government's ARPA Knowledge Sharing Effort. It is a language and protocol for exchanging information and knowledge that defines a number of performative verbs and allows message content to be represented in a first-order logic-like language called KIF [5]. Currently the most used and studied agent communication language is the FIPA ACL, which incorporates many aspects of KQML [6]. The primary features of FIPA ACL are the possibility of using different content languages and the management of conversations through predefined interaction protocols. Coordination is a process in which agents engage to help ensure that a community of individual agents acts in a coherent manner [7]. There are several reasons why multiple agents need to be coordinated including: agents' goals may cause conflicts among agents' actions, agents' goals may be interdependent, agents may have different capabilities and different knowledge, and agents' goals may be more rapidly achieved if different agents work on each of them. Coordination among agents can be handled with a variety of approaches including organizational structuring, contracting, multi-agent planning and negotiation. Negotiation is probably the most relied upon technique for coordinating agents. In particular, negotiation is the communication process of a group of agents in order to reach a mutually accepted agreement on some matter [8]. Negotiation can be competitive or cooperative depending on the behavior of the agents involved. Competitive negotiation is used in situations where agents have independent goals that interact with each other; they are not a priori cooperative, share information or willing to back down for the greater good. Cooperative negotiation is used in situations where agents have a common goal to achieve or a single task to execute. In this case, the multi-agent system has been centrally designed to pursue a single global goal [9], [10].

The Tag4M is an embedded system incorporating a sensor interface, a 32-bit CPU, 128kB RAM, 512kB ROM and 2kB NVM memory, eCos real time operating system, complete Wi-Fi networking solution, TCP/IP network stack, crypto accelerator, power management system and real time clock [11], [12].

This platform has been designed from inception to maximize battery life and achieves ultra low-power consumption with a combination of proprietary technologies. The power management module includes logic

to power-down functions not required and to control switching between sleep and active modes. When in sleep mode, the real-time clock and sensor interface remain active. The real-time clock can be programmed to wake up the Tag4M at any required interval. Similarly, the Tag4M can be programmed to wake up when a specified condition is detected. This "instant-on" feature of the Tag4M can be utilized by mobile devices, allowing them to remain in the low-power standby mode until Wi-Fi is needed, and then powering on the Wi-Fi "instantly". The Tag4M can wake, and join a network using WPA2-PSK in under 35 msec. This allows it to stay asleep until required, but wake and join a network without any noticeable user delay.

Wi-Fi wireless LAN adapters are much more powerful and capable of reaching data transmission rates approaching 54Mbps. Wi-Fi products also have strong security protocols (WEP, WPA or WPA2), which make them a better network solution. Generally, it is considered that key attributes for Wi-Fi are wider bandwidth and flexibility.

The Tag4M proposed system's novelty, in contrast with existing Wi-Fi solutions, is its ultra low power Wi-Fi capability which makes it suitable for sensing applications where battery power management is critical. The batteries must deliver a current peak up to 1A, but the pulse duration is very short, of about 1-2 ms, due to high transmission rate [13], [14].

II. AGENT TECHNOLOGY

Agent-**O**riented **P**rogramming (AOP) essentially models an application as a collection of components called agents that are characterized by, among other things, autonomy, proactivity and an ability to communicate. Being autonomous they can independently, without the direct intervention of humans (or other agents), carry out complex, long-term tasks, and have control over their own actions and internal state. Being proactive they can take the initiative to perform a given task even without an explicit stimulus from a user. Being communicative they can interact with other entities to assist with achieving their own and others' goals [15].

A multi-agent system is a network composed by entities which work together to solve problems that cannot be solved by each individual entity. The architectural model of an agent-oriented application is intrinsically peer to peer, as any agent is able to initiate communication with any other agent or be the subject of an incoming communication at any time.

When adopting an agent-oriented approach to solving a problem, an agent-oriented middleware is needed to provide the domain-independent infrastructure. In this case JADE (**J**ava **A**gent **DE**velopment framework) is used, a completely distributed middleware system with a flexible infrastructure, allowing easy extension with add-on modules. The framework facilitates the development of complete agent-based applications by means of a runtime environment where agents can live, a library of classes that programmers use (directly or by extending them) to develop their agents and a suite of graphical tools that allows administrating and monitoring the activity of running agents. JADE is written completely in Java so it benefits from the huge set of language features and libraries and thus offers a rich set of programming abstractions [15].

Each running instance of the JADE runtime environment is called a Container and contains several agents. Each of the containers can be on the same computing system or different systems. The set of active containers is called a Platform. Each platform must have a Main Container always active and all other containers register with it as soon as they start [16]. It holds two specialized agents called the AMS (**A**gent **M**anagement **S**ystem) agent and the DF (**D**irectory **F**acilitator) agent. The AMS agent is the only agent that can create and kill other agents, kill containers and shut down the platform. The DF agent implements a yellow pages service where agents in the platform can register their services allowing other agents requiring those services to find them.

One of the most important characteristics of an agent is autonomy. Agents have some state but have control over their state in that they can encapsulate behaviors, in addition to these states. Therefore an autonomous agent is situated within an environment, senses that environment and is able to act on its own and to modify its internal state so that it can accomplish its goals. In multi-agent systems an agent needs to interact with other agents in order to achieve its tasks. It is possible agents have common or conflicting goals. They can communicate and negotiate with each other, cooperating for mutual benefit or may compete to serve their own interests.

In a multi-agent system, agents communicate using a special language. The **F**oundation for **I**ntelligent **P**hysical **A**gents (FIPA) develops computer software standards for heterogeneous and interacting agents and agent-based systems [17]. Agent communication is the most important category of the FIPA multi-agent system model and deals with **A**gent **C**ommunication **L**anguage (ACL) messages, which represent actions or communicative acts[18]. The second fundamental aspect of agent systems in FIPA specifications is agent management, a normative framework that establishes a model for the creation, registration, location, communication, migration and operation of agents. Agent management consist from **A**gent **P**latform (AP), the physical infrastructure in which agents are deployed, Agent, the process that inhabits an AP, **A**gent **M**anagement **S**ystem (AMS), a mandatory component of an AP, which retains a unique name and the current state of every agent on the AP, **D**irectory **F**acilitator (DF), an optional component of an AP providing yellow pages services to other agents, and the **M**essage **T**ransport **S**ervice (MTS), a service provided by an AP to transport FIPA-ACL messages between agents as shown in [19].

If an agent wants to make public its services, it should find an appropriate facilitator agent and ask for registration, obtaining this way the right to provide these services. An

agent may search in order to request information from a DF.

The JADE platform allows the coordination of multiple FIPA-compliant agents and the use of the standard FIPA-ACL communication language. It includes all mandatory components that manage the platform, that is the ACC (**A**gent **C**ommunication **C**hannel), the AMS and the DF.

The agent platform can be distributed on several hosts. On each host there is only one application, and therefore only one Java Virtual Machine (JVM) that is executed. Each JVM work as a container for agents and provides a runtime environment for agent execution allowing agents to concurrently execute in the same host.

All agent communication is performed through asynchronous message passing. JADE creates and manages a queue of incoming ACL messages, private to each agent. Agents can access their queue in different ways like blocking, polling, timeout and pattern matching based. The full FIPA communication model has been implemented including interaction protocols, envelope, ACL, content languages, encoding schemes, ontologies and transport protocols. The transport mechanism adapts to each situation choosing the best available protocol [20]. When crossing platform boundaries, messages are automatically transformed from JADE's internal Java representation into proper FIPA-compliant syntaxes, encodings and transport protocols.

When agents are created they are automatically assigned a globally unique identifier and a transport address. Each agent has to execute tasks. JADE uses the Behaviour abstraction to model the tasks that an agent is able to perform and agents instantiate their behaviours according to the needs and capabilities.

Agents are implemented as one thread per agent which is important in environments with limited resources, but they often need to execute parallel tasks. JADE supports scheduling of cooperative behaviours. A scheduler, implemented by the base Agent class and hidden to the programmer, carries out a round-robin non-preemptive policy among all behaviours available in the ready queue, allowing the execution of a behaviour until it will release the execution control by itself. This means that it is the programmer who defines when an agent switches from the execution of one behavior to the execution of another. There are implemented and ready to use behaviours for the most common tasks in agent programming, such as sending and receiving messages and structuring complex tasks as aggregation of simple ones [20].

III. SYSTEM ARCHITECTURE

A. Hardware architecture

The system architecture based on multi-agent technology is presented in Figure 1. This figure presents the network option that allows Tag4M tags to connect to the Access Points, and transfer information, via Central Agent, to a PC. On the PC is implemented a multi-agent society for receive the Tag4M tags information and manage the connection (association) between Tag4M tags and Access Points [21].

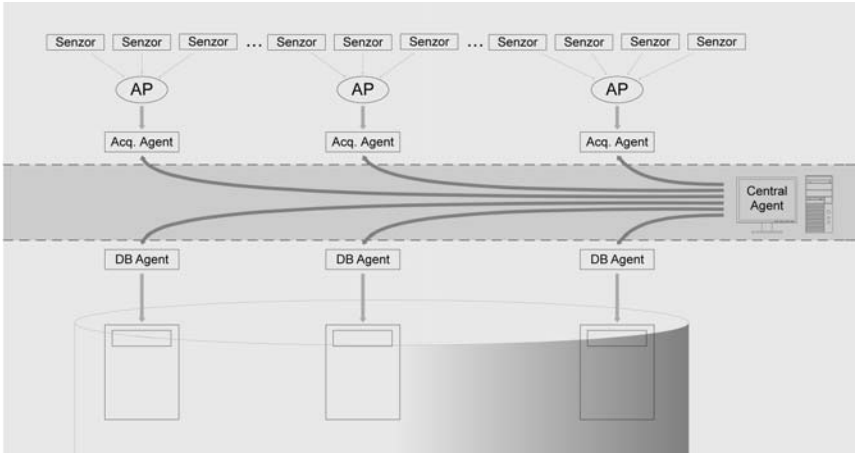


Fig. 1. System architecture

When tags wish to connect to a network, they first scan for available networks. When an available network receives a probe request, it will transmit a probe response packet containing the necessary information required to use the network including channel information. The tag then decides which Access Point is the best choice for their connection.

After the tag has decided which Access Point it wants to connect to, it will transmit a management frame with an authentication sub-type, requesting authentication to the network. In the case of open networks, the Access Point simply responds with an authentication success message and the connection process continues. In the case of a network utilizing WEP, the Access Point will attempt to validate the WEP key of the tag that wishes to access the network by sending them an authentication challenge. Once the tag gets an authentication success message from the access point, the authentication process is complete. A tag may choose to authenticate to multiple Access Points simultaneously.

When the tag is authenticated, then it will start the association process. A tag can associate to only a single Access Point at any given time. For the association process a tag will send a management frame with the associate request sub-type to the Access Point. The Access Point will check to make sure the tag has already authenticated. If an entry exists indicating the tag has already authenticated to the AP, it will generate an associate response message to the tag. Once authenticated and associated to the Access Point the Point will scan again the network and will re-start the authentication and association process with a new available Access Point.

The Tag4M presented in Figure 2 at scale 1:1 is used to measure, calculate and transmit the data to an AP [22]. In addition Tag4M-based tags can monitor and record environmental parameters such as temperature, light, humidity or rain gauge. These parameters can be monitored continuously and stored on-chip until a Wi-Fi connection is available. The Tag4M supports a large set of sensors.

Fig2. Tag4M Wi-Fi Measurement Sensor

The power-management implemented at the hardware level is enhanced at the software level, giving a continuous acting time for the sensors of about 1-5 years. This long acting time is possible, taking into account that the wake-up time is only 0.1%-0.6% comparing to the sleep time.

A. Software architecture

A system architecture based on multi-agent technology is proposed and is implemented on different levels:

- data acquisition level – composed of transducers and data acquisition equipment;
- central level – managing the communication tasks of the agents;
- database level – where the information is stored;

On each level agents are used to receive and send the information from the current level to the next one. Each level provides the following functions:

- data acquisition level – data acquisition: digital and analogue signals from sensors. This is where the acquisition agents are implemented. These agents send the information using **A**gent **C**ommunication **L**anguage (ACL) to the next level;
- central level – controller: interpretation of the incoming messages from the acquisition level; the query is extracted and sent to the database level. Also it has to create and kill agents or register them in the DF (**D**irectory **F**acilitator). It holds only one agent called the central agent;

tag needs to acquire an IP address. The tag use the DHCP protocol to request a dynamic allocation from a DHCP server, implemented on the Access Point. The tag is configured to use Power Save Poll mode. This allows the tag to sleep while waiting the DHCP response. Once an IP address is acquired, the tag will be able to communicate with the Access Point, using the UDP protocol. If an Access Point becomes unavailable, the Tag4M tags associated with the unavailable Access

- database level – storing: receiving messages from the central level and storing information into database using database agents;

The number of agents from the acquisition level is the same as the one on the database level.

The **Data Acquisition Level** is composed of sensors. The signals from these sensors get to different access points. For each access point there is an agent receiving the information from it and send it further to the central level. No modification of data is done at this level.

The **Central Level** works as a controller. It contains an agent which creates the necessary agents for the acquisition and database levels and kills them when they are no longer needed. It receives the messages from the acquisition level and finds the right agent on database level to send it. It also receives information about the number of active and properly working access points. As soon as one is down it kills the appropriate agents. It implements an informative user interface listing the active agents and the repartition of sensors to each access point.

The **Database Level** stores the information into the database using agents that manage the connection to the database. Each agent has its own table corresponding to an access point. If an access point is down the agent terminates. At this level the query is extracted from the message received from the central level and is inserted into the database. These agents need to be located near the database server because there is a considerable data flow exchange between agents and the database

In order to receive the messages from the central agent the database agent need to register their agent description that includes its own **A**gent **Id**entifier (AID) and a list of provided services to an agent facilitator. This means that the agent from the central level issues a search request to a DF to discover descriptions matching supplied search criteria. It must provide the DF with a template description. Each database agent also implements an agent interface printing the information that is to be stored.

The **Informative Interface** is listing the access points and the assigned sensors. For each access point there is an acquisition agent and a database agent, so the information from one access point is read from an agent and written in a separate table in the database. The acquisition and the database agents are shown in this interface.

The **Agent Interface** prints for each of the agents into a separate window the content of its database table.

The systems architecture can also be seen as a Model-View-Controller (MVC) architectural pattern. The model is

used to manage information and encapsulate the application's state, in this case at database level. The view renders the content of the model into a user interface element. The controller receives input and initiates a response by making calls or model objects. A controller accepts input from the user and instructs the model and the viewport to perform actions based on that input.

The information is passed through each layer, from the sensors into the database, using agents and FIPA-ACL messages. The relational database is chosen because it allows the storing of large amount of data that can be easily found by interrogating the database.

IV. IMPLEMENTATION

The agents are implemented on the JADE platform, passing the values read by sensors into the database and providing user interfaces.

The data that needs to be stored into the database contains the values read by sensors. This information has a specific structure: the date at which the value was read, the name of the sensor and the read value and is stored into a relational database designed and implemented using MySQL. MySQL is a relational database management system that runs as a server providing multi-access to a number of databases. A relational database is a collection of relations (tables), is easier to use and is currently the predominant choice in storing information.

The hardware part is a software simulation. The sensors in this application are implemented as threads that generate random numbers. These values are sent to access point but in this case are written into files. The number of sensors and access points can be set from inside the application. The sensors are optimally assigned to access points using a simple algorithm. For every access point there is an acquisition agent and a database agent. There is a central agent that checks the number of access points and creates these agents. If an access point breaks down the corresponding agents are killed and there is a mechanism implemented so that the sensors assigned to that access point get to be reassigned to other access points in an optimal way. In this case, since the access points are implemented as files, each sensor is reassigned to the file with the smallest size. This is done by the central agent using a cyclic behaviour.

The central agent has two tasks to execute. It checks in a cyclic behaviour the activity of the access points and it receives the messages from the acquisition agents which are then sent to the database agents.

One acquisition agent gets the information from one access point. At the same time the sensors keep writing into files so a synchronization method needs to be implemented, in this case each file (access point) has a semaphore. Once the semaphore is acquired, the agent starts to read the information. Line by line data from the file is sent to the central agent using ACL messages. The reading from files is implemented as a cyclic behaviour.

The central agent receives the data, this task being also implemented as a cyclic behaviour. It creates a search pattern and issues a search request to a DF in order to find the right database agent and sends the data.

For each access point there is only one database agent. When the agent is created it receives the database semaphore since there are more agents accessing a single database connection. Again, the synchronization is implemented using semaphores. The semaphore is acquired by one agent at a time. Upon consumption of the resource, the semaphore is released. After creation each agent has to register its agent identifier to DF. Agent identifier includes the agent AID and the services it provides.

The database agent receives the message from the central level and extracts the query which is then sent to the database. Each agent provides a user interface listing all the messages it receives.

The central level provides an interface listing the acquisition and database agents active on the platform as well as the access points with the assigned sensors.

Redundancy is the duplication of critical components of a system with the intention of increasing reliability of the system, usually in the case of a backup. This refers to building or designing a number of components that can take over if one part of the system fails to work. In virtual redundancy there is no physical backup implemented but if one component fails to work an alternate path is offered to ensure the needed operations. However it might degrade performance but won't stop the working.

In this application the virtual redundancy is implemented at acquisition level. Each access point has a number of sensors assigned. If one access point is not working properly the sensors assigned to it need to send the information or data is lost. Since there is no backup of that access point the sensors need to relocate themselves to other access points and send the signals. Each of the sensors must check with each available access point in order to find the one with the less number of assigned sensors and send the information to it.

Since in this application the hardware part is a software simulation the sensors are actually threads that generate a random number. The access points are files that have a number of threads assigned. If the file is deleted each of the threads finds the file with the smallest size and then writes the number into it. Fig. 3 presents the application screen.

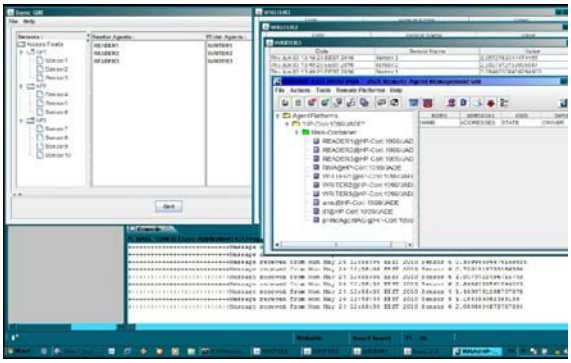


Fig3. Simulation

V. RESULTS

The application was tested using three files as access points and ten threads as sensors. This means that the central agent created three acquisition agents and three database agents. The acquisition agents are created with a local name, a file name and the semaphore of a file. The database agents are created with a local name and the database semaphore.

At the acquisition level the sensors are implemented as threads that generate random numbers and write them into files, which are the access points. Each agent reads its file and sends every line as a FIPA-ACL message to the central agent.

At the database level once agents are created, they register to a facilitator agent in order to be found by the central agent and receive the right message. Once the message is received the query is extracted and prepared for insertion into the database. Agents acquire the database semaphore to obtain a database connection. Every agent creates its own table into the database and starts the insertion process. After the insertion completes the semaphore is released and acquired from another agent.

At the central level, there is only one agent which has two tasks: it has to create the acquisition and the database agents for each access point and then check the functioning of every access point deciding in case of an access point failure which are the agents that need to be killed. Secondly it's assigned with receiving messages from the acquisition agents and sending them to the appropriate database agent. It has to interrogate a service facilitator in finding the right database agent to send the message to.

If an access point is down the sensors have to be reassigned to other access points in an optimal way.

The central agent provides a user interface for monitoring the activity of acquisition and database agents, and listing the access points and the sensors assigned to each of these APs. Every database agent offers an interface for listing the messages it receives.

VI. CONCLUSIONS

The main goal of using agents is that they are able to communicate with systems resources, users and each other using FIPA-ACL messages. Agents have a queue of messages where the JADE runtime posts messages sent by other agents. This way they can work on platforms that run on different hosts

JADE is a software platform that provides basic middleware-layer functionalities which are independent of the specific application and which simplify the realization of distributed applications that use the software agent abstraction.

In this application, sensors are implemented as threads and the access point as files. So the sensors generate random numbers which are then written into a file. In order to improve the application all that is needed is to create the hardware part. This way the signals read by sensors in a real application can be sent to an access point and then wireless to a computer.

REFERENCES

[1] J.J. Gomez-Sanz, J. Pavon and F. Garijo, "Meta-models for building multi-agent systems", Proceedings of the 2002 ACM Symposium on Applied Computing, Madrid, Spain, 2002, pp. 37–41.

[2] N. R. Jennings and M. Wooldridge, "Applications of Agent Technology", editors, Agent Technology: Foundations, Applications, and Markets. Springer-Verlag, March 1998.

[3] Searle JR, "Speech acts: an essay in the philosophy of language", Cambridge University Press, Cambridge, UK, 1969.

[4] Tim Finin, Yannis K Labrou, and James Mayfield, "Evaluating KQML as an agent communication language", InBook, Intelligent Agents II, 1996.

[5] Genesereth, M. R. and Ketchpel, S. P., "Software agents. Communications of the ACM", 1994, 37 (7):48-53

[6] Grosof BN, Labrou Y., "An approach to using XML and a rule based content language with an agent communication language", IBM Research report, 1999.

[7] Nwana HS. "Software agents: an overview". The Knowledge Engineering Review, 11(3), 1996, pp 205-244.

[8] S. Bussmann, J. Müller, "A Negotiation Framework for Cooperating Agents", In S.M.Deen (ed.), Proc. of the CKBS-SIG (CKBS'92), DAKE Centre, Univ. of Keele, 1992, pp. 1-17.

[9] N.R. Jennings, "An agent-based approach for building complex software systems", Communications of the ACM 44, 2001, (4), pp. 35–41.

[10] Fabio Bellifemine, Agostino Poggi, Giovanni Rimassa, "Developing Multi-agent Systems with JADE", ATAL 2000

[11] G2 Microsystems, "G2C547 SoC Data Sheet", product brief, G2 Microsystems, Inc., Campbell, CA, USA, 2008

[12] G2 Microsystems, "G2C547 Software, Example Application", Technical Report. G2 Microsystems, Inc., Campbell, CA, USA, 2008

[13] M. Ghercioiu, S. Folea, I. Monoses, "The WiTAG - a WiFi Sensor TAG", The 2007 International Conference on Wireless Networks (ICWN'07: June 25-28, 2007), Las Vegas, Nevada, USA, pg. 376

[14] S. Folea, M. Ghercioiu, "Ultra-Low Power Wi-Fi Tag for Wireless Sensing", 2008 IEEE International Conference on Automation, Quality and Testing Robotics, May 22-25, Cluj-Napoca, Romania.

[15] F. Belifemine, G. Caire, and D.P.A. Greenwood, "Developing multi-agent systems with JADE", *John Wiley & Sons Ltd*, 2007

[16] G. Caire, "JADE Tutorial: JADE Programming for beginners", unpublished

[17] FIPA – Foundation for Intelligent Physical Agents, *http://www.fipa.org*

[18] FIPA SC00037J – Communicative Act Library Specification, *http://www.fipa.org/specs/fipa00037/SC00037J.html*

[19] FIPA XC00023H – Agent Management Specification, *http://www.fipa.org/specs/fipa00023/XC00023H.html*

[20] Java Agent DEvelopment Framework – Technical Description, *http://jade.tilab.com*

[21] Adina Morariu, Silviu Folea, Honoriu Valean, "Reliable Agent Based Monitoring System", 24th International Conference on Computers and Their Applications, April 8-10, 2009 Holiday Inn Downtown-Superdome New Orleans, Louisiana, USA, pp. 99-105, INSPEC and DBLP

[22] S. Folea, M. Ghercioiu, "Tag4M, a Wi-Fi RFID Active Tag Optimized for Sensor Measurements", InTech Education and Publishing, Austria, 2009

The Control of Infected Cell Populations

R. Dobrescu, *Member IEEE*, E. Iancu, *Member IEEE*, E. Petre, *Member IEEE*, Ionela Iancu

***Abstract* — The medical world manifests an increasing interest in the modeling of physiological systems. Mathematical modeling turns the synthesis of the experimentally obtained data into a unitary system, allowing the underlining of the internal structure and the causal links between component parts and measures the weight with which every subsystem intervenes in accomplishing the system's functions. This paper presents a possibility to model a population of infected cells, in order to keep an infection under control. The authors have taken into consideration the existence of a dead-time between the evolution of sepsis and the body's immune response. Furthermore, the circulatory system introduces its own dead-time. The purposes are to predict the necessary treatment and to eliminate the infected population. All of these are possible due to the use of the Smith predictor.**

I. Introduction

MATHEMATICAL modeling and simulation are extremely effective methods, frequently used in all areas. During past years, important steps have been taken in order to involve systems theory and in the study of living organisms. Unfortunately, these attempts have not always been appreciated by biologists and medics, the main argument being the insufficient accuracy with which abstract systems model living systems. There are two objective causes that limit the performance of these models:

- The complexity of living organisms
- Difficulties met with experimental data acquisition and processing

The complexity of living systems, even in the case of inferior organisms is vastly superior to technical man-made structures. These consist of a large number of sub-systems, between which multiple connections are established. The isolation of a biological sub-system, with the purpose of studying it, is, inevitably, accompanied by altering the system's function and behaviour. For this reason, experimental data differs from that corresponding to a normal functioning. In conclusion, it is recommended that the system is studied, as much as possible, in its whole. This conception means treating living systems as causal systems, deterministic or stochastic, multi-variable and ierarchybased. The living organism is a complex of system, in a dynamic balance, permanently controlled through control loops (*feedback*). A characteristic of evolved living organisms is the robustness and adaptability to the environment. These qualities are the consequence of the control structures. Robustness is ensured by multiple control loops for each parameter, while short-term adaptability is achieved with the help of adaptive control systems, with variable structure, optimal or extreme.

Another difficulty in modeling and simulation is the non-linear character of most living systems as well as the non-stationary parameters. Data acquisition and processing raises problems technical in nature, because, structurally, a large part of the system's parameters are either inaccessible, or can not be converted into measurable units. For this reason, most of the times, the data that is available is insufficient to globally characterise the system.

From a systemic point of view, processes that take place at a cellular level (including cancerous tumors) have a strong non-linear character. Although, regarding their study and modeling remarkable progress has been made, the development and application of modern methods is slower in comparisons to other areas. This delay is caused, mainly, by two characteristics typical for bioprocesses.

- First of all, modeling these is exceptionally difficult. These systems consist of a series of interactions with other processes and, as a result, their functioning and especially their growth dynamics are, most of the times, hard to comprehend, strongly non-linear and non-stationary. Also, the reproductability of experiments is uncertain, while the lack of measurement accuracy can lead to a series of identification issues.
- Secondly, the application of monitoring strategies is confronted, in most cases, with the absence of typical instrumentations, secure and cheap, destined for the direct and/or in real time monitoring of biological and biochemical variables. Currently, the market offers few sensors capable of supplying this kind of

This work was supported by National Center for Programme Management - CNMP, Romania, under Grant 61-031/2007.

Radu Dobrescu, member IEEE, is with the Politehnica University of Bucarest, Faculty of Automation and Computers, Splaiul Independentei street, no. 313, sectorul 6, Bucuresti, 060042, Romania; (e-mail: radud@ isis.pub.ro).

Eugen Iancu, member IEEE, is with the University of Craiova, Faculty of Automation, Computers and Electronics, Decebal street, no. 107, Craiova, 200440, Romania (e-mail: eugen.iancu@automation.ucv.ro).

Emil Petre, member IEEE, is with the University of Craiova, Faculty of Automation, Computers and Electronics, Decebal street, no. 107, Craiova, 200440, Romania (e-mail: epetre@automation.ucv.ro).

Ionela Iancu is with the University of Medicine and Pharmacy of Craiova, Petru Rares street, no. 2, Craiova, 200349, Romania (e-mail: iancu@automation.ucv.ro).

measurements, while inaccessible or immeasurable parameters need to be determined through off-line laboratory analysis. The cost and duration of such analysis limits their frequency and leads to an increase in overall expenses.

For the overcoming of such obstacles, it is necessary to use advanced modeling and identification technics that use software sensors, in order to reproduce states and/or immeasurable parameters. Mathematical modelling turns the synthesis of the experimentally obtained data into a unitary system, allowing the underlining of the internal structure and the causal links between component parts and measures the weight with which every subsystem intervenes in accomplishing the system's functions. Simulation ensures the validation of concurrent theories, the understanding of physiopathological modifications and suggests relevant experiments. From what we have shown, we can tell that the modelling and simulation of living systems will be part of the modern trend of integrating obtained knowledge through inter-disciplinary collaborations.

II. Model Predictive Control

Model Predictive Control (MPC) has known a spectacular development in the last decade. Its success is due to good performance obtained for processes with "*difficult*" dynamics (non-minimal phase, dead-times, etc.), that lead to difficulties for classical automated control. However, predictive control methods are not generally useable. They can only be applied to those processes for which a mathematical model and reference are known a-priori. These two conditions basically, represent the disadvantage of predictive methods. These, however, can not be avoided, because in order to predict a process' future behavior, we need to use a process model and the use of a pre-established reference benchmark. The main objective of MPC is to make predictions regarding a process' future actions, based on a mathematical model and to select, according to these and the imposed reference, the correct command inputs. Types of MPC:

a) *Linear MPC*
- A linear model is used: $\dot{x} = Ax + Bu$
- A square cost function: $F = x^T Qx + u^T Ru$
- Linear restrictions: $Hx + Gu < 0$

b) *Non-linear MPC*
- A non-linear model is used: $\dot{x} = f(x,u)$
- The cost function may not be squared: $F(x,u)$
- Non-linear restrictions: $h(x,u) < 0$

All model predictive control (MPC) systems rely on the idea of generating values for process inputs as solutions of an on-line (real-time) optimization problem. That problem is constructed on the basis of a process model and a predictive algorithm. Figure 1 shows the structure of a typical MPC system. The whole philosophy of predictive control regarding the creation of an anticipative effect, by

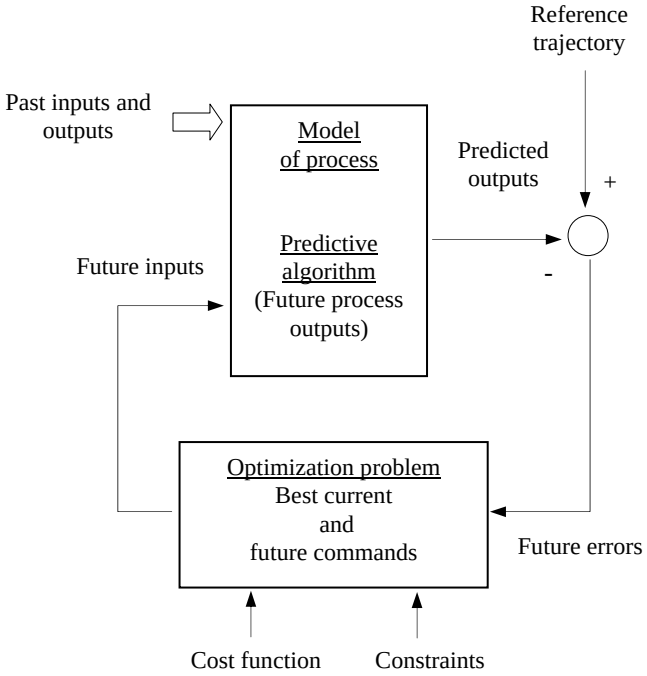


Fig. 1. Model predictive control scheme.

using explicit knowledge of the future trajectory can be presented shortly as follows:

- The defining of a mathematical model of the system, in order to predict its future behavior.
- The minimizing of a square cost function on a finite horizon, by using prediction errors
- The elaboration of a future control sequence, with only the first value being applied to the system and model.
- The repeating of the entire procedure for the next sample, according to the chosen horizon.

III. Smith Predictor for Delayed Systems

The more popular scheme for control processes affected by time delay was proposed by O. J. M. Smith [1] and is shown in Figure 2. Let $P(s) = G(s)e^{-s\tau}$ be the transfer function of the process and let's indicate the setpoint with y°.

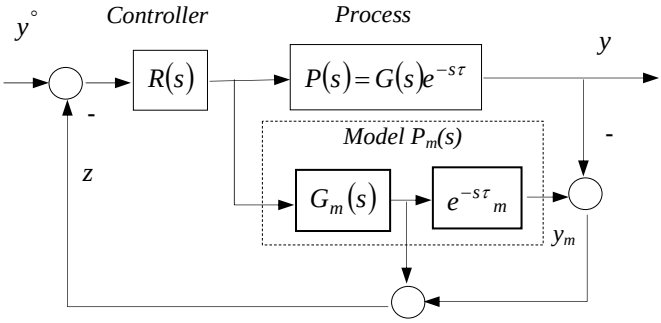


Fig. 2. The structure for Smith predictor.

This algorithm requires a minimal knowledge of the process to describe it through a transfer function (model):

$$P_m(s) = G_m(s)e^{-s\tau_m} \quad (1)$$

As shown in Figure 2, the feedback is closed not on the process value *y*, but on the *z* variable, which has the same value that *y* had τ_m seconds earlier, and therefore it is in some ways a „*prediction*" of the measure; that is why this control architecture is called Smith „*predictor*" [2].

IV. The Model of Sepsis Evolution

The molecular or cellular network is modeled as a collection of nonlinear differential equations, where reaction rates and compound concentrations are the variables [3]. Brause uses a reduced order approximation model, with three variables [4].

- $P(t)=P$ representing the pathogen influence, $P \in [0,1]$
- $M(t)=M$ representing the immune response, namely the macrophage action, $M \in [0,1]$, and
- $D(t)=D$ representing the percent of damaged noble cell tissue, which is destroyed in the fight between *P* and *M*, $D \in [0,1]$.

The equations of the mathematical model are [3]:

$$\dot{P} = \alpha_1 P(1-P) - \alpha_2 MP\ ,\ \alpha_i > 0, i = \overline{1,2} \quad (2)$$

$$\dot{M} = -\beta_1 M + M(1-M)(\beta_2 P + \beta_3 D)\ ,\ \beta_i > 0, i = \overline{1,3} \quad (3)$$

$$\dot{D} = -\gamma_1 D + \gamma_2 h((M-\theta)/\gamma_3)\ ,\ \gamma_i > 0, i = \overline{1,3} \quad (4)$$

where $\theta = 0.5$ is a threshold value and

$$h(x) = 1/[1+\exp(-x)] \quad (5)$$

A typical parameter regime takes the next values:

$$\begin{aligned} &\alpha_1 = 0.1; \alpha_2 = 1.0; \\ &\beta_1 = 1.0; \beta_2 = 10.0; \beta_3 = 1.0; \\ &\gamma_1 = 0.1; \gamma_2 = 0.04; \gamma_3 = 0.25; \end{aligned} \quad (6)$$

The dynamical model described by equation (2, 3 and 4) reflects some basic qualitative features of the sepsis phenomenon and assume that the rate of cells damage increases also with a sigmoid function (5) of macrophages action, limited by a threshold θ.

In the application developed by the authors, we have used a linearized version of the mathematical model described in equations (7, 8 and 9), accomplishing through this, a compatibility with the Smith predictor.

$$\dot{\Delta P} = [\alpha_1(1-2P_0) - \alpha_2 M_0]\Delta P - \alpha_2 \Delta M \quad (7)$$

$$\begin{aligned} \dot{\Delta M} &= \beta_2 M_0(1-M_0)\Delta P + \\ &+ [(\beta_2 P_0 + \beta_3 D_0)(1-2M_0) - \beta_1]\Delta M - \beta_3 M_0 \Delta D \end{aligned} \quad (8)$$

$$\dot{\Delta D} = \frac{\gamma_2}{\gamma_3}\exp[-(M_0-\theta)/\gamma_3]\Delta M - \gamma_1 \Delta D \quad (9)$$

where $P_0 = 0.2$, $M_0 = 0.15$ and $D_0 = 0.1$ are the steady state values calculated in [3].

Dobrescu, V. E. Oltean and Popescu make an important change in the initial Brause model. They have introduced a new parameter *T* that signifies the initialization of a treatment (medication procedure) [3]. Sepsis treatment can be modelled by introducing an exogenous signal into the right hand term of (2):

$$\dot{P} = \alpha_1 P(1-P) - \alpha_2 MP - \alpha_3 T\ ,\ \alpha_i > 0, i = \overline{1,3} \quad (10)$$

Medication is carried, with the help of the circulatory system. This induces a dead-time that has to be taken into account for when, the amount of active substance, required for combating infected cells, is decided. Also, the authors have taken into consideration the existence of a dead-time between the evolution of sepsis and the body's immune response. The modified mathematical model is:

$$\begin{aligned} \dot{\Delta P}(t) &= [\alpha_1(1-2P_0) - \alpha_2 M_0]\Delta P(t) - \\ &- \alpha_2 \Delta M(t-\tau_1) - \alpha_3 T(t-\tau_2) \end{aligned} \quad (11)$$

$$\begin{aligned} \dot{\Delta M}(t) &= \beta_2 M_0(1-M_0)\Delta P(t) + \\ &+ [(\beta_2 P_0 + \beta_3 D_0)(1-2M_0) - \beta_1]\Delta M(t) - \\ &- \beta_3 M_0 \Delta D(t) \end{aligned} \quad (12)$$

$$\begin{aligned} \dot{\Delta D}(t) &= \frac{\gamma_2}{\gamma_3}\exp[-(M_0-\theta)/\gamma_3]\Delta M(t-\tau_1) - \\ &- \gamma_1 \Delta D(t) \end{aligned} \quad (13)$$

V. Cardiovascular System Structure

The anatomical structure of the cardio-vascular system is presented in fig. 3. Due to the structure of the circulatory system, we have a time delay before the medication reaches a homogeneous concentration in the body. This dead time is put in evidence by the simplified mathematical sintetized by K. Minato, M. Kuwahara and Y. Yonekura [5], [6].

The right heart has an atrium (A_R) and a ventricle (V_R). It receives the blood from the body and pumps it in the lung. The left heart has an atrium (A_L) and a ventricle (V_L). It receives the blood from the lung and pumps it in the body. In fig. 3 the notations have the next significances:

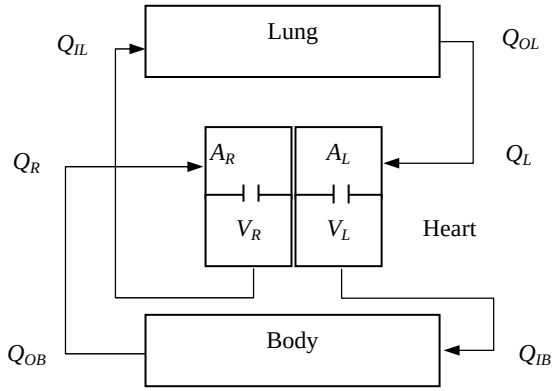


Fig. 3. The simplified structure of the cardiovascular system.

- Q_L – represent the blood flow input in the left heart.
- Q_{IB} - represent the blood flow input in the body.
- Q_{OB} - represent the blood flow output in the body.
- Q_R - represent the blood flow input in the right heart.
- Q_{IL} - represent the blood flow input in the lung.
- Q_{OL} - represent the blood flow output in the lung.

Also, the blood flows Q [mls^{-1}] between the different compartments are considered equals and constants.

$$Q_R = Q_{IL} = Q_{OL} = Q_L = Q_{IB} = Q_{OB} = Q = ct. \quad (14)$$

On suppose that the cardio-vascular system is working in steady state and we consider the supposition that all the involved blood volumes are constant.

- W_R [ml] - represent the blood volume in the right heart;
- W_L [ml] - represent the blood volume in the left heart;
- W_{lung} [ml] - represent the blood volume in the lung;
- W_{body} [ml] - represent the blood volume in the body;

We have modeled the injection of the pharmacological substance in a peripheral vein by the relation:

$$W_i \frac{dc_i(t)}{dt} = i(t) - Q_i c_i(t), \quad c_i(0) = 0 \quad (15)$$

where W_i represent the volume of the substance, $c_i(t)$ represent the substance concentration at the injection level and Q_i is the blood flow which receive and transport the substance. Also:

$$i(t) = \begin{cases} I / \tau_i, \text{ for } 0 \le t \le \tau_i \\ 0, \text{ for t } > \tau_i \end{cases} \quad (16)$$

where

- τ_i [s] - represents the interval of injection
- I [mg] – represents the quantity of substance in the unwashed solution.

The equation which describe the transport from the injection place to the right heart is:

$$W_R \frac{dc_R(t)}{dt} = Q_i c_i(t - \tau_i) + Q c_{body}(t - \tau_{body}) - Q c_R(t) \quad (17)$$

where τ_{body} represents the delay necessary for the transport of the injectable solution and $c_L(t)$, $c_R(t)$, $c_{body}(t)$ represent respectively, the substance concentrations at the left heart, right heart and at body level. Similarly, the equations for the transport in the next compartments are:

- for the lung:

$$W_{lung} \frac{dc_{lung}(t)}{dt} = Q c_R(t) - Q c_{lung}(t) \quad (18)$$

- for the left heart:

$$W_L \frac{dc_L(t)}{dt} = Q c_{lung}(t - \tau_{lung}) - Q c_L(t) \quad (19)$$

- for the body:

$$W_{body} \frac{dc_{body}(t)}{dt} = Q c_L(t) - Q c_{body}(t) \quad (20)$$

From (15), (17), (18), (19) and (20) results:

$$\dot{c}_i(t) = -\frac{1}{T_i} c_i(t) + \frac{1}{W_i} i(t) \quad (21)$$

$$\dot{c}_R(t) = \frac{1}{T_{Ri}} c_i(t - \tau_t) - \frac{1}{T_R} c_R(t) + \frac{1}{T_R} c_{body}(t - \tau_{body}) \quad (22)$$

$$\dot{c}_{lung}(t) = \frac{1}{T_{lung}} c_R(t) - \frac{1}{T_{lung}} c_{lung}(t) \quad (15)$$

$$\dot{c}_L(t) = \frac{1}{T_L} c_{lung}(t - \tau_{lung}) - \frac{1}{T_L} c_L(t) \quad (23)$$

$$\dot{c}_{body}(t) = \frac{1}{T_{body}} c_L(t) - \frac{1}{T_{body}} c_{body}(t) \quad (24)$$

where

$$T_i = W_i / Q_i, \qquad T_{Ri} = W_R / Q_i,$$
$$T_R = W_R / Q, \qquad T_{lung} = W_{lung} / Q,$$
$$T_L = W_L / Q, \qquad T_{body} = W_{body} / Q$$

represent time constants. Using Laplace transformation (we suppose that the initial conditions are zero) it is possible to calculate the transfer function for each channel of process:

- for the injection zone:

$$H_i(s) = \frac{c_i(s)}{i(s)} = \frac{1/Q_i}{T_i s + 1} \quad (18)$$

- for the right heart:

$$c_R(s) = \frac{e^{-\tau_i s} T_R}{T_{Ri}(T_R s + 1)} c_i(s) + e^{-\tau_{body} s} \frac{1-k}{T_R s + 1} c_{body}(s) \quad (19)$$

- for the lung:

$$H_{lung}(s) = \frac{c_{lung}(s)}{c_R(s)} = \frac{1}{T_{lung} s + 1} \quad (20)$$

- for the left heart:

$$H_L(s) = \frac{c_L(s)}{c_{body}(s)} = e^{-\tau_{body} s} \frac{1}{T_L s + 1} \quad (21)$$

- for the body:

$$H_{body}(s) = \frac{c_{body}(s)}{c_L(s)} = \frac{1}{T_{body} s + 1} \quad (22)$$

During the simulation it is possible to modify the value of concentration of the substance and time of injection. Also, we can choose physiological or pathological values for all the parameters of the cardiovascular system.

So, it is possible to simulate the dilution of drugs administrated in the cardiovascular system. The results of simulation are presented in the next figures.

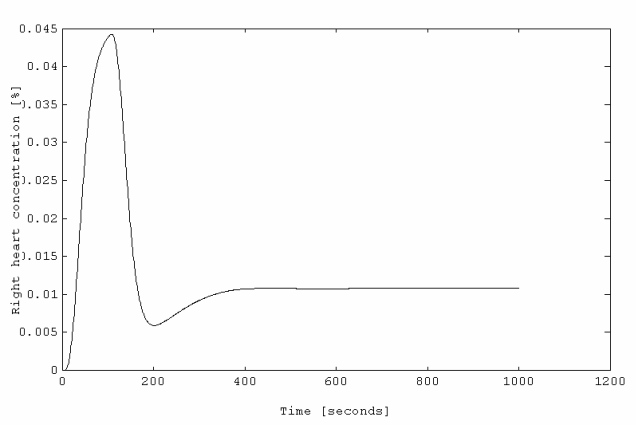


Fig. 4. The time evolution concentration at the right heart.

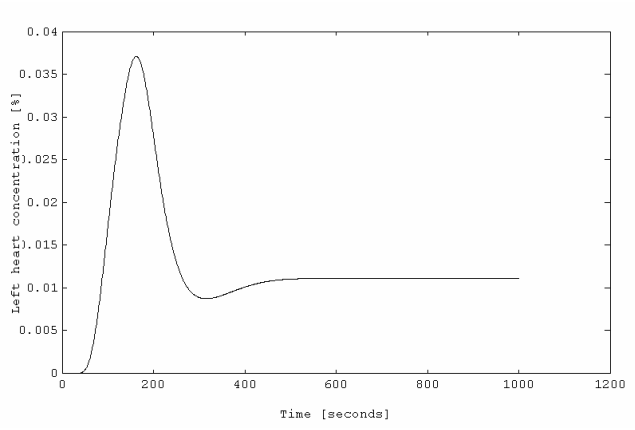


Fig. 5. The time evolution concentration at the left heart.

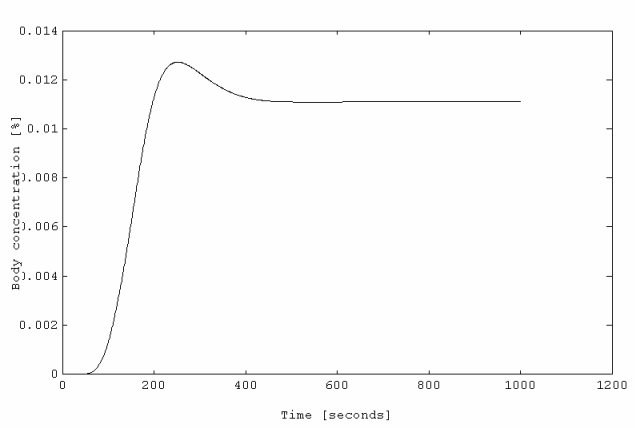


Fig. 6. The time evolution concentration at the body level.

In the following figures we have represented:

- $P(t)$ representing the pathogen influence (Fig. 7);
- $M(t)$ representing the macrophage action (Fig. 8);
- $D(t)$ representing the percentage of damaged cell tissue, which is destroyed in the fight between P and M (Fig. 9).

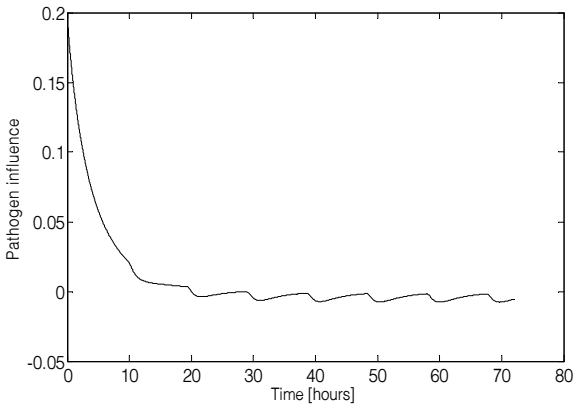


Fig. 7. The pathogen influence.

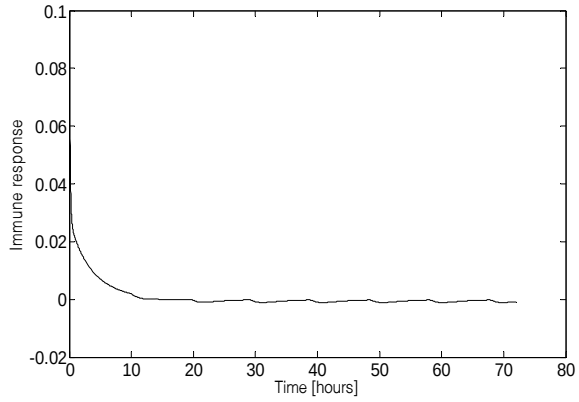


Fig. 8. The macrophage action.

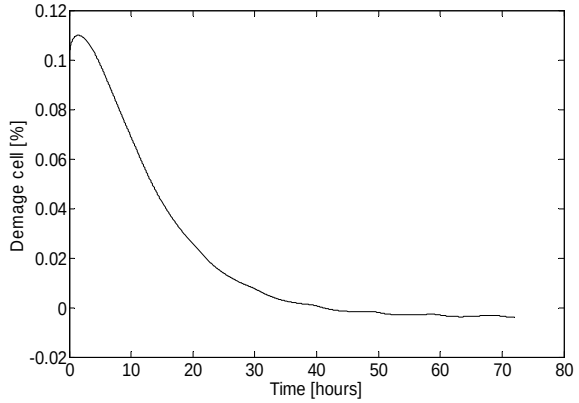


Fig. 9. The percentage of damaged cell.

A special situation occurs in intensive care units (ICUs), where most of the patients are not aware. Usually, sepsis represent the systemic inflammatory response syndrome (SIRS) associated with infection.

The most critical aggravation of sepsis is the septic shock. In ICUs the septic shock is a very critical situation of the patient. The diagnosis of septic shock is still made too late, because at present there are no adequate tools to predict the progression of sepsis to septic shock [7].

The advantage of the ICUs is the fact that medication can be fed constantly, in a controlled environment. In order to keep the concentration of active substance and control and avoid over-doses, we are proposing he use of a control structure based on the Smith predictor.

VI. The Structure Proposed for Sepsis Control

The control structure proposed by the authors takes into account dead times introduced by the cardiovascular system. Specifically, we propose that the mathematical model describing the sepsis process control is formed by the connection with the mathematical model of the cardiovascular system. The controller which will command the pump for the medication substance, will be synthesized using Smith predictors. We therefore ensure a good behavior of the control system. The command required for the pump is anticipated ahead of time with a period equal with dead time. It also improves behavior in relation to disturbance. It is worth mentioning that in the synthesis of the control law, we must consider the ensuring of closed-loop stability.

Acknowledgment

This paper is part of the project *Models and morphometric techniques with applications in improving image-based diagnosis in gastrointestinal cancers and evaluating new therapies - IMAGO*, at the University of Politehnica of Bucuresti and was supported by National Center for Programme Management - CNMP, Romania.

References

[1] O. J. Smith, “A controller to overcame dead time”, *ISA Journal*, Vol. 6, No.2, 1959, pp. 28-33.

[2] Veronessi M., “Performance improvement of Smith predictor through automatic computation of dead time”, *Yokogawa Technical Report English Edition*, No. 35, 2003.

[3] R. Dobrescu, V. E. Oltean, D. Popescu, “Adaptive hybrid model of the immune system response in sepsis”, (Published Conference Proceedings style) *Proceedings of the International Symposium on Systems Theory, Automation, Robotics, Computers, Informatics, Electronics and Instrumentation SINTES 13*, Craiova, 2007, ISBN 978-973-742-816-5.

[4] R. Brause, “Adaptive modelling of biochemical pathways”, (Published Conference Proceedings style) *Proc. of the 15th Int. Conf. on Tools with Artificial Intelligence – ICTAI* 2003, IEEE Computer Society, 2003.

[5] Minato, K., M. Kuwahara, Y. Yonekura, A. Hirakawa, “Parameter Estimation of Radiocardiogram Using a Minicomputer”, *Automatica*, vol.15, 1979, pp. 521 - 529.

[6] I. Iancu, E. Iancu, “Mathematical Modelling in Human Physiology. Application in Diagnosis”, (Published Conference Proceedings style) *International Symposium on System Theory, Automation, Computers and Electronics, SINTES 10*, Craiova, Romania, 2000, pp. A38 – A41.

[7] A. Seely, N. Christou, “Multiple Organ Dysfunction Syndrome, Exploring the Paradigm of Complex Nonlinear Systems”. *Crit. Care Med.* 28(7), 2000, pp. 2193-2200.

Combined Technologies for Fuel Economy Improvement on Hybrid Vehicles

Radu Dobrescu, *Senior Member, IEEE,* Ecaterina Virginia Oltean, and Dan Popescu, *Member, IEEE*

Abstract— **In this paper, two advanced technologies (hybrid vehicles and sensor networks) are combined in order to obtain fuel economy over common urban drive cycles. First, the performance of a hybrid powertrain is analyzed by simulation in Matlab (using ADVISOR program) and compared with performance characteristics of a baseline vehicle. Then the advantages of the telematics capability to extract information about traffic with a sensor network are presented. As a final investigation, the potential for using feedforward information from a sensor network with a hybrid drivetrain is discussed.**

I. Introduction

Hybrid electric powertrains have established a presence in the market place primarily based on the promise of fuel savings through the use of an electric motor in place of the internal combustion engine during different stages of driving. However, these fuel savings associated with hybrid vehicle operation come at the tradeoff of a significantly increased initial vehicle cost due to the increased complexity of the powertrain. On the other hand, telematics-enabled vehicles may use a relatively cheap sensor network to develop information about the traffic environment in which they are operating, and subsequently adjust their drive cycle to improve fuel economy based on this information – thereby representing 'intelligent' use of existing powertrain technology to reduce fuel consumption.

As well as the technology changes internal to the vehicle, the telematics revolution of the past decade has generated the possibility for a vehicle to communicate with the road infrastructure and other vehicles to obtain greater information about the traffic environment in which it is operating. Systems such as the PATH program [1], [2] have demonstrated that platooning of vehicles in an Automated Highway System can lead to increased driver safety, decreased road congestion and improved fuel economy. With fuel consumption in urban environments up to 50% higher than during highway driving, there are even greater possibilities in addressing this area of operation. As a result, a first-generation 'intelligent' vehicle operating in an urban environment could be envisaged as providing a driver aid based on acquired information about some degree of traffic flow from the environment.

This work was supported in part by the Ministry of Education and Research under Grant PN2 12100/2008

The authors are with the POLITEHNICA University of Bucharest, Faculty of Control and Computers, Splaiul Independentei 313, Bucharest 7100, Romania (telephone: +40214029105, e-mail: radud @isis.pub.ro).

Incorporation of telematics providing the information between vehicles over a dedicated radio bandwidth would not only address this issue in the long term, but also provide information to the vehicle about traffic flow over a larger distance than if each vehicle were operating completely autonomously. To obtain even greater information to the vehicle over a longer look ahead distance it is most likely that some form of communication between the infrastructure and the vehicle is required.

II. Related Work on Fuel Optimization using Hybrid Powertrain

Since the 1990s, several automobile industries have started developing hybrid electric vehicles (HEVs). This is due to the drawbacks inherent in conventional internal combustion engine (ICE) vehicles as well as electric vehicles (EVs). In fact, the exhaust emissions of ICE vehicles are the major source of urban pollution and should be replaced by clean and efficient vehicles. Although EVs with no emissions attracted the attention of many in the car industry for a few decades, many problems, including short driving distance, long recharging time of batteries, and high costs, gave strong motivation to develop HEVs.

There are two kinds of HEV, series and parallel. In series HEVs, the powertrain, as in EVs, is provided by an electric motor. There is also a small ICE, which charges the batteries when the state of charge is below a certain level. In parallel HEVs, which are the most popular and are considered in this paper, the powertrain is provided by both an ICE and an electric motor that are coupled together mechanically. One important issue in parallel HEVs is the amount of torque produced by these powertrain sources in different driving conditions. The contributions of the ICE and the electric motor in different driving conditions are defined as follows [3]:

- when the required torque is more than the maximum torque of ICE, the electric motor compensates for the extra torque
- below a certain velocity of the vehicle or when the vehicle stands still, the electric motor provides the entire required torque, since ICEs have low efficiency in these situations
- when the state of charge of the batteries is low, in addition to driving the vehicle, the ICE provides the required energy for charging the batteries. In this case, the electric motor works as a generator and charges the batteries.

The operating style of each source (i.e. ICE or electric motor) and the amount of their contribution in producing the required torque at any time is determined by a controller. Moreover, this controller should minimize the fuel consumption. However, minimizing the fuel consumption could lead to considerable torque reduction, which may not yield very pleasant driving conditions. Therefore, in addition to fuel minimization, it is sometimes necessary to maximize the torque as well.

There are many solutions proposed for fuel minimization, one of the first being that of Powell *et al.* which have used a combination of several PID controllers, one for every section of the vehicle, due to the highly nonlinear system [4]. Lyshevski has derived a nonlinear control method based on the Lyapunov stability theory [5]. Lin *et al*. have employed dynamic programming, with a sequential decision making method, to minimize the fuel consumption of hybrid vehicles [6]. Delprat *et al.* have used classical optimal control methods to achieve minimum fuel consumption for a given speed cycle [7]. Optimal control methods usually require a precise model of the system. Schouten *et al*. have designed a fuzzy controller with nine and 27 rules, respectively, to control hybrid vehicles [8]. Ippolito *et al*. have used fuzzy c-means, along with genetic algorithms, for power-flow management (e.g. the contribution of the electric motor and ICE) in different driving cycles of hybrid electric vehicles [9].

The previous work on torque maximization is less representative, because always it leads to high fuel consumption. Lee *et al.* have used fuzzy systems, a fuzzy predictive controller with nine rules for converting the driver's commands to appropriate torques [10]. Won and Langari propose an intelligent energy management agent for optimal torque distribution on HEV [11]. In a recent survey Murphey overviews the progress of vehicle power management technologies that shape the modern automobile [12].

As far as the authors of this paper are aware, there are few papers in the literature which have considered both fuel minimisation and ICE torque maximisation. One way to maximise the output torque of parallel HEVs is to maximise the output torque of the ICE [13]. This is mainly due to the fact that parallel HEVs are designed in such a way that when the driver demands high torque, the ICE provides its maximum torque; the remaining required torque is provided by the electric motor.

One goal of this paper is to find a compromise solution between fuel consumption and torque production by designing an adaptive controller. The adaptation process is performed online, which makes the closed-loop system more robust against disturbances and changes in the system parameters. The other goal is to obtain fuel economy by adapting the fuel consumption to the characteristics of the traffic flow.

III. Models and Simulations

A. Description of the simulation environment

In simulations, ADVISOR software has been employed. This is powerful software for the simulation of all kinds of vehicles, such as conventional, electrical, series hybrid and parallel hybrid. This software operates in a MATLAB/SIMULINK environment. ADVISOR consists of several blocks, each representing a particular part of a vehicle, such as the engine, electric motor, batteries, gearbox, clutch, controllers and exhaust system. The models of different parts of vehicles in ADVISOR are quasi-static. That is, data have been collected in steady state (e.g. constant torques and speeds), and then have been corrected for transient effects such as the rotational inertia of drive train components. Also, different driving cycles can be defined in this software.

In order to develop a fair comparison between the technologies, a model (for urban driving) hybrid vehicle that matches the performance characteristics of a baseline vehicle is used. A parallel hybrid configuration was assumed for modelling the hybrid electric vehicle in this study. This configuration consists of an electric motor (EM) and internal combustion engine (ICE) that can simultaneously or individually drive the transmission (and subsequently propel the vehicle). The split is determined by the vehicle's hybrid control strategy, and in this case a simple default strategy was used which (subject to constraints on the battery state of charge) uses the EM for slow city driving condition and assists the engine for peak acceleration, hill climbing, and extremely fast highway driving conditions. Furthermore, the EM can act in reverse mode to become a generator during regenerative braking and consequently used to recharge the batteries.

TABLE I
CONVENTIONAL VEHICLE MODEL SPECIFICATIONS

Characteristics	Standard values
Total weight	1642 kg
Chassis weight	1000 kg
Vehicle length	5.00 m
Centre of mass height	0.5 m
Transmission Manual	5 speed
Transmission efficiency	95%
Final drive ratio	2.92
Gear ratios	3.5:2.14:1.39:1:0.78
Tyre rolling radius	0.314 m
Tyre pressure	240 kPa

The baseline vehicle chosen for this study was a conventional family sedan. The main characteristics are shown in Table I. It is important to note that the simulation

essentially works in a reverse direction to what happens in the real world – that is, the drive cycle is the input to the vehicle model, and the required changes to the vehicle speed are calculated based on the drive cycle. This change in vehicle speed is then converted to engine speed and torque requirements by taking into account the current gear ratio (a shifting map is supplied to the model) and the efficiencies of the transmission. The fuel use is then calculated from a look-up table of fuel rate against engine operating point (defined by engine speed and torque).

The fuel use map for the vehicle model as a function of operating point was adapted from steady state maps provided in Samuel *et al.* [14] and is illustrated in Fig. 1.

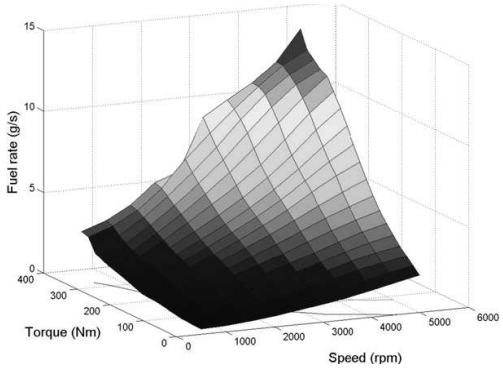


Figure 1. Fuel consumption map of the test vehicle as a function of operating point

In order to accurately compare fuel economy with the conventional powertrain vehicle presented above, the hybrid vehicle dimensions and properties should be maintained, with the obvious exception of the internal combustion engine being downsized and replaced by a hybrid drivetrain. The necessary sizing of the hybrid vehicle's internal combustion engine is directly coupled to the size of the electric motor used, and hence an optimization process is required to ensure that the configuration is capable of meeting the performance requirements of the vehicle but in doing so uses the minimum amount of fuel. The performance requirements of the conventional drivetrain were set as constraints on the optimization process are listed in Table II.

TABLE II
CONSTRAINTS IMPOSED UPON HYBRID VEHICLE

Parameters	Performance
Acceleration 0–100 km/h	68.5 s
Maximum acceleration	3.8 m/s^2
Maximum speed	210 km/h
Gradeability*	>15.5%
Battery state of charge**	<0.5%

* Sustain 88.5 km/h for 82 s towing 1000 kg load
**Difference between initial and final battery state of charge

Another constraint during the optimization process is the change in state of battery charge at the beginning and end of the cycle to prevent misleading fuel economy results arising from excessive use of the electric motor (which would then require battery replenishment and subsequently increased fuel use the next time the vehicle is run).

Once identified the constraints to be placed on the hybrid vehicle, the next point to consider is the type of electric motor and internal combustion engine to use as base models for scaling. After consideration of an earlier study [15] it was decided that a 1.3 l, 71 kW turbocharged GM Opel engine would be a suitable starting point for the hybrid's ICE. This choice was motivated by the result that after turbocharging a small engine, the output power is increased by 30–40% relative to the non-turbocharged engine of the same size. This subsequently allows a smaller engine and hence lower frictional losses in comparison to a larger engine supplying the same amount of power as the turbocharged one. Furthermore, by using a smaller capacity engine, the weight can be reduced. A 49 kW Honda permanent magnet brushless DC motor was chosen as the base motor to be scaled and the electric power was provided through NiMH battery modules.

As a result there are three factors that can influence the total performance of the baseline hybrid vehicle – the scaling of the internal combustion engine (ICE), the scaling of the motor and the number of battery modules used. Naturally increasing the power output of the powertrain components and the number of batteries also increases their weight, and hence will impact on fuel economy. In order to get the best performance of the hybrid vehicle, the engine configuration was optimized over a given drive cycle in order to find the best motor size, number of battery modules and engine size.

B. Designing the adaptive controller

The controller has two inputs: the load torque and the state of charge (SOC) of the battery pack. The ICE operation point (i.e. the throttle angle of engine) is adjusted based on these inputs. The desired electric motor torque T_{EM_Des} can be calculated as

$$T_{EM\_Des} = T_{load} - T_{ICE\_Set} \tag{1}$$

where T_{load} is the load torque, and T_{ICE_Set} is the desired torque of the ICE, defined by the controller. In order to achieve a compromise solution between three goals (namely, maximizing fuel economy, reducing vehicle output emissions, and maintaining acceptable powertrain performance by maximizing the vehicle output torque), an adaptive controller will be designed. The adaptation law for changing weights is that which minimize the error function *e(k)*:

$$e(k) = \alpha_1(|T_{MAX} - T|) + \alpha_2(|fuel_{MIN} - fuel|) \tag{2}$$

where *fuel* is the consumed fuel by ICE per time unit, *T* is the output torque of ICE, $fuel_{MIN}$ is the desired fuel

consumption designated for fuel minimization, T_{MAX} is the desired output torque of ICE, chosen to maximize the output torque, and α_1 and α_2 are the weighting factors for output torque and fuel consumption, respectively.

The block diagram of the proposed method is presented in Figure 2.

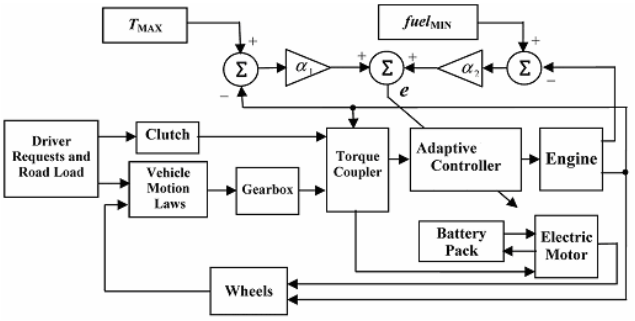


Figure 2. Block diagram of the proposed controller for the parallel HEV.

C. Simulations results

In ADVISOR, there are two different control methods: i) fuel minimization controller for minimizing fuel consumption; ii) efficiency-mode controller for maximizing output torque [16].

The simulation results of the proposed controller will be compared with these two benchmark methods. In the simulation two weighting factors were used for the adaptive (α_1 and α_2). In the simulations, firstly it is assumed that $\alpha_1=\alpha_2=1$. That is, both fuel minimization and torque maximization are equally important for the driver. For further work, another two scenarios will be tested: only fuel minimization (i.e. $\alpha_1=0$ and $\alpha_2=1$) and no fuel minimization (i.e. $\alpha_1=1$ and $\alpha_2=0$), which stands for the output torque maximization of ICE.

The optimization process now represents a constrained multidimensional problem solved using the approach described in Manzie *et al.* [17]. It was found that with more urban driving conducted in the drive cycle then a larger electric motor could provide more assistance in reducing fuel consumption. The fuel use map of the scaled engine is shown in Fig. 3.

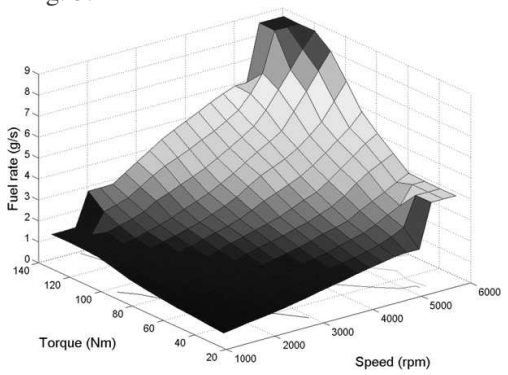


Figure 3. Fuel consumption map of the turbocharged engine used in hybrid powertrain

It is worth noting that the optimized configuration corresponds to an almost equal split in power between the electric motor and the internal combustion engine, which agrees with the hybrid configurations used in urban cycles.

Having obtained an optimized hybrid configuration, this vehicle was then simulated through urban drive cycles with the restriction that the state of charge of the battery at the end of the drive cycle must be within 0.5% of the state of charge at the beginning. An example of the desired trajectory for vehicle speed in an urban cycle is shown in Figure 4. In this driving cycle, at the instant of t=20 s, greater acceleration is requested by the driver, followed by pressing the brake pedal at t=295 s. In this cycle, the vehicle covers a distance of 6.4 km in 329 s. The error between the desired speed and the actual speed will be converted to the desired torque using a PI controller. This difference must be compensated for by the vehicle powertrain system.

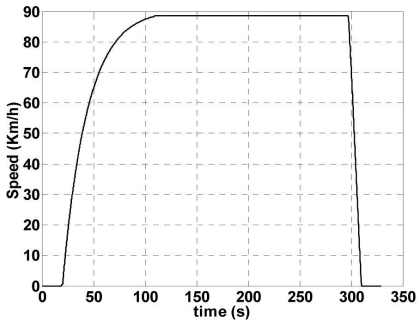


Figure 4. Speed profile in an urban cycle

The results are provided in Table III, and demonstrate an average 20% fuel economy improvement across the drive cycles achieved through hybridization.

However, the concentration on urban drive cycles in the optimization procedure results in an almost equal ratio between the electric motor and internal combustion engine power, which is unlikely to occur in production for a family sedan of this size. Despite these limitations, Table III does allow a benchmark to be set against which the fuel economy of an intelligent vehicle can be measured.

TABLE III
FUEL ECONOMIES (L/100 KM) FOR CONVENTIONAL AND OPTIMIZED HYBRID VEHICLES

Drive cycle	Conventional drivetrain	Hybrid drivetrain	Improvement with hybrid (%)
Low	10.2	8.1	20
Moderate	10.8	8.2	24
Heavy	11.9	10.1	16

IV. Telematics Role in Intelligent Vehicle Control

A. Conventional powertrain

One of the most significant contributors to fuel use in an urban environment is the stop start behaviour of traffic flow. Through the use of telematics, a given vehicle can be made aware of the traffic and infrastructure in which it is operating and adjust the driving condition en route.

For the intelligent vehicle in this paper it is assumed that there exists a sensor network potentially incorporating inter-vehicle communication, radar and laser technologies that can be used to convey information about the surrounding traffic. This traffic preview information can then be used to adjust the vehicle's instantaneous velocity, whilst arriving at the destination at he same time as an un-equipped vehicle. Then an algorithm that outlines how the intelligent vehicle may utilize the feedforward traffic information to adjust its own speed to minimize vehicle accelerations is described. It is assumed that the output of the sensor network is assumed to be previewed traffic velocity information, $v_p(t)$, which is available up to T_p seconds ahead of the current time. This allows the position of a vehicle subjected to this traffic flow to be predicted based on its current position, $x(t)$.

Estimate vehicle position at time T_p in the future according to:

$$\hat{x}(t+T_p) = x(t) + \alpha \sum_{i=1}^{T_p/\Delta} v_p(t+i\Delta)\Delta \qquad (3)$$

where Δ is the sampling period (considered uniform) and α is a conversion constant. In order to reach this predicted position with minimum stop–start behaviour the intelligent vehicle should attempt to use a constant speed, $v_{int}(t)$, calculated as follows:

$$v_{\text{int}}(t) = \frac{\hat{x}(t+T_p) - x(t)}{\alpha T_p} \qquad (4)$$

This process, using Eqs. (3) and (4), is repeated every time new information becomes available. In the drive cycles used in this study, this period corresponds to once per second.

It is important to note that a critical assumption in this algorithm is that fuel use is linearly proportional to engine speed for a given torque requirement. Examination of the fuel use maps indicates this is a reasonable assumption for relatively small changes in engine speed providing the current gear is maintained. In the event that there was a highly non-linear relationship between fuel use and engine speed, it may be necessary to consider engine-dependent strategies.

While significant improvements in fuel economy through the use of feedforward information about traffic conditions supplied by a sensor network were demonstrated, it is unlikely that there would be a quantum shift to this technology by all road users. To gauge the impact of the technology using low levels of consumer uptake, it is necessary to remove the assumption that the intelligent vehicle's velocity (and position) can be assigned irrespective of the traffic flow in which it operates. For example, the adapted velocity profiles obtained using the velocity modification algorithm demonstrate a phase lead at certain points in time. This indicates the intelligent vehicle starts moving before it would have if no preview information were available. Naturally, there will be some cases whereby this situation is not feasible, one example is a traffic signal enforcing vehicle stationarity, and hence it is important to reflect this by applying constraints to the velocity algorithm. The constraint applied in this paper is that the intelligent vehicle cannot overtake the position it would occupy on the road if it were traveling without any preview information, i.e. it cannot overtake the un-intelligent vehicle position, or equivalently an unintelligent vehicle directly in front of the intelligent vehicle and traveling at a velocity indicated by the drive cycle.

B. Hybrid powertrain

An 'intelligent hybrid' is a vehicle with a hybrid drivetrain that is equipped with telematics enabling some feedforward information about speed trajectories to be obtained. In order to enforce a change in the state of charge of less than 0.5% between the start and end of the drive cycle, the default hybrid control strategy used and the initial state of charge varied until the constraint was met.

The resulting fuel economy for the intelligent hybrid using both intelligent vehicle velocity modification algorithms (overtaking allowed and not allowed) over each of the three drive cycles are illustrated in Fig. 5 (a – low traffic, b – moderate traffic, c – heavy traffic).

From the plots shown in Fig. 5, only limited conclusions can be drawn. Firstly, it is noted that there is a general trend in decreasing fuel use with increasing information seen for all three drive cycles. However, this decrease in fuel use is non-monotonic, and this non-monotonicity is due to several factors.

The primary one is that the optimum fuel use will not depend solely on minimizing the vehicle speed deviations, since decelerations are required to use regenerative braking. This is why hybrid vehicles are significantly better then conventional drivetrains in terms of fuel economy during urban drive cycles, but the level of improvement decreases during highway driving. The second is that the minimization of fuel also depends on the individual efficiency maps of the internal combustion engine and electric motor, as well as the switching strategy of the hybrid controller.

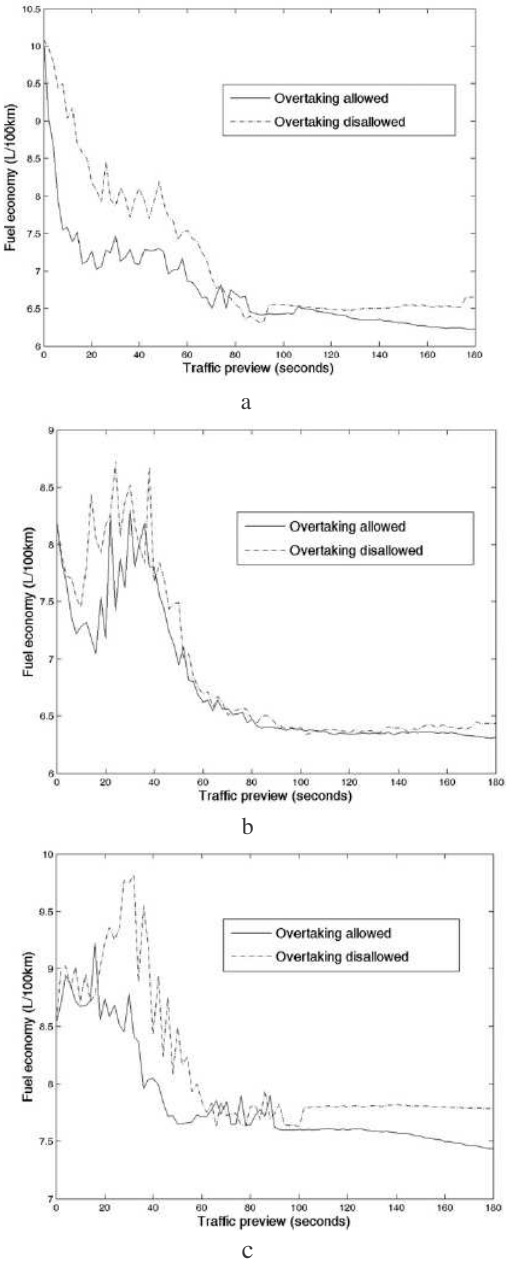


Figure 5. Fuel economy of intelligent hybrid vehicle as a function of preview time

V. CONCLUSION

This paper presents a comparison between two of the emerging technologies in automotive systems, hybrid drivetrains and telematics capability. Following the development of an optimal hybrid configuration that matches the performance of the baseline test vehicle, it was found through simulation that the fuel economy improvements possible through optimal hybridization ranged between 15% and 25% relative to the baseline vehicle.

Two technologies (hybrid and telematics) were combined in one vehicle model to create an 'intelligent hybrid' vehicle model. While there were general trends indicating improvement in fuel economy with traffic preview, the multi-dimensional nature of the problem (current vehicle speed, power split between the electric motor and internal combustion engine and battery state of charge must all be considered) ensures that the optimal use of the feedforward information, particularly for short traffic previews, remains an ongoing research problem.

REFERENCES

[1] R. Rajamani, H.-S.Tan, B.K Law, and W.-B. Zhang, "Demonstration of Integrated Longitudinal and Lateral Control for the Operation of Automated Vehicles in Platoons", *IEEE Transactions on Control Systems Technology*, 8, pp. 695–708, 2000

[2] R. Bishop, *Intelligent Vehicle Technology And Trends,* Artech House, 2005

[3] I. Husain, *Electric and Hybrid Vehicles: Design Fundamentals,* CRC Press, 2003

[4] B. K. Powell, K. E. Bailey, and S. R. Cikanek "Dynamic Modelling and Control of Hybrid Electric Vehicle Powertrain Systems," *IEEE Control Syst. Mag.*, vol. 18, No. 5, pp.17-33, 2008

[5] S.E. Lyshevski, "Diesel-electric drivetrains for hybrid-electric vehicles: new challenging problems in multivariable analysis and control", *IEEE International Conference on Control Applications*, Vol. 1, pp.840-845, 1999

[6] C.C Lin, J.M. Kang, J.W. Grizzlq, and H. Peng, "Energy management strategy for a parallel hybrid electric truck", *American Control Conference*, Vol. 4, pp.2878-2883, 2001

[7] S. Delprat, T.M. Guerra, and J. Rimaux, "Control strategies for hybrid vehicles: optimal control", *IEEE Vehicular Technology Conference*, Vol. 3, pp.1681-1685, 2002

[8] N.J. Schouten, M.A. Salman,. and N.A. Kheir, "Energy management strategies for parallel hybrid vehicles using fuzzy logic", *Control Engineering Practice*, Vol. 11, pp.171-177, 2003

[9] L. Ippolito, V. Loia, and P. Siano, "Extended fuzzy c-means and genetic algorithms to optimize power flow management in hybrid electric vehicles", *IEEE Conference on Control Applications*, Vol. 1, pp.115-119, 2003

[10] H.D. Lee, E.S. Koo, S. Sul, J. Kim, M. Kamiya, H. Ikeda, S, Shinohara, and H.Yoshida, "Torque control strategy for a parallel-hybrid vehicle using fuzzy logic", *IEEE Industry Applications Magazine*, Vol. 6, No. 6, pp.33-38, 2000

[11] J-S. Won, R. Langari, "Intelligent energy management agent for a parallel hybrid vehicle," *IEEE Transactions on Vehicular Technology*, volume 54, issue 3, pp. 935 – 953, 2005

[12] Y.L. Murphey, "Intelligent Vehicle Power Management - An Overview", *Computational Intelligence in Automotive Applications*, SCI 132, Springer Verlag, pp.223-251, 2007

[13] M. Mohebbi and M. Farrokhi, "Adaptive neuro control of parallel hybrid electric vehicles", *Int. J. Electric and Hybrid Vehicles*, Vol. 1, No. 1, pp. 3-19, 2007

[14] S. Samuel, L. Austin, and D. Morrey,"Automotive test drive cycles for emissions measurement and real-world emission levels – a review", *Proceedings of the Institution of Mechanical Engineers*, 216, pp.555–564, 2002

[15] H.G.Rosenkranz, H.C. Watson, W. Bryce, and A. Lewis, "Driveability, fuel consumption and emissions of a 1.3 l turbocharged spark ignition engine developed as replacement for a 2.0 l naturally aspirated engine". *Proceedings of the Institution of Mechanical Engineers*, 118, pp.139–150, 1986

[16] NREL, National Renewable Energy Laboratory (2005), *ADVISOR documentation*, available online, http://www.ctts.nrel.gov/analysis/

[17] C. Manzie, H.Watson, H. and S. Halgamuge, "Fuel economy improvements for urban driving: Hybrid vs. intelligent vehicles", *Transportation Research Part C: Emerging Technologies,*Volume 15, Issue 1, pp. 1-16, 2007

Reliability Prediction under Uncertainties using Fuzzy/Possibility Approach

Otilia Elena DRAGOMIR, *Member, IEEE*, Florin DRAGOMIR, Rafael GOURIVEAU and Eugenia MINCA, *Member, IEEE*

***Abstract*—In maintenance field, traditional concepts like preventive and corrective strategies are progressively completed by new ones like predictive and proactive maintenance. For that purpose, a fundamental task is the estimation of the provisional reliability of equipment as well as its remaining useful life. However, traditional approach of reliability based on statistical analysis can be not suitable as very few knowledge can be available. Within this frame, the general purpose of the work is to explore the way of developing a fuzzy approach of reliability modeling and estimation in order to take into account the uncertainty as well as possible. From that, different ways of considering uncertainty in reliability modeling are discussed (probabilistic, fuzzy/possibility approaches), and the inherent limits of all methods are pointed out.**

I. Introduction

The area of the maintenance has increased in the last years due to the triple performance concept, the actual engine of researchers and industrials. The necessity of keeping and improving system availability and safety, as well as product quality, undergoes several major developments. The strategies oriented as well as over social, economical and environmental aspects have been improved with new approaches like proactive ones. The necessity of understanding and evaluation of the future states of the analysed system is the premise of the present paper.

In many situations, especially when a fault or a failure has catastrophic consequences (e.g. nuclear power plant), it is desirable to predict the chance that a machine operates without failure up to some future time (for example, the next inspection), given the machine's current condition and its past operational profile.

In order to solve these problems, the article proposes an evaluation of reliability with a modeling method based on the prognosis under uncertainty. The prognosis makes a better connection between proactive context and perception/ evaluation of the future.

Classical approach of the estimation of the provisional reliability of equipment as well as its remaining useful life using statistical analysis can be not suitable as very few knowledge can be available.

Within this frame, the general purpose of the work is to explore the way of developing a fuzzy/ possibility approach of reliability modeling and estimation in order to take into account the uncertainty as well as possible. The paper is organized as follows: the first part is dedicated to the problem's statement. Fundaments of reliability (concepts, measures and approaches) are given and the principles of degradation modeling are also presented. As a global point of view, the provisional reliability prediction is compared to the prognostic process: two global tasks must be ensured, a first one to predict the evolution of a situation (degradation of the equipment), and a second one to "assess" this predicted situation with regards to an evaluation referential. According to these aspects, potentials informational frameworks in reliability modeling are studied in the second and third parts. The second part is based on the probabilistic formalization of the failure mechanisms for reliability prediction that can be critiqued since the available information may be insufficient. Thus, in the third part the uncertainties in reliability modeling are discussed (probabilistic, fuzzy/possibility approaches), and the inherent limits of all methods are pointed out.

II. RELIABILITY FUNDAMENTS

A. *Reliability: concept and measures*

The reliability (R) performs appropriate analyses and tasks to ensure the system will meet its requirements. The International Committees for Standardization defines it as "the ability of an item to perform required function under given conditions for a given time interval" [1]. A key aspect of reliability is to define "failure". Because reliability is a probability, even highly reliable systems have some chance of failure. Various definitions of failure can be used. The most commonly failure definition, used also in this paper is: "the determination of the ability of an item to perform a required function" [2].

The reliability can be measured in many ways depending

Otilia Elena Dragomir is with Valahia University of Targoviste, Electrical Engineering Faculty, Automation, Computer Science and Electrical Engineering Department, 18 Unirii Av., Targoviste, Romania (e-mail: drg_otilia@yahoo.com).

Florin Dragomir is with Valahia University of Targoviste, Electrical Engineering Faculty, Automation, Computer Science and Electrical Engineering Department, 18 Unirii Av., Targoviste, Romania (e-mail: drg_florin@yahoo.com).

Rafael Gouriveau is with Institut FEMTO-ST, CNRS - UFC / ENSMM / UTBM, Département AS2M,24 rue Alain Savary, 25000 Besançon, France (e-mail: gouriveau@ens2m.fr)

Eugenia Mincă is with Valahia University of Targoviste, Electrical Engineering Faculty, Automation, Computer Science and Electrical Engineering Department, 18 Unirii Av., Targoviste, Romania (e-mail: minca@valahia.ro).

on the particular situation. For systems with a clearly defined failure time, the empirical distribution function of these failure times can be determined. This implies that the system's has an empirical failure distribution (parameterised with a Weibull or a log-normal model) and a constant failure rate. In this case R is defined as follows:

$$R(t) = 1 - F_{\vartheta}(t) = 1 - \int_0^t f_{\vartheta}(u).du \quad (1)$$

Were $R(t)$ is the probability that a failure does not occur before time t, ϑ is time to failure, $f_{\vartheta}(t) = Pr(\vartheta = t)$ is the probability distribution function (known as the failure density function)

Otherwise, the reliability measurement needs using aggregated measures like: the mean time between failures ($MTBF$) and the mean time to failure ($MTTF$).

$$MTTF = \int_0^{\infty} R(t).dt \quad (2)$$

B. Reliability approaches

Usually the performance of a system can be evaluated by one or several critical variables/measures. The performance measures are variables that are highly correlated with the performance of the system during the operation life. Thus, considering all possible system failure modes, each failure mode may be correlated to one or more physical performance measures.

In the case of a *single failure mode* and a single physical performance measure/ variable, failure can be defined when the performance measure exceeds a critical value (threshold) [3].

For a failure mode that is related to *multiple performance variables* (y_1, y_2, …,y_p), the failure definition can be represented by a surface modelled by a function of g((y_1, y_2, …,y_p)= 0. When multiple performance measures lie inside (or outside) the surface/volume, failure is assumed to occur. Assume that are m failure modes considered in a system, the critical surface is given by the function $s_i(y_1, y_2, \ldots, y_p) = 0$ for failure mode i (i = 1,2, ...,m). It should be noted that for each function s_i, it is not necessary to include all y_i (i = 1,2, ...,m) in the function. In other words, each failure mode may be affected by some of the performance measures [3].

The concept of system *performance reliability prediction with multiple performance measures and multiple failure modes* can be briefly explained as follows. Suppose $f_t(y_1, y_2, \ldots, y_p)$ represents a joint probability density function of performance variables at time t. Then the probability that the system will failure in mode i (i= 1,2, ...,m) up to time t is given by (for a smaller-is-better case):

$$F_i(t) = \int \ldots \int_{\Omega_i} f_t(y_1, y_2, \ldots y_p) dy_1 dy_2 \ldots dy_p \quad (3)$$

Where, Ω_i is the space determinate by $s_i(y_1, y_2, \ldots, y_p) \rangle 0$

The overall system reliability considering all m failure modes can be evaluated as

$$F_i(t) = \int \ldots \int_{\Omega} f_t(y_1, y_2, \ldots y_p) dy_1 dy_2 \ldots dy_p \quad (4)$$

Where $\Omega : \Omega_1 \cup \Omega_2 \cup \ldots \cup \Omega_m$.

The integration over area Ω_1 represents the failure probability regarding failure mode 1. The integration over area Ω_2 represents the failure probability regarding failure mode 2. The integration over the union of Ω_1, Ω_2… Ω_m represents the failure probability with respect to both failure modes — overall system failure probability.(fig. 1)

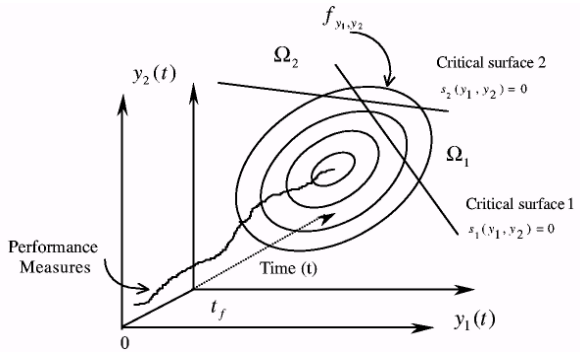


Fig.1. The concept of multivariate performance reliability assessment inputs and multiple outputs [3]

The above discussion reveals that the key issue in system reliability assessment with multiple failure modes and multivariate performance measures is to find *the joint density distribution of multiple performance measures* [4]. If a joint multivariate normal distribution is assumed, then the joint probability density function of the distribution is represented in terms of a mean vector and a covariance matrix of performance measures. The mean vector and covariance matrix are obtained from forecasting results of performance measures. The forecasted mean vector and covariance matrix determine the distribution of performance measures at a future time. The distribution and failure definitions determine the conditional performance reliability.

Eqs. (3)-(4) provide formulas for calculating system performance reliability, considering each failure mode and overall performance reliability with regard to all failure modes. From these equations, it is clear that performance reliability is obtained by implementing a multidimensional integration over a predefined area. The integral function is the joint probability density function, and the integration limits/ ranges are determined by predefined critical surfaces in terms of performance measures that are determined by failure definitions. In the case of a system exhibiting multiple failure modes, each failure mode has its own defined integration area [5].

III. RELIABILITY PREDICTION BASED ON DEGRADATION MODELING

A. *Reliability prediction literature*

Reliability prediction based on degradation modeling can be an efficient method to estimate reliability for some highly reliable parts or systems when observations of failures are rare. Although the concept of reliability prediction based on degradation modeling is relatively recent, there have been several important and successful applications.

Lu [6] presented one of the first successful applications of degradation modeling to predict reliability. They have used a regression model for the analysis of degradation data at a fixed level of stress (i.e. no acceleration) to estimate a time-to-failure distribution. A 'two-stage' method was used to estimate the mixed-effect path model parameters. Monte Carlo simulation was then used to estimate the time-to-failure distribution function and they suggested bootstrap methods for estimating confidence intervals.

Other researchers have developed different perspectives and modeling approaches to this reliability prediction paradigm. Sethuraman [7] developed a cumulative damage threshold crossing model. Under this model, an item consists of a large number of components that suffer damage at regular moments of time. Failure occurs as soon as the maximum cumulative damage to some component crosses a certain threshold. Time-to-failure data is used to estimate the model parameters.

Zuo [8] introduced three approaches for reliability modeling of continuous state devices. The three approaches are based on a random process model, the general path model and the multiple linear regression models, respectively. They also proposed a mixture model that can be used to model both catastrophic failures and degradation failures.

Luo [5] developed a technique for predicting system performance reliability in real-time considering multiple failure modes. The technique includes on-line multivariate monitoring and forecasting of performance measures and conditional performance reliability estimates. The performance measures are treated as a multivariate time series and a state-space approach is used to model the multivariate time series. The predicted mean vectors and covariance matrix of performance measures are used for the assessment of system reliability. The technique provides a means to forecast and evaluate the performance degradation of an individual system in a dynamic environment in real-time.

There have also been other successful examples of degradation modeling.

Abdel [9] introduced failure models of devices subject to deterioration (degradation). He discussed the properties of different classes of life distributions (such as increasing failure rate, increasing failure rate average, etc.) under various threshold distributions.

Pascual [10] used a random fatigue-limit model to describe the variation in fatigue life and the unit-to-unit variation in the fatigue limit.

B. *Degradation modeling for reliability prediction*

In order to predict the evolution of a system or to understand the reliability of a component, two separate fields of investigation are common. The physics of failure approach used to understand the failure mechanisms involved, such as: crack propagation, chemical corrosion or the parts stress. The second one is the empirical method used for reliability prediction based on counting the number and type of components of the system, and the stress they undergo during operation.

The operational context that leads our researches suits well with the part stress modeling. Two objectives have to distinguish: the prediction and evaluation of future state of the system and the knowledge of failure referential.

The mathematic of reliability proposed in this context supposes that failure can be characterized by a random variable. This can be difficult to obtain and another way of formalizing the reliability is that of degradation modeling.

Let assume now that the failure is characterized by the fact that the degradation of the equipment (y) overpass a degradation limit (y_{lim}). At any time t, the failure probability can thereby be defined as follows:

$$F_{fail}(t) = Pr\left[y(t) \geq y_{lim}\right] \tag{5}$$

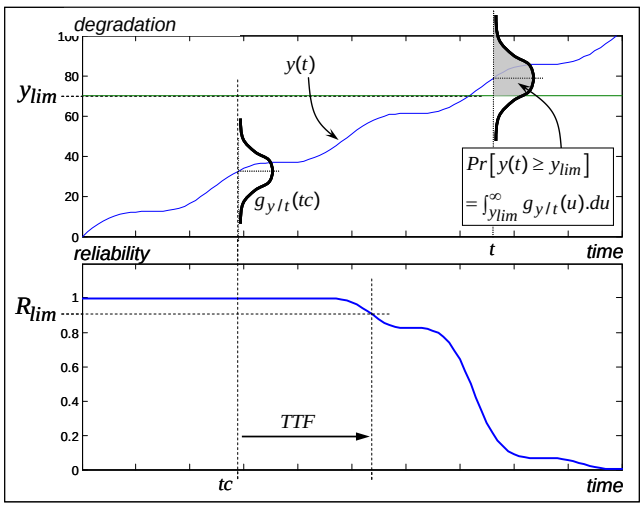


Fig. 2. Reliability and degradation modeling.

Assuming that the degradation can be probabilistically modeled and $g_{y/t}(t)$ it's the probability distribution function at time t, by analogy with reliability theory, the reliability modeling is, at time t:

$$R(t) = 1 - F_{fail}(t)$$
$$R(t) = 1 - Pr\left[y(t) \geq y_{lim}\right] \quad (6)$$
$$R(t) = 1 - \int_{y_{lim}}^{\infty} g_{y/t}(u).du$$

The remaining time to failure (TTF) of the system can finally be expressed as the remaining time between current time (tc) and the time to underpass a reliability limit (R_{lim}) fixed by the practitioner (Fig. 2). This can be generalized with a multi-dimensional degradation signal. The above statements reveal that a key issue in reliability prediction is the apprehension of the failure mechanism.

C. *Probabilistic approach: illustration and discussion*

An illustration of the probabilistic approach is proposed in fig. 3.in which the 2D-graph is that of the resulting failure and reliability modeling.

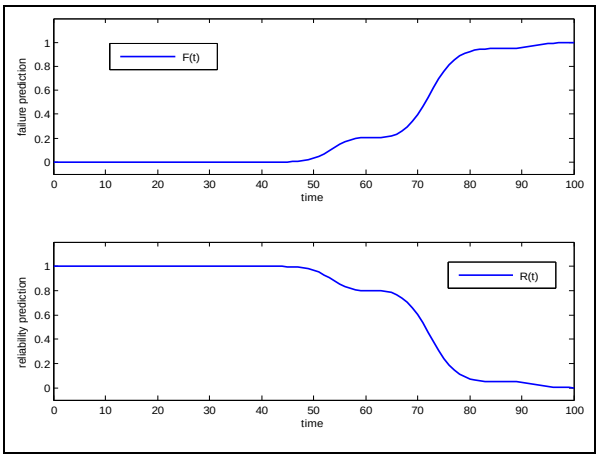


Fig. 3. Illustration of the probabilistic approach of reliability modeling

In traditional approach, the failure distribution is either obtained empirically by evaluating the ratio of items that do not perform their function in a stated period to the total number in the sample, or expressed by an expert. In degradation modeling approach, the same method can be conducted to construct a degradation model $y(t)$. However, if failures observations are rare, these approaches can be difficult to achieve, even impossible. Indeed, industrial not always can test several systems since it can be too expensive. In addition, results depend on the operational conditions and the extrapolation to a specific case can require heavy mathematical treatments. All this is all the more critic as the system evolves and the reliability estimation and prediction are thereby difficult to perform.

IV. RELIABILITY PREDICTION USING FUZZY/ POSSIBILITY THEORIES

A. *From prognostic to reliability*

Reliability modeling can however be assimilated to the prognostic process. According to Dragomir [11] acceptation of prognostic, prognostic and reliability modeling by using degradation modeling are very similar. Thus, prognostic could be split into 2 sub-activities: a first one to predict the evolution of a situation at a given time (forecasting process), and a second one to assess this predicted situation with regards to a referential. For clarity of presentation, both levels of prognostic are here discussed sequentially. They are however linked together in reality.

At the prediction level, a prognostic system should be able to determine the future state of equipment as closely as possible to the future real state. Also, the control of the performance of prediction is the premise of a good global prognostic system. At the evaluation level, the predicted situation should be evaluated quantitatively and qualitatively regarding the reference levels, RUL, confidence, accuracy, etc., which implies the definition of prognostic measures.

Following that, two type of uncertainty must be taken into account: this one inherent to the degradation prediction. As there can be very few information about the phenomena under study (and its evolution), probability should be used carefully and another one, inherent to referential limits. The acceptability of the degradation limits of equipment can be unclear and difficult to formalize.

Finally, the reliability modeling is relevant if both types of uncertainties are well bounded and treated in prognostic systems.The purpose of next part is to estimate the reliability with prognostic tools in an operational context under uncertainty in a particular case: the referential used in performance prognosis evaluation is the same with the failure level.

B. *Towards fuzzy/possibilistic prediction of reliability*

In practice, the reliability prediction by using the probabilistic approach of degradation can be used if both the degradation signal and the degradation limit are expressed as probability functions. Thus, when performing the prediction step of prognostic, this implies applying statistical techniques in order to take into account the uncertainty of the degradation signal estimation. However, these techniques can be difficult and computationally time expensive to deploy if real systems are complex and non-linear.

With regard to the degradation limit, many experiments on failure occurrence should be used in order to assign the confidence (pdf) on performance threshold. However, if few knowledge of the failure mechanism is available, it can be difficult to formalize it in probabilistic terms. Moreover, translating an expert's knowledge in probabilistic terms injects unjustified information in formalized data.

In order to undergo some limits of probability theory a fuzzy/possibility approach modeling is proposed in next section

C. Principle of the fuzzy / possibilistic reliability modeling approach

In a few words, fuzzy logic and possibility theory aim at reasoning with imprecise or vagueness knowledge, by introducing a novel way of taking into account uncertainty. Globally, possibility theory enables judging from the veracity of a proposition by the use of two indicators (whereas probability theory that is found on a single measure): the possibility measure (labeled Π) and the necessity measure (labeled N). In order to introduce these measures, let have a look to the possibility distribution concept.

A possibility distribution, labeled π, is an application from the universe of the discourse Ω to the interval [0 , 1]. It characterizes a fact defined on Ω and designates an appreciation on the belonging of all value of Ω to the fact represented. Note that a possibility distribution function is normalized: $\sup[\pi(w)] = 1, w \in \Omega$.

Possibility theory introduces confidence measures that allow evaluating the degree with which a fact is in accordance with a reference set and to evaluate the degree of certainty of this assertion.

Let note π_F the possibility distribution membership of a fact F, and μ_{Ref} the possibility distribution membership of a reference situation Ref, then, possibility and necessity indicators are:

$$\begin{aligned} \Pi_{F \in Ref} &= \sup_{w \in \Omega} \left\{ min\left[\mu_{Ref}(w), \pi_F(w) \right] \right\} \\ \mathrm{N}_{F \in Ref} &= \inf_{w \in \Omega} \left\{ max\left[\mu_{Ref}(w), 1 - \pi_F(w) \right] \right\} \end{aligned} \quad (7)$$

The possibility measure can be interpreted as the degree of intersection between the values compatibles with Ref and the set of possible values for F and designates thereby the possibility that F corresponds to Ref (scale between 0 and 1). Note, that this measure does not exclude the possibility of the contrary: at least one of both propositions is completely possible.

The necessity measure traduces the inclusion degree between the set of possible values of F with the compatibles values of Ref. This indicator completes the possibility measure by indicating the degree with which the information is certain.

Let finally introduce an interesting characteristics of possibility and necessity measures with regard to probability. It can be shown that an equivalence class $\mathcal{P}$ of probabilities measures Pr for an event A can be defined as:

$$\mathcal{P} = \left\{ Pr / \forall A, \mathrm{N}(A) \le Pr(A) \le \Pi(A) \right\} \quad (8)$$

According to eq. (7), possibility and necessity indicators enable evaluating the degree with which the assertion "the degradation signal is within the failure member set" and the degree of certainty of this assertion as follows. At time t,

$$\begin{aligned} \Pi_{degr \in fail}(t) &= \sup_{u \in Y} \left\{ min\left[\mu_{y lim / t}(u), \pi_{y/t}(u) \right] \right\} \\ \mathrm{N}_{degr \in fail}(t) &= \inf_{u \in Y} \left\{ max\left[\mu_{y lim / t}(u), 1 - \pi_{y/t}(u) \right] \right\} \end{aligned} \quad (9)$$

According to eq. (8), possibility and necessity indicators enable bounding the probability of occurrence of an event. Thereby

$$\mathrm{N}_{degr \in fail}(t) \le Pr\left[failure \right](t) \le \Pi_{degr \in fail}(t) \quad (10)$$

Following that, the reliability modeling expression is at time t:

$$\begin{aligned} 1 - \Pi_{degr \in fail}(t) \le 1 - Pr\left[failure \right](t) \le 1 - \mathrm{N}_{degr \in fail}(t) \\ 1 - \Pi_{degr \in fail}(t) \le R(t) \le 1 - \mathrm{N}_{degr \in fail}(t) \end{aligned} \quad (11)$$

D. Fuzzy/possibility approach: illustration and discussion

An illustration of the fuzzy / possibilistic approach is proposed in fig. 4.which the 3D-graph is that of the possibilistic modeling of the degradation signal an limit, and the 2D-graph, that of the resulting failure and reliability modeling.

In practice, this approach can be used if both the degradation signal and the degradation limit are expressed as possibility distributions functions. Thus, when performing the prediction step of prognostic, this implies that one must be able to fuzzyfy the degradation signal estimation. In regard to the degradation limit, fuzzy modeling is particularly adapted to the formalization of expert knowledge. At this stage, note that one can directly construct the set of non acceptable degradation value without looking for a defined threshold: indeed, in Figure 8, $\mu_{y lim}(t, y)$ can be seen as a "cumulative distribution function".

Moreover, if sufficient experiences are available, the degradation limit membership functions can be obtained (like with statistical approach) by using neuro-fuzzy system as proposed by [12].

According to eq. (11), possibility theory enables giving boundaries to the reliability modeling and, thereby, allows estimating more confidently the remaining useful life or time to failure of a system (as proposed in Figure 1). In opposite to it, probability theory does not "conserve" the uncertainty of knowledge since the process of reliability modeling results in a single aggregated indicator: the same confidence will be accorded to two situations for which the formalized knowledge can be very different (spreading of the degradation signal and limit).

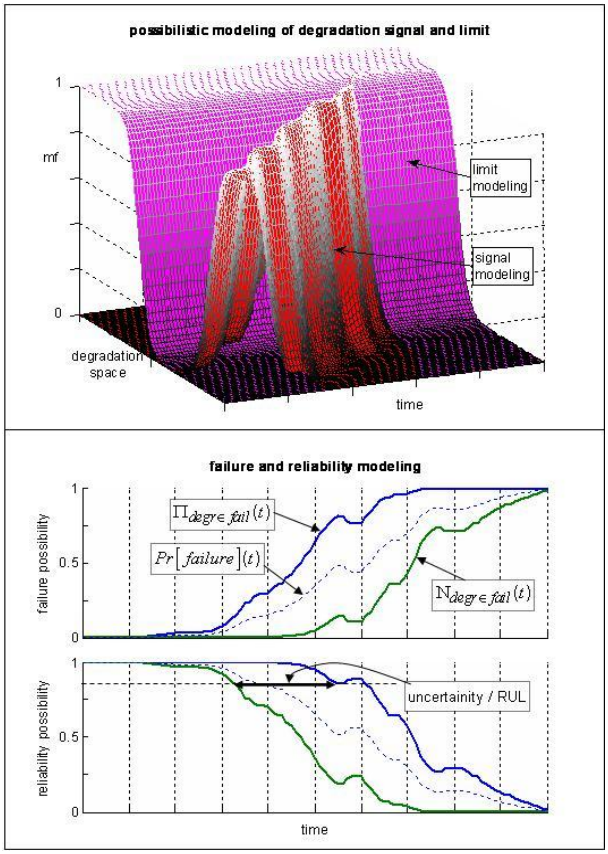


Fig. 4. Illustration of the probabilistic approach of reliability modeling.

Following that, fuzzy / possibilistic approach of reliability modeling enables practitioners to be more critic on the risk incurred by the system and therefore, to build adequate maintenance policies. Let also note that the approach is much more computationally effective than that of probability theory since it is based on "min, max, sup, inf" operators

V. Conclusion

In maintenance field, traditional concepts like preventive and corrective strategies are progressively completed by new ones like predictive and proactive maintenance. For that purpose, a fundamental task is the estimation of the provisional reliability of equipment as well as its remaining useful life. In the paper, reliability modeling is assimilated to the prognostic process since two mains tasks must be ensured: a first one to predict the evolution of the degradation of the system, and a second one to asses this predicted situation with regard to a degradation limit referential. Following that, two ways of taking into account the uncertainty are discussed.

The traditional approach of reliability based on statistical analysis can be not suitable as very few knowledge can be available: real systems are complex; there are many uncertainties upon their behaviors. Moreover, it can be difficult and computationally time expensive to deploy if real systems are complex and non-linear.

The fuzzy / possibilistic approach of on-line reliability modeling and estimation is well adapted to the integration of expertise. Moreover, such an approach aims at considering the available knowledge as it is: uncertain and imprecise if necessary. As a consequence, the processed reliability indicators can be considered with more confidence. In addition, this approach is not so time expensive. However, it requires the fuzzyfication of the degradation signal prediction.

The work reported here is still a prospective one and it is obviously extended. The main developments that are at present led deal with the reliability modeling of components with various failure modes and to the extension of the approach to a global system (composed of different components whose reliability characteristics can be fuzzily estimated). In such cases, although there is a great interest in considering knowledge as something imprecise and uncertain, global treatments can led to situation in which there is no way to conclude upon adequate maintenance strategies since individual reliabilities are expressed as intervals…

References

[1] CEN/TC 319 (2001). Maintenance terminology. European Standard, *European Committee for Standardization*

[2] ISO 13381-1 (2004). Condition monitoring and diagnostics of machines - prognostics - Part1: General guidelines. *International Standard, ISO*.

[3] Lu, S., H. Lu and W.J. Kolarik (2001). Multivariate performance reliability prediction in real-time, *Reliability Eng. and System Safety*, vol. 72, pp. 39-45

[4] Wang, P. and D. Coit (2004). Reliability prediction based on degradation modeling for systems with multiple degradation measures. *In: Proc. of Reliab. and Maintain. Ann. Symp. - RAMS*, pp. 302–307

[5] Luo J, Bixby A, Pattipati K, Qiao L, Kawamoto M, Chigusa S (2003). An interacting multiple model approach to model-based prognostics. *Syst Secur and Assurance*. 1: 189-194

[6] Lu, C. J. and Meeker, W.Q. (1993), Using Degradation Measures to Estimate a Time-to-Failure Distribution, *Technometrics*, Vol. 35, No.2, p161-173.

[7] Sethuraman, J. and Young, Thomas R. (1986), Cumulative damage threshold crossing models, in Reliability and Quality Control, A. P. Basu (Editor*), Elsevier Science Publishers* B. V. (North-Holland), pp309-319.

[8] Zuo, Ming J., Jiang , Renyan, and Yam Richard C.M. (1999), Approaches for reliability modeling of continuous-state devices, *IEEE Transactions on Reliability*, Vol. 48, No. 1,pp9-18.

[9] Abdel-Hameed, M. (1992), Failure models of devices subject to deterioration, Order Statistics and Nonparametrics: *Theory and Applications*, Elsevier Science Publishers B. V.,pp307-313.

[10] Pascual, Francis G., Meeker, William Q. (1999), Estimating fatigue curves with the random fatigue-limit model (discussion and reply). *Technometrics*, Vol. 41 no4, Nov. 1999, p. 277-302.

[11] Dragomir, O., R. Gouriveau and N. Zerhouni (2007). Framework for a distributed and hybrid prognostic system. *In: 4th IFAC Conf. on Management and Control of Production and Logistics*, MCPL 2007, Sibiu, Romania

[12] Chinnam, R. and B. Pundarikaksha (2004). A neurofuzzy approach for estimating mean residual life in condtition-based maintenace systems. *Int. J. materials and Product Technology*, vol. 20:1-3, pp. 166–179.

One- and Two-Degree-of-Freedom Fuzzy Control of an Electromagnetic Actuated Clutch

Claudia-Adina Dragoş, Stefan Preitl, *Senior Member, IEEE*, Radu-Emil Precup, *Senior Member, IEEE*, Cristian-Sorin Neş, Emil M. Petriu, *Fellow, IEEE*, and Gabriela Tîrtea

***Abstract*—This paper presents two new fuzzy control solutions dedicated to the position control of an electromagnetic actuated clutch. The two control solutions employ a Takagi-Sugeno one-degree-of-freedom (1 DOF) fuzzy controller and a Takagi-Sugeno two-degree-of-freedom (2 DOF) fuzzy controller. The digital simulations results corresponding prove that both fuzzy control solutions ensure good control system performance with respect to the modifications of the reference input. Comparisons between the two fuzzy control solutions and a classical control solution are highlighted.**

I. Introduction

THE research concerning the design of control systems for automotive applications is challenging due to the very good control system performance they require to be obtained at low cost in terms of ensuring both fuel economy and safety [1], [2]. With this regard the design of control systems for electromagnetic actuated clutches are important as actuators in the framework of automotive control systems, and several control solutions for electromagnetic actuated clutches are reported in the literature.

An internal model control structure with two-degree-of-freedom (2 DOF) PID controller is suggested in [3]. It achieves good control system performance in terms of reference input tracking and disturbance rejection as well when controlling the typical integrator and dead-time plant.

Model predictive control approaches to control the magnetically actuated mass-spring-damper system are offered in [4]. The model predictive control architecture is implemented to optimize only the nonlinear dynamics of the mechanical subsystem because the electromagnetic subsystem controlled by an inner-loop controller exhibits much faster dynamics.

The 2 DOF controllers are designed in [5] to solve some control problems specific to flight motion simulators. They satisfy robust stability and robust performance as well.

The one-degree-of-freedom (1 DOF) and 2 DOF controllers prove to be successful in many other applications [6]–[19]. The controlled plants in such applications are very close to those used in mechatronic control systems.

This paper aims the design of new Takagi-Sugeno 1 DOF and 2 DOF fuzzy control solutions. They are advantageous because of the very good control system performance they offer accounting for low cost design and implementations.

The paper is organized as follows: based on the mathematical model of the controlled plant presented in Section II, Section III is focused on the design of the two original fuzzy control solutions. Section IV is dedicated to the digital simulation results concerning the position control of an electromagnetic actuated clutch. Section V highlights the conclusions.

II. Modeling of Electromagnetic Actuated Clutch

The state-space mathematical model (MM) of the controlled plant in the electromagnetic actuated clutch is derived on the basis of the first principle MM of a magnetically mass-spring-damper system which consists of an electrical and a mechanical subsystem [20]–[24]. The nonlinear state-space MM of the controlled plant is

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= -\frac{k}{m}x_1 - \frac{c}{m}x_2 + \frac{k_a}{m\,k_b^2}x_3^2, \\ \dot{x}_3 &= -\frac{R\,k_b}{2k_a}x_3 + \frac{1}{k_b}x_2x_3 + \frac{k_b}{2k_a}V, \\ y &= x_1, \end{aligned} \tag{1}$$

where: $x_1 = x$ – the mass position, $x_2 = \dot{x}$ – the mass speed, $x_3 = i$ – the current, V – the control signal, y – the controlled output, k – the stiffness of the spring, c – the coefficient of the damper, R – the resistance of the resistive circuit subjected to magnetic flux variations according to Faraday's law, and k_a, k_b – the constants in the relation between the magnetic flux and the current. The parameters afferent to (1) are given in [22]–[24].

It is convenient to linearize the MM of the controlled plant around ten operating points

Manuscript received April 10, 2010. This work was supported in part by the CNMP and CNCSIS of Romania.

C.-A. Dragoş, S. Preitl, R.-E. Precup and G. Tirtea are with the "Politehnica" University of Timisoara, Department of Automation and Applied Informatics, Bd. V. Parvan 2, RO-300223 Timisoara, Romania (phone: +40 2564032 -29, -30, -24; fax: +40 2564032-14; e-mail: claudia.dragos@aut.upt.ro, spreitl@aut.upt.ro, radu.precup@aut.upt.ro, gabriellee3203@yahoo.com).

C.-S. Nes is with the "Politehnica" University of Timisoara, Department of Strength of Materials, Bd. M. Viteazu 1, RO-300222 Timisoara, Romania (e-mail: cristianedonis@yahoo.com).

E. M. Petriu is with the University of Ottawa, School of Information Technology and Engineering, 800 King Edward, Ottawa, ON, K1N 6N5 Canada (e-mail: petriu@site.uottawa.ca).

$$
\begin{aligned}
&1:\{x_{10}=0.0033,\quad x_{30}=1,\quad V_0=1.2\},\\
&2:\{x_{10}=0.0027,\quad x_{30}=2,\quad V_0=2.4\},\\
&3:\{x_{10}=0.0023,\quad x_{30}=3,\quad V_0=3.6\},\\
&4:\{x_{10}=0.0021,\quad x_{30}=4,\quad V_0=4.8\},\\
&5:\{x_{10}=0.002,\quad x_{30}=5,\quad V_0=6\},\\
&6:\{x_{10}=0.0021,\quad x_{30}=6,\quad V_0=7.2\},\\
&7:\{x_{10}=0.0023,\quad x_{30}=7,\quad V_0=8.4\},\\
&8:\{x_{10}=0.0027,\quad x_{30}=8,\quad V_0=9.6\},\\
&9:\{x_{10}=0.0033,\quad x_{30}=9,\quad V_0=10.8\},\\
&10:\{x_{10}=0.0038,\quad x_{30}=9.8,\quad V_0=11.76\}.
\end{aligned}
\tag{2}
$$

in order to make use of the linearized MMs for the controller designs. The linearized state-space MMs obtain the unified expression

$$
\begin{aligned}
&\dot{\mathbf{x}}=\mathbf{A}\,\mathbf{x}+\mathbf{b}\,\Delta V,\\
&\Delta y=\mathbf{c}^T\mathbf{x},\\
&\mathbf{x}=[x_1=x\quad x_2=\dot{x}\quad x_3=\lambda]^T,\\
&\mathbf{A}=\begin{bmatrix}0 & 1 & 0\\ -k/m & -c/m & 2k_a x_{30}/(mk_b^2)\\ 0 & x_{30}/k_b & -x_{20}/k_b-Rk_a/(2k_b)\end{bmatrix},\\
&\mathbf{b}=\begin{bmatrix}0\\0\\k_b/(2k_a)\end{bmatrix},\ \mathbf{c}^T=[1\quad 0\quad 0].
\end{aligned}
\tag{3}
$$

The transfer function (t.f.) of the linearized MM defined in (3) for the variations of all variables with respect to (2) is

$$
H_P(s)=\frac{\Delta y(s)}{\Delta V(s)}=\frac{k_P}{(1+sT_1)(1+sT_2)(1+sT_3)},
\tag{4}
$$

where $T_1>T_2>T_3$.

III. Design of Fuzzy Control Solutions

A. Design of Takagi-Sugeno 1 DOF Fuzzy Control Solution

The control structure with Takagi-Sugeno 1 DOF fuzzy controller is presented in Fig. 1, where TS-FC is a Takagi-Sugeno fuzzy controller with the integration of the output variable. The design of the block TS-FC starts with the design of the linear continuous controllers with the t.f.

$$
C(s)=\frac{k_c}{s}(1+T_c s)=\frac{k_C}{sT_i}(1+T_i s),
\tag{5}
$$

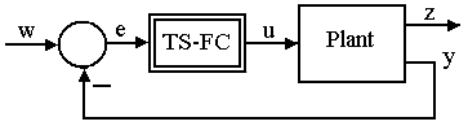


Fig. 1. Structure of control system with Takagi-Sugeno 1 DOF fuzzy controller.

and the Modulus Optimum method leads to

$$
T_c=T_i=T_1,\ k_c=\frac{k_C}{T_i}=\frac{1}{2k_P(T_2+T_3)}.
\tag{6}
$$

The continuous PI controller with the t.f. (5) is discretized using Tustin's method with the sampling period T_s. Five quasi-continuous digital PI controllers with the following transfer functions are obtained:

$$
\Delta u_k^i=\gamma(k_I^i\Delta e_k+\alpha k_P^i e_k),
\tag{7}
$$

where $i\in\{3,5,6,7,8\}$ is the rule index which corresponds also to the index of the operating points defined in (2), e_k is the control error, Δe_k is the increment of control error, and Δu_k^i is the increment of control signal. The expressions of the parameters presented in (7) are

$$
k_P^i=k_C^i(1-\frac{T_s}{2T_i^i}),\ k_I^i=\frac{k_C^i T_s}{T_i^i},\ \alpha=\frac{k_I^i}{k_P^i}.
\tag{8}
$$

The fuzzification of the two input variables of TS-FC, e_k and Δe_k, is solved by three linguistic terms with the triangular membership functions presented in Fig. 2.

The parameters of the 1 DOF fuzzy controller with output integration, B_e and $B_{\Delta e}$, are tuned as follows according to the modal equivalence principle [25]:

$$
B_e=0.01,\ B_{\Delta e}=(k_P^i/k_I^i)B_e=\alpha B_e=0.0002.
\tag{9}
$$

The block TS-FC makes use of the weighted average method for defuzzification. The inference engine uses the MAX and MIN operators, and the rule base is

$$
\begin{aligned}
&\text{IF}\ (e_k\ \text{IS LTE}^i\ \text{AND}\ \Delta e_k\ \text{IS LTDE}^i)\\
&\text{THEN}\ \Delta u_k=\Delta u_k^i,
\end{aligned}
\tag{10}
$$

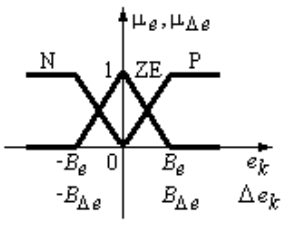


Fig. 2. Input membership functions of TS-FC.

where $\mathrm{LTE}^i, \mathrm{LTDE}^i \in \{\mathrm{N}, \mathrm{ZE}, \mathrm{P}\}$ are the linguistic terms of the two input linguistic variables, and the consequents of the five rules are expressed in (7).

B. Design of Takagi-Sugeno 2 DOF Fuzzy Control Solution

The control structure with Takagi-Sugeno 2 DOF fuzzy controller is presented in Fig. 3, where three components are highlighted: the controller $T(z)$, which ensures the feed-forward control of the system (w – the reference input referred to also as the setpoint for constant values of w), the controller $R(z)$ which deals with the control loop, and TS-FC – the fuzzy block which fuzzifies the linear integral component $S(z)$ for the sake of performance enhancement.

The design of the 2 DOF fuzzy control solution involves two phases. First the linear 2 DOF controller is designed and second the block TS-FC is designed to ensure the integral component of the 2 DOF fuzzy controller.

The design of the linear 2 DOF controller aims the calculation of the polynomials $T(z)$, $R(z)$ and $S(z)$. With this regard the controlled plant is characterized by the pulse t.f.

$$H_P(q^{-1}) = \frac{q^{-1}B(q^{-1})}{A(q^{-1})},\ n_A = \deg A(q^{-1}),\quad n_B = \deg B(q^{-1}), \tag{11}$$

where the polynomials $A(q^{-1})$ and $B(q^{-1})$ are expressed as follows for the linearized MMs corresponding to five operating points extracted out of the points (2):

$$\begin{aligned}
&\text{For operating point 3:}\\
&A(q^{-1}) = 1 - 1.32q^{-1} + 0.885q^{-2} - 0.46q^{-3}\\
&B(q^{-1}) = 0.04\cdot 10^{-4}\cdot q^{-1} + 0.13\cdot 10^{-4}\cdot q^{-2} - 0.03\cdot 10^{-4}\cdot q^{-3}\\
&\text{For operating point 5:}\\
&A(q^{-1}) = 1 - 0.52q^{-1} + 0.057q^{-2} - 0.45q^{-3}\\
&B(q^{-1}) = 0.06\cdot 10^{-4}\cdot q^{-1} + 0.18\cdot 10^{-4}\cdot q^{-2} - 0.04\cdot 10^{-4}\cdot q^{-3}\\
&\text{For operating point 6:}\\
&A(q^{-1}) = 1 - 0.42q^{-1} + 0.07q^{-2} - 0.008q^{-3}\\
&B(q^{-1}) = 0.17\cdot 10^{-3}\cdot q^{-1} + 0.025\cdot 10^{-3}\cdot q^{-2} - 0.003\cdot 10^{-3}\cdot q^{-3}\\
&\text{For operating point 7:}\\
&A(q^{-1}) = 1 - 0.83q^{-1} + 0.135q^{-2} - 0.09q^{-3}\\
&B(q^{-1}) = 0.8\cdot 10^{-4}\cdot q^{-1} + 0.007\cdot 10^{-4}\cdot q^{-2} - 0.08\cdot 10^{-4}\cdot q^{-3}\\
&\text{For operating point 8:}\\
&A(q^{-1}) = 1 - 0.53q^{-1} - 0.16q^{-2} - 0.09q^{-3}\\
&B(q^{-1}) = 0.83\cdot 10^{-4}\cdot q^{-1} + 0.24\cdot 10^{-4}\cdot q^{-2} - 0.09\cdot 10^{-4}\cdot q^{-3}
\end{aligned} \tag{12}$$

Use is made of the reference model with the t.f.

$$H_{RM}(s) = \frac{\omega_m^2}{s^2 + 2\omega_m\varsigma_m s + \omega_m^2} = \frac{3025}{s^2 + 99s + 3025}, \tag{13}$$

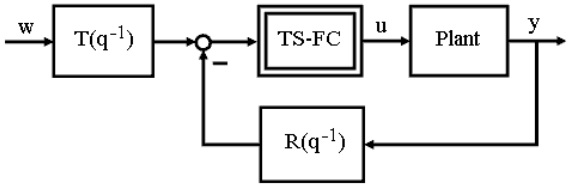


Fig. 3. Structure of control system with Takagi-Sugeno 2 DOF fuzzy controller.

which is discretized in terms of

$$H_{RM}(z) = \frac{B_m(z)}{A_m(z)} = \frac{0.0087z + 0.008}{z^2 - 1.764z + 0.78} \tag{14}$$

to impose the desired closed-loop t.f. of the control system with respect to the reference input.

Next the polynomial $P(q^{-1})$ is the denominator of the output sensitivity function, the following Diophantine equations are solved:

$$\begin{aligned}
&A(q^{-1})S(q^{-1}) + q^{-d}B(q^{-1})R(q^{-1}) = P(q^{-1}),\\
&n_P = \deg P(q^{-1}) \le n_A + n_B + d - 1,\\
&n_S = \deg S(q^{-1}) \le n_B + d - 1,\\
&n_R = \deg R(q^{-1}) \le n_A - 1,
\end{aligned} \tag{15}$$

where

$$P(q^{-1}) = 1 + p_1 q^{-1} + p_2 q^{-2} + \ldots, \tag{16}$$

and the polynomials $S(q^{-1})$ and $R(q^{-1})$ are obtained:

$$\begin{aligned}
&\text{For operating point 3:}\\
&S(q^{-1}) = 1 + 0.6q^{-1} + 0.094q^{-2}\\
&R(q^{-1}) = -1.2\cdot 10^4 + 1.4\cdot 10^4\cdot q^{-1} + 1.3\cdot 10^4\cdot q^{-2} - 1.5\cdot 10^4\cdot q^{-3}\\
&\text{For operating point 5:}\\
&S(q^{-1}) = 1 - 0.16q^{-1} - 0.1079q^{-2}\\
&R(q^{-1}) = -1.2\cdot 10^4 + 4.3\cdot 10^4\cdot q^{-1} - 4.1\cdot 10^4\cdot q^{-2} + 1.1\cdot 10^4\cdot q^{-3}\\
&\text{For operating point 6:}\\
&S(q^{-1}) = 1 + 0.127q^{-1} + 0.0135q^{-2}\\
&R(q^{-1}) = 7.5\cdot 10^3 - 2.2\cdot 10^3\cdot q^{-1} - 0.3\cdot 10^3\cdot q^{-2} - 0.04\cdot 10^3\cdot q^{-3}\\
&\text{For operating point 7:}\\
&S(q^{-1}) = 1 + 0.4q^{-1} + 0.33q^{-2}\\
&R(q^{-1}) = 4\cdot 10^3 - 2.7\cdot 10^3\cdot q^{-1} + 4.7\cdot 10^3\cdot q^{-2} - 3.5\cdot 10^3\cdot q^{-3}\\
&\text{For operating point 8:}\\
&S(q^{-1}) = 1 - 0.2q^{-1} + 0.03q^{-2}\\
&R(q^{-1}) = 1.6\cdot 10^4 - 1.2\cdot 10^4\cdot q^{-1} + 0.2\cdot 10^4\cdot q^{-2} - 0.03\cdot 10^4\cdot q^{-3}
\end{aligned} \tag{17}$$

The fuzzification of $S(z)$ makes use of the same fuzzification (Fig. 2), inference engine and defuzzification method as for the Takagi-Sugeno 1 DOF fuzzy controller. However the rule base is different to that presented in (10):

$$\begin{aligned} &\text{IF } (e_k \text{ IS LTE}^i \text{ AND } \Delta e_k \text{ IS LTDE}^i) \\ &\text{THEN } u_k^i = k_S^i + \Delta u_k^i = k_S^i + k_P^i e_k + k_I^i \Delta e_k, \end{aligned} \tag{18}$$

where the parameters k_I^i, k_P^i and k_S^i are functions of the parameters of the conventional controllers $S(z)$ of each linear 2 DOF controller (with the index i) designed for the models linearized around five operating points:

$$\begin{aligned} k_I^i &= k_C^i T_s / T_i^i, \\ k_P^i &= k_C^i (1 - T_s /(2T_i^i)), \\ k_S^i &= s_0^i - s_1^i - s_2^i. \end{aligned} \tag{19}$$

The modal equivalence principle leads to the parameters of the fuzzy controller

$$B_e = 20, \; B_{\Delta e} = 16 \,. \tag{20}$$

IV. Digital Simulation Results

The designed control systems are tested with respect to the step and rectangular modifications of the reference input. The behavior of the 1 DOF fuzzy control system is illustrated in Fig. 4.

The 2 DOF fuzzy control system exhibits different behaviors depending on the linear components $T(z)$ and $R(z)$ separately designed around four operating points. The block TS-FC is the same in all four cases but the feed-forward $T(z)$ and feedback $R(z)$ filters depend on the MMs of the controlled plant which are linearized around the four operating points.

The behaviors of the 2 DOF fuzzy control system are presented in Figs. 5.b to 8.b. Some comparisons with a classical controller were done to highlight the good tracking of the fuzzy control structures. The behaviors of the PID control system are illustrated in Figs. 5.a to 8.a.

V. Conclusion

This paper offers two original Takagi-Sugeno fuzzy control solutions dedicated to the position control of an electromagnetic actuated clutch. The low cost design of the controllers is based on the linearized MM of the controlled plant and it offers good control system performance with respect to the modifications of the reference input.

The digital simulation results presented for four operating points validate the fuzzy control solutions. They show the improvement of the control system behavior for some values of the reference input and for some operating points.

The future research will be focused on the improvement of the performance indices. Other controlled plants and controller structures will be involved [26].

Acknowledgment

This work was supported by the CNMP and CNCSIS of Romania. This work was partially supported by the strategic grant POSDRU 6/1.5/S/13 (2008) of the Ministry of Labor, Family and Social Protection, Romania, co-financed by the European Social Fund – Investing in People.

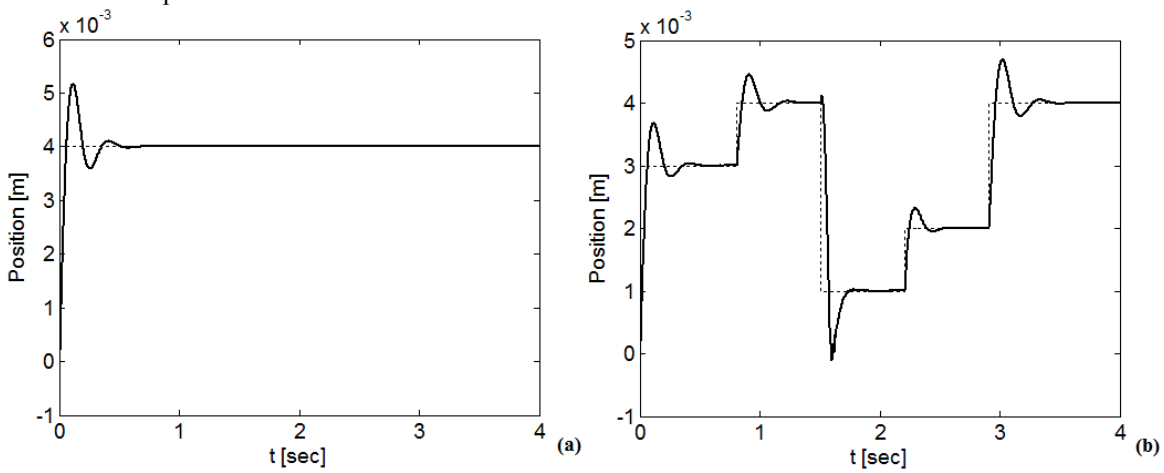


Fig. 4. Behavior of control system with Takagi-Sugeno 1 DOF fuzzy controller.

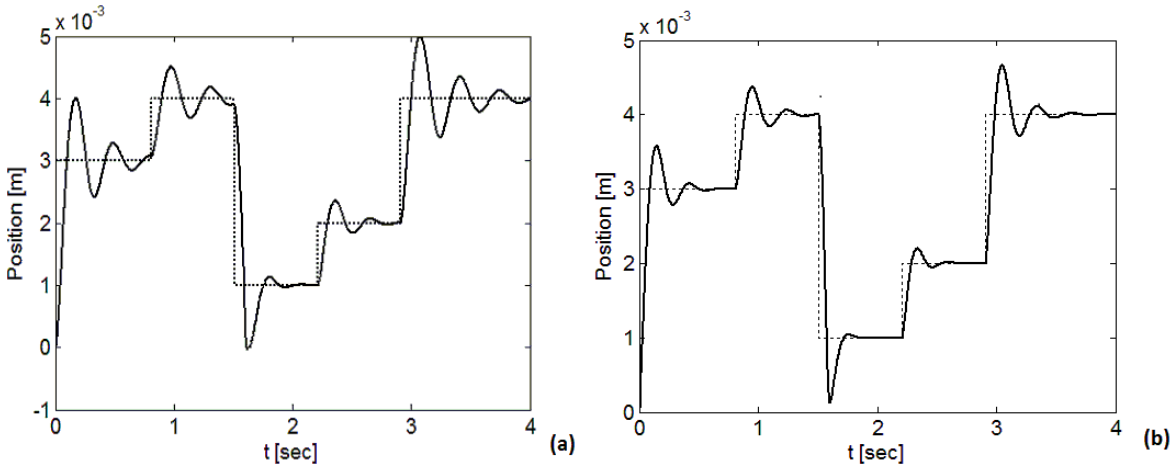


Fig. 5. Behavior of control system with PID (a) and Takagi-Sugeno 2 DOF fuzzy controller (b) designed for the operating point 3.

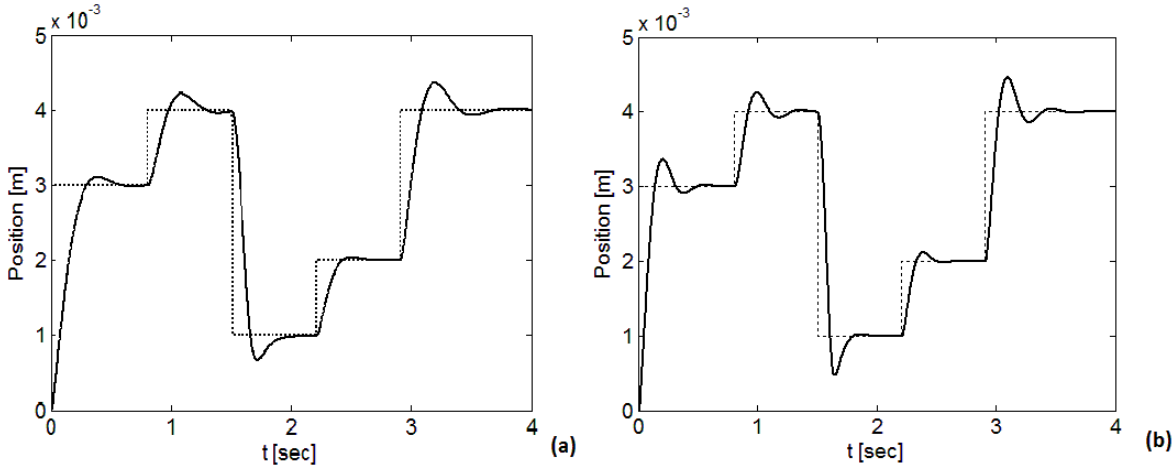


Fig. 6. Behavior of control system with PID (a) and Takagi-Sugeno 2 DOF fuzzy controller (b) designed for the operating point 5.

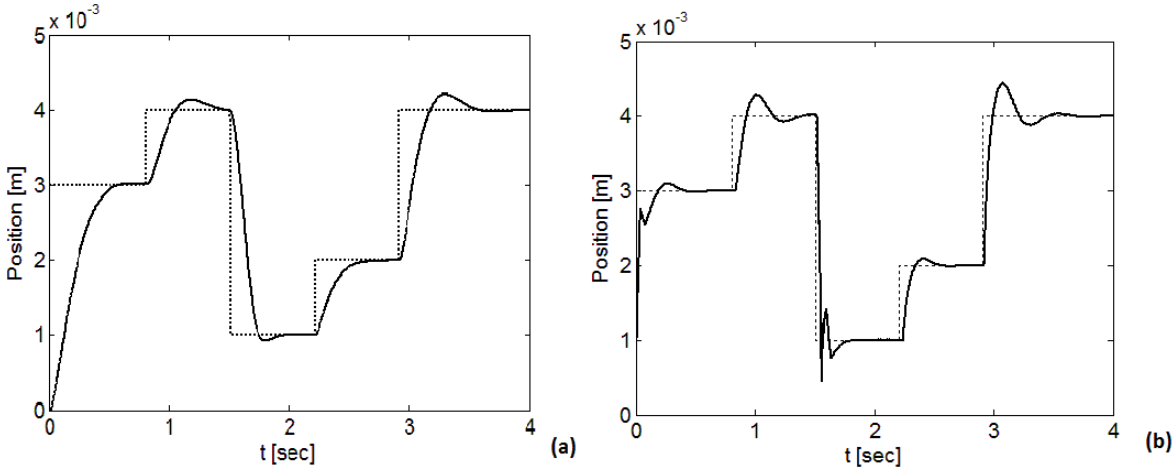


Fig. 7. Behavior of control system with PID (a) and Takagi-Sugeno 2 DOF fuzzy controller (b) designed for the operating point 6.

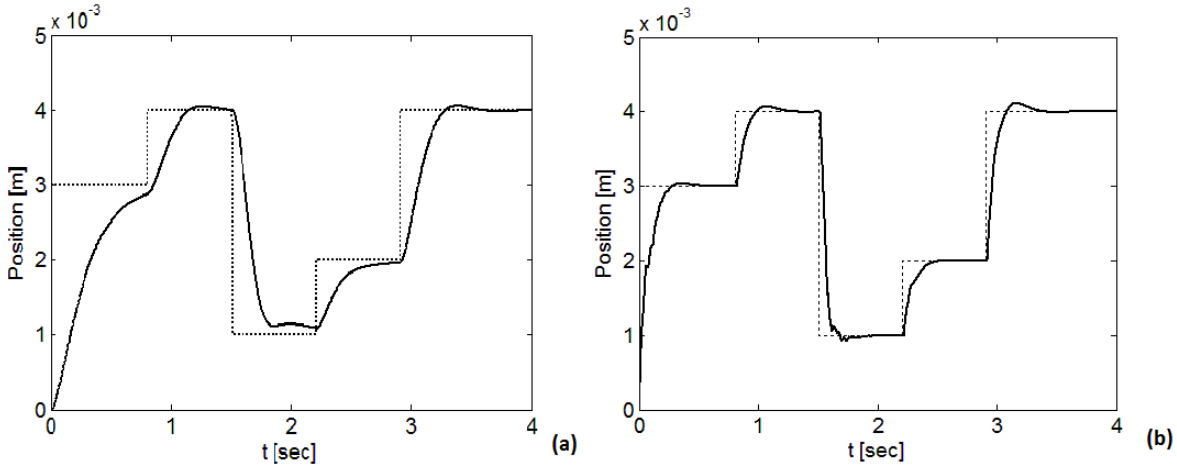


Fig. 8. Behavior of control system with PID (a) and Takagi-Sugeno 2 DOF fuzzy controller (b) designed for the operating point 7.

REFERENCES

[1] R. Isermann, *Mechatronic Systems: Fundamentals*. Berlin, Heidelberg, New York: Springer-Verlag, 2005.

[2] U. Kiencke and L. Nielsen, *Automotive Control Systems for Engine, Driveline and Vehicle*, 2nd ed. Berlin, Heidelberg, New York: Springer-Verlag, 2005.

[3] J. Zhang, J. Wang, and Z. Zhao, "A novel two-degree-of-freedom PID controller for integrator and dead Time Process," in *Proc. 6th World Congr. Intelligent Control and Automation (WCICA 2006)*, Dalian, China, 2006, vol. 2, pp. 6388–6391.

[4] S. Di Cairano, A. Bemporad, I. Kolmanovsky, and D. Hrovat, "Model predictive control of magnetic automotive actuators," in *Proc. 2007 American Control Conference (ACC '07)*, New York, NY, USA, 2007, pp. 5082–5087.

[5] J. Ma and Y. Yao, "Robust two-degree-of-freedom control design for a flight motion simulator," in *Proc. 7th Asian Control Conf. (ASCC 2009)*, Hong Kong, 2009, pp. 320–325.

[6] L. Horváth and I. J. Rudas, *Modeling and Problem Solving Methods for Engineers*. Burlington, MA: Academic Press, Elsevier, 2004.

[7] I. Škrjanc, S. Blažič, and O. E. Agamennoni, "Identification of dynamical systems with a robust interval fuzzy model," *Automatica*, vol. 41, pp. 327–332, Feb. 2005.

[8] J. Deur, J. Petri, J. Asgari, and D. Hrovat, "Recent advances in control-oriented modeling of automotive power train dynamics," *IEEE/ASME Trans. Mechatronics*, vol. 11, pp. 513–523, Oct. 2006.

[9] Z. C. Johanyák, D. Tikk, S. Kovács, and K. K. Wong, "Fuzzy rule interpolation Matlab toolbox - FRI toolbox," in *Proc. 15th Intl. Conf. Fuzzy Systems (FUZZ-IEEE'06)*, Vancouver, BC, Canada, 2006, pp. 1427–1433.

[10] I. Harmati, "Urban traffic control and path planning for vehicles in game theoretic framework," in *Robot Motion and Control 2007*, K. Kozlowski, Ed. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 437–444.

[11] R. E. Haber, R. Haber-Haber, A. Jiménez, and R. Galán, "An optimal fuzzy control system in a network environment based on simulated annealing. An application to a drilling process," *Appl. Soft Comput.*, vol. 9, pp. 889–895, Jun. 2009.

[12] V. E. Oltean, "On qualitative behaviours of a class of piecewise-linear control systems (Part II: A case study)," *Rev. Roum. Sci. Techn. - Électrotechn. et Énerg.*, vol. 54, pp. 205–212, Jun. 2009.

[13] R. Dobrescu, V. E. Oltean, and M. Dobrescu, "Simulation models and Zeno path avoidance in a class of piecewise linear biochemical processes," *Control Eng. App. Inf.*, vol. 10, pp. 33–39, Mar. 2008.

[14] C.-Y. Chen, T.-H. S. Li, and Y.-C. Yeh, "EP-based kinematic control and adaptive fuzzy sliding-mode dynamic control for wheeled mobile robots," *Inf. Sci.*, vol. 179, pp. 180–195, Jan. 2009.

[15] J. Vaščák, "Using neural gas networks in traffic navigation," *Acta Technica Jaurinensis, Series Intelligentia Computatorica*, vol. 2, pp. 203–215, Dec. 2009.

[16] X. Li, G. M. Dimirovski, Y. Jing, and S. Zhang, "A Q-learning model-independent flow controller for high-speed networks," in *Proc. American Control Conference (ACC '09)*, St. Louis, MO, USA, 2009, pp. 1544–1548.

[17] A. E. Bălău, C. F. Căruntu, D. I. Pătrașcu, C. Lazăr, M. H. Matcovschi, and O. Păstrăvanu, "Modelling of a pressure reducing valve actuator for automotive applications," in *Proc. 3rd IEEE Multi-conf. Systems and Control (MSC 2009)*, Saint Petersburg, Russia, 2009, pp.1356–1361.

[18] L. Kovács, A. György, B. Benyó, and A. Kovács, "Type 1 diabetes regulated by ANFIS at molecular levels," in *Proc. World Congr. Medical Physics and Biomedical Engineering (WC'09)*, Munich, Germany, pp. 841–844, Sep. 2009.

[19] D. I. Pătrașcu, A. E. Bălău, C. F. Căruntu, C. Lazăr, M. H. Matcovschi, and O. Păstrăvanu, "Modelling of a solenoid valve actuator for automotive control systems," in *Proc. 17th Intl. Conf. Control Systems and Computer Science (CSCS-17)*, Bucharest, Romania, 2009, pp. 541–546.

[20] S. Di Cairano, A. Bemporad, I. V. Kolmanovsky, and D. Hrovat, "Model predictive control of magnetically actuated mass spring dampers for automotive applications," *Int. J. Control*, vol. 80, pp. 1701–1716, Nov. 2007.

[21] C.-A. Dragoș, S. Preitl, and R.-E. Precup, "Low cost Takagi-Sugeno fuzzy controller for an electromagnetic actuator," *Sci. Bull. "Politehnica" Univ. Timişoara, Romania, Trans. Automatic Control Comp. Sci.*, vol. 54(68), pp. 87–92, June 2009.

[22] C.-A. Dragoș, "Study concerning model based predictive control," PhD Report 1, "Politehnica" Univ. Timişoara, Timişoara, Romania, 2009 (in Romanian).

[23] C.-A. Dragoș, "Study concerning the modeling of nonlinear processes and control solutions," PhD Report 2, "Politehnica" Univ. Timişoara, Timişoara, Romania, 2009 (in Romanian).

[24] C. Lazăr et al., "Real-time informatics technologies for embedded-system-control of power-train in automotive design and applications," Research Report 1 of the SICONA CNMP Grant, "Gh. Asachi" Tech. Univ. Iaşi, Iaşi, Romania, 2009 (in Romanian).

[25] S. Preitl, R.-E. Precup, and Z. Preitl, "Two-degree-of-freedom fuzzy controllers: Structure and development," in *Proc. Intl. Conf. In memoriam John von Neumann*, Budapest, Hungary, 2003, pp. 49–60.

[26] C. Lazăr et al., "Real-time informatics technologies for embedded-system-control of power-train in automotive design and applications," Research Report 2.1 of the SICONA CNMP Grant, "Gh. Asachi" Tech. Univ. Iaşi, Iaşi, Romania, 2009 (in Romanian).

On Extracting of Fuzzy Rules from High-Dimensional Heart Disease Databases by Neuro-Fuzzy Systems

V. Dugan, R. Solea, *Member, IEEE*, and A. Filipescu

***Abstract*— The paper is a trial, an experiment, to see how can be extract fuzzy *"if-then"* rules from data sets with many inputs (here *"The Heart Diseases Data Set"*-HDDS), using a hybrid approach of adaptive neuro-fuzzy system (anfis). In other words, has been tested the capabilities of anfis to obtain these rules that are easy to understand, verify and extend. The rule extraction method has been based on estimating clusters in the data, each cluster obtained corresponding to a fuzzy rule that relates a region in the input space to an output region. After the number of rules and initial fuzzy inference system are obtained, the rule parameters are optimized with anfis, using a hybrid method (backpropagation/ gradient descent and r.m.s. error). Unfortunately, the estimated clusters in our data with many inputs are too many, as a consequence the fuzzy rules are too numerous and, finally (as a general conclusion), for this case of the HDDS, the classical way of an expert physician is a better one.**

I. Introduction

HUMANS collect, store, add, update, remove and retrieve information about all cognitive and comprehension human skills and problems in databases, i.e. collections of data. In these databases, most attribute values of objects are numerical, to be able to obtain good information or decision in the future, or when it is necessary. But, humans use natural language to socialize or to communicate information, and because this, often linguistic *"if-then"* rules human content for any set of data would be very desirable. The fuzzy rules extracted from data allows relationships in the volume of numerical values of data attributes, and can be easy used and verify.

Much more: in the applications with high-dimensional inputs for any databases, a human expert cannot often express knowledge explicitly, and an automatically extract fuzzy rules from experimental input/output data is a good alternative.

Rules are a good way of representing information or bits of knowledge. This concept uses a set of *if-then* (conditional) rules for classification. Because in the rule antecedent (precondition; *"if"*-part), the condition consist if one or more *attribute tests* (in our data base are twelve, logically ANDed), any rule is assessed by its *coverage* and *accuracy*. For example, if a rule R *covers* n_{cov} of tuples from a data set D (with |D| tuples), and classify correct the n_{corr} of tuples from D, the coverage (*cov*) and accuracy (*acc*) of R can be defined as $cov(R) = n_{cov}/|D|$ (i.e. percentage of tuples covered by R), and $acc(R) = n_{corr}/n_{cov}$ (i.e. percentage of tuples correctly classify by R, from n_{cov}).

From a database, the process of automatic discovery of unknown patterns, rules and other regular contents implicitly present in the volume data, is known as *Knowledge Discovery in Databases (KDD)*. Other title, *Data Mining (DM),* denotes discovery of patterns in database, being often used as a synonym for KDD. In this moment there exists some KDD commercial tools purpose-built, as *CiteSeer* (an useful and interesting custom-digital-library generator, in a general research area from the World Wide Web, [1],), *JAM* (Java Agents for Metalearning) tool (which analyses huge number of credit card transactions processed on the Internet daily, [2]), *Recon* (a stock-selection advisor to buy or not a stock, after a database of over 1000 companies successfuly is analyzed, [3]), but a general-purpose tool not yet.

Other known KDD techniques are the *association rules*, used in [4], technique which relate the presence of items in transactions and not only.

In this paper, to find *if-then* rules in a Heart Disease Data Set, from UCI ML Repository [5], a hybrid neuro-fuzzy system, *anfis*, and some clustering methods in off-line mode have been used. The main purpose was to see if it is possible or not to obtain *if-then* rules using anfis, i.e. to obtain these rules in an automatic mode.

Because in the mentioned heart disease database (with twelve inputs or more) we do not have a clear idea how many clusters (and *if-then* rules) there should be for this set of data, was used *subtractive clustering,* [6]. This method is a fast, one-pass algorithm, to estimating the number of clusters and the cluster centers, in any set of data. The cluster estimates obtained were used to initialize an iterative optimization-based clustering method, *fuzzy C-means*, [7], and to generate a Sugeno-type fuzzy inference system that best models the data behavior using a minimum number of rules. The rules partition themselves according to the fuzzy qualities associated with each of the data clusters.

In other words, to obtain the fuzzy rules from the heart diseases data, these were covered following the steps: (1)

Manuscript received May, 2010.

V. Dugan is with the Department of Control Systems and Industrial Informatics, Computer Science Faculty "Dunarea de Jos" University of Galati, Domneasca 47, 800008, Galati, Romania (corresponding author to provide phone/fax: +40-236-460182; e-mail: viorel.dugan@ugal.ro).

R. Solea, was with the Department of Control Systems and Industrial Informatics, Computer Science Faculty "Dunarea de Jos" University of Galati, Domneasca 47, 800008, Romania (e-mail: razvan.solea@ugal.ro).

A. Filipescu, was with the Department of Control Systems and Industrial Informatics, Computer Science Faculty "Dunarea de Jos" University of Galati, Domneasca47, 800008, Romania(e-mail: adrian.filipescu@ugal.ro).

Clustering of (training) heart diseases data set with *subtractive* algorithm to automatically determine the number of clusters, (2) *Generation of the initial if-then fuzzy rules,* for each cluster found at step one, and (3), *Optimization of the fuzzy rules.* In the third step has been used a neuro-adaptive learning technique, i.e. a method to learn information about the data set. The learning technique compute the membership function parameters that best allow the associated fuzzy inference system to track the given I/O data, and works similarly to that of neural networks.

II. TOOLS USED

A. Clustering Techniques

To find similarities in data (i.e. if-then rules) in the amount of data in The Heart Disease Data Set [5], as in other fields (pattern recognition, computer vision etc.), has been used the clustering method as a pre-processing of data set. Clustering algorithms organize, categorize and discover the relevance knowledge in data. All clustering techniques have as approach the finding of cluster centers that represent each group from clusters. For some clustering techniques it necessarily to know from beginning the number of clusters, for others it is not necessary. As example, the techniques K-means clustering (sometimes named as hard C-means), and fuzzy C-means (fcm), are two techniques for which it is necessary the knowing of clusters number a priori, and as a consequence, when the number of clusters is not known, the above clustering methods cannot be used. For other methods, as mountain clustering or as a better alternative, subtractive clustering, is not necessary to have the number of clusters known from beginning. In these cases the algorithm initially finds the first large cluster, and then the second, and so on. Mountain and Subtractive clustering methods are of that type. The mentioned algorithms can be used in both types of problems. The clustering algorithms can be implemented any of on-line or off-line mode. In on-line clustering, all cluster centers are updated based on every new input sample, and the system learns where the (new) cluster centers are. In off-line mode, a training data set to find the cluster centers, and a checking data set to check ‘ the results’ of training data set are used. After the cluster centers from database are found, the if-then rules are later to be found. Subtractive clustering used in the paper being a similar but better version of mountain clustering, below are two short descriptions about both methods.

1) Mountain (function) clustering

This method was proposed by Yager and Filev, in [8]. Firstly, the data space is ‘cover’ with a grid, the intersection lines of the grid being the *potential* cluster centers, denoted below as a set ***C***. Then, a mountain function which is *a data density measure* is constructed. The peak of the mountain function at a point c $\square$ *C* is equal to

$$m(\boldsymbol{c}) = \sum_{i=1}^{N} \exp(-\|\boldsymbol{c}-\boldsymbol{x}_i\|^2/2\sigma^2) \quad (1)$$

where $\boldsymbol{x_i}$ is the i^{th} data point, and σ - a specific constant which determines the peak as well as the smoothness of the resultant mountain function (1). The eq. (1) shown that the *data density* measure at a point $\boldsymbol{c}$ is affected by all the points $\boldsymbol{x_i}$ in the data set and, is inversely proportional to the distance between the data points $\boldsymbol{x_i}$ and the point considered $\boldsymbol{c}$.

The next step involves selecting the cluster centers by sequentially destroying of the mountain function before. Because the first cluster center $\boldsymbol{c}_1$ is determined by selecting the point from grid with the greatest density measure, for the next cluster center the effect of the first cluster must be eliminated. For this, a new revised mountain function is formed by subtracting a scaled Gaussian function centered in $\boldsymbol{c}_1$:

$$m_{new}(\boldsymbol{c}) = m(\boldsymbol{c}) - m(\mathbf{c}_1)*\exp(-\|\boldsymbol{c} - \boldsymbol{c}_1\|^2/(2\sigma^2)) \quad (2)$$

After subtraction of $m(\mathbf{c}_1)\exp(-\|\boldsymbol{c} - \boldsymbol{c}_1\|^2/2\sigma^2)$ amount in eq.(2), the effect of the first cluster determined with (1) is reducing to zero, and the second cluster center is selected as the *intersection* line point having the greatest value for the new mountain function. The process is repeated until a sufficient number of cluster centers is attained.

2) Subtractive Clustering

Subtractive clustering, similar to mountain clustering, was proposed, [6], because the computation grows exponentially with the problem dimension. I.e., if the mountain function has to be evaluated at each grid point, the new proposed method, subtractive clustering, uses only data points as the candidates for cluster centers. Therefore, the computation is proportional to the problem size (positions of the data points) instead of the problem dimension (every possible position in the data space). Consequently the number of calculations for density function is significantly reduced.

In this method each data point being a candidate for a cluster center, a density measure at a data point x_i is defined as

$$Di = \sum i=1 n \exp(-\|xi-xj\|2/(r/2)2) \quad (3)$$

where *r* is a positive constant, namely neighborhood radius (cluster radius). Thus, a data point will have a high density value if it has many neighboring data points. After the first cluster center c1 is chosen as the xc1 data point, with the largest density value Dc1, the new density measure for each data point xi is modified similarly as in above (mountain) method:

$$Di = Di - Dc1*\exp(-\|xi-xc1\|2/(r*/2)2) \quad (4)$$

In eq. (4), r* is a positive constant to be possible a neighborhood that has measurable reductions in the density measure. Therefore, the points from data set near the first

cluster center xc1 will have a more reduced density measure. As in the first step, with the new density function (4), the next cluster centers are selected as the points having the greatest density value. The process continues until the possible number of clusters is obtained.

B. The Commercial Neuro-fuzzy (Anfis) System

The cluster estimates obtained above were used to generate a Sugeno-type fuzzy inference system that best models the data behavior using a minimum number of rules. For the third step above (*Optimization of the fuzzy rules),* a neuro-adaptive learning technique can be used. From the some *commercial* neuro-fuzzy systems, to adjustment of membership function (MF) parameter, was used the **anfis** (adaptive neuro- fuzzy inference system (and function)) from Matlab [9], [10]. ANFIS constructs a fuzzy inference system (FIS) using a given input/output data set, and to identify for training the membership function parameters of single-output Sugeno type, the system use a hybrid combination of least-squares and backpropagation gradient descent methods [11]. Thus, it is possible to model the given heart disease data set of input/output data. In other words, the initially FIS is a model that maps input characteristics to input membership functions, input membership function to rules, rules to a set of output characteristics, output characteristics to output membership functions, and the output membership function to a single-valued output or a decision associated with the output.

C. The Heart Diseases Data Set

An abstract of *The Heart Diseases Data Set* (HDDS) can be seen below. In reality the data set is a set of four databases: Cleveland Clinic Foundation (USA), Hungarian Institute of Cardiology, University Hospitals (Zurich & Basel) Switzerland, and the VA Medical Center Long Beach (USA).

The database contains 75 attributes, but all published experiments refer to using a subset of 14 of them. In particular, our experiments used only a subset of 12 of them. The "goal" field refers to the presence of heart disease in the patient. It is integer valued from 0 (no presence, value 0) to 4 (values 1, 2, 3, 4, presence).

Data set characteristics are: the number of instances, which is 303, these being multivariate. The attribute (inputs) characteristics are: initial number 75 (in our paper only 12), being of type categorical, integer and real. From all important risk factors as: biological determinant (blood pressure, serum cholesterol, glicemy, weight), life style determinant (alcohol abuse, type of diet, physical activity, smoker), and general determinant (age, sex, genetic predisposition, education, ethnic, work condition), only 12 were used.

Attribute Information is for only 12/75 attributes which was used (numbers #3, #4,…, #n, being the numbers in the complete attribute documentation):

1. #3 (**age**: in years); **2.** #4 (**sex**: 1 = male; 0 = female); **3.** #9 (**cp:** chest pain type: Value 1: typical angina; Value 2: atypical angina; Value 3: non-angina pain; Value 4: asymptomatic); **4.** #10 (**trestbps**: resting blood pressure in [mm Hg], on admission to the hospital); **5.** #12 (**chol:** serum cholesterol in [mg/dl]); **6.** #16 (**fbs:** fasting blood sugar > 120 [mg/dl]: 1 = true; 0 = false); **7.** #19 (**restecg:** resting electrocardiographic results: Value 0: normal; Value 1: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 [mV]); Value 2: showing probable or definite left ventricular hypertrophy by Estes' criteria); **8.** #32 (**thalach:** maximum heart rate achieved); **9.** #38 (**exang:** exercise induced angina: 1 = yes; 0 = no); **10.** #40 (**oldpeak:** = ST depression induced by exercise relative to rest); **11**. #41 (**slope:** the slope of the peak exercise ST segment: Value 1: up sloping; Value 2: flat; Value 3: down sloping); **12.** #44 (**ca:** number of major vessels (0-3) colored by fluoroscopy. Some attributes (risk factors), more or less important, as **cigs**-cigarettes per day; **years**-number of years as a smoker, etc); above #13; #14, etc., were missed. To see detail about what are the influences of each attribute, the reader can go to references [12], [13] etc.

III. Simulations and Remarks

To can obtain "if-than" rules from *The Heart Diseases Data Set* in automatic mode using 'anfis'- GUI (Graphical User Interface), the following steps have been followed: (1) the initial database, [5], has been loaded in Mat lab workspace (the **'**load' function), see Figure 1; (2) several *training data* have been chosen (here we have only considered the case of 75 training data out of 303), see Figure 2; (3) the rest of the data have been chosen as *checking data* in a similar way, see Figure 3; (4) the training data from (2) have been loaded in 'anfis', and after, (5) - has been generated the first (initial) fuzzy inference system (FIS) using subtractive clustering: was obtained 72 clusters, i.e. 72 if-then rules (Figure 4, where inputs was used 'gaussmf'as membership functions, and constant and linear functions for outputs); (6) for the optimal training of the neural network the hybrid learning method has been used (i.e. back propagation with different epochs, and r.m.s. for error); the Anfis model structure obtained can be seen in Figure 5 (72 clusters using *subtractive* clustering, 72 fuzzy rules, which are too many, and the number of data is smaller than number of modifiable parameters (936)); (7) the verification of the correct training of the neural networks with training data from point (3), is shown in the Figure 6.

The same result has been obtained when was used "genfis2**"** function in command-line, using *subtractive* clustering, to generate an initial FIS for Anfis training. Some of the surfaces obtained with the Anfis viewer for combinations of two inputs can be shown in the Figures 7-10. As inputs has been used some of the two combinations

with the smallest errors between training and checking errors, i.e. from the *pain-pulse; pain-segST; pain-slopeST; pain- fluoroscopy; pulse-segST; pulse- fluoroscopy; angina-fluoroscopy; segST- fluoroscopy etc etc*.

Because the number of clusters and the "*if-than*" rules obtained after the cluster process are too many (72 clusters compare with 75 training data) and, on the other part, the number of data is smaller than number of modifiable parameters, we concluded that the databases with several entries (*The Heart Diseases Data Set,* has 12 attributes!), cannot be rightly managed by 'anfis'. That is to say that the *"if- then*" obtained rules, though correct, cannot be used as such: they are too numerous and the experience of the physician is then necessary. From this point of view, also considering that in *The Heart Diseases Data Set* case not all inputs/ attributes necessarily present a cardiac disease as an output, have been selected the best input (s) (out of 12), that produces the 'best' output (i.e. the least output error). In Matlab, more precisely in neuro-fuzzy modeling, an exhaustive search to selection of one (or more) input(s) from a set of input candidates, is used the 'exhsrch' function.

We used this function thinking that is preferable a model with a simple structure for better generalization, not one with too many rules. The 'exhsrch' function performs the exhaustive search on selecting 1 to (max) 4 inputs from all input candidates (12 in the paper), and builds up an anfis model for each input or input combination, trains the neural network corresponding to the pattern for one epoch only, then compares and shows up the obtained results: the most influential input/ attribute (or input combination of attributes) is the one that gives the smallest error.

The number of `anfis` models (of combinations!) determined by the '`exhsrch`' function, can be obtained by using the formula $\mathbf{C_n^m = n! / (n-m)!}$ (where "**n**" – is the inputs/ attributes number (n = 12), **m** – the combination number, - the factorial sign) and there are: $C_n^1 = 12$ (Figure 11); $C_n^2 = 66$ (Figure 12); $C_n^3 = 220$ (without figure); $C_n^4 = 11880$ (without figure).

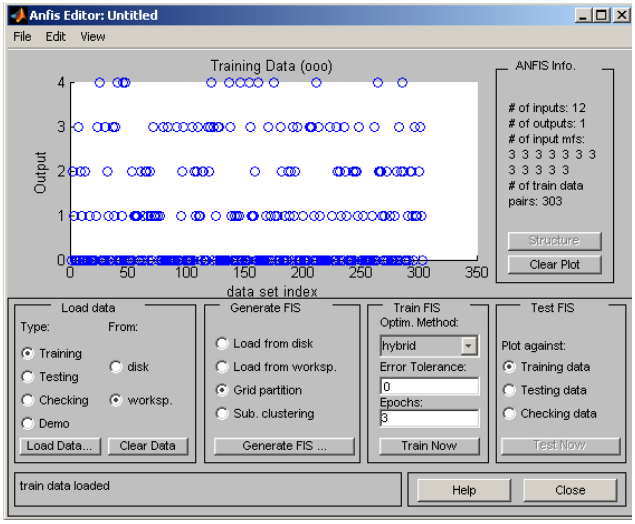


Fig.1: The data from The Heart Disease Data Set

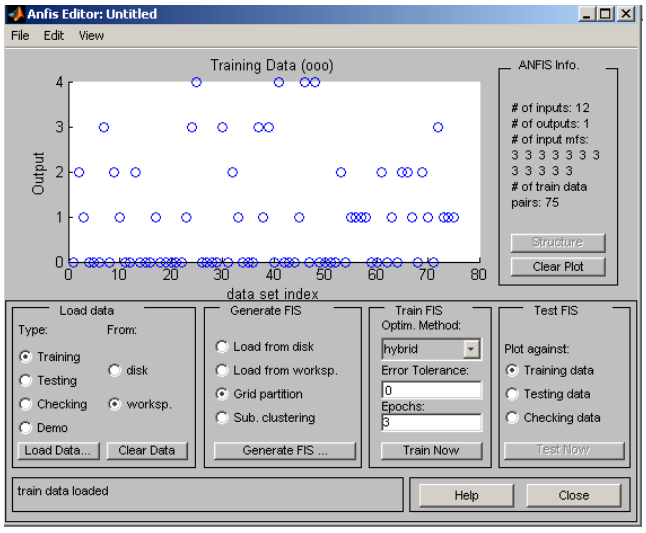


Fig.2: The 75 training data from The Heart Disease Data Set

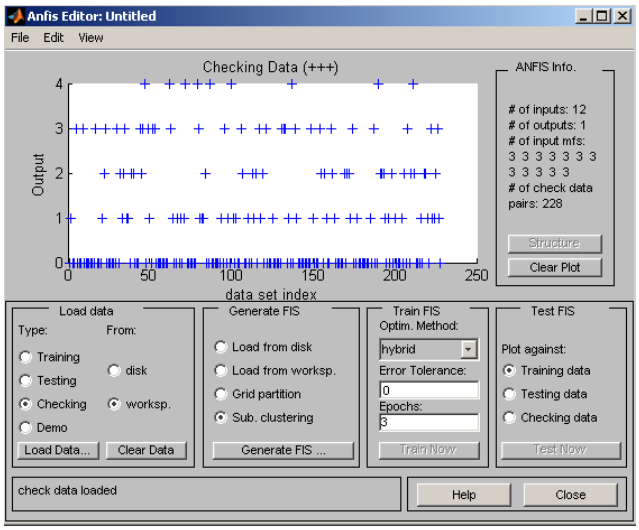


Fig.3: The 228 checking data from The Heart Disease Data Set

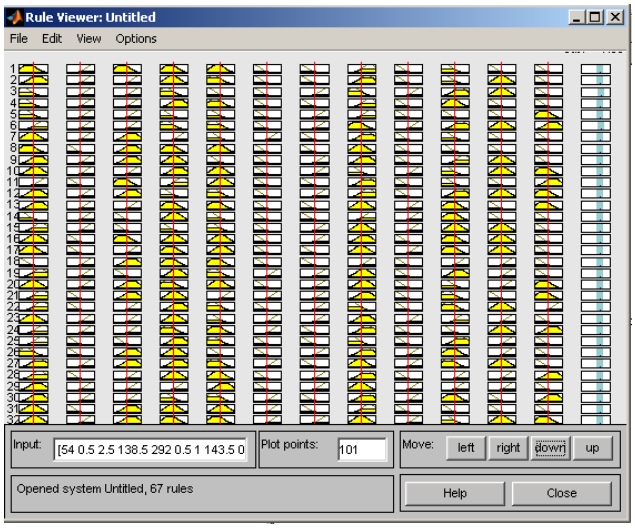


Fig.4: The 72 *'if-than'* rules finded for each cluster in the training data

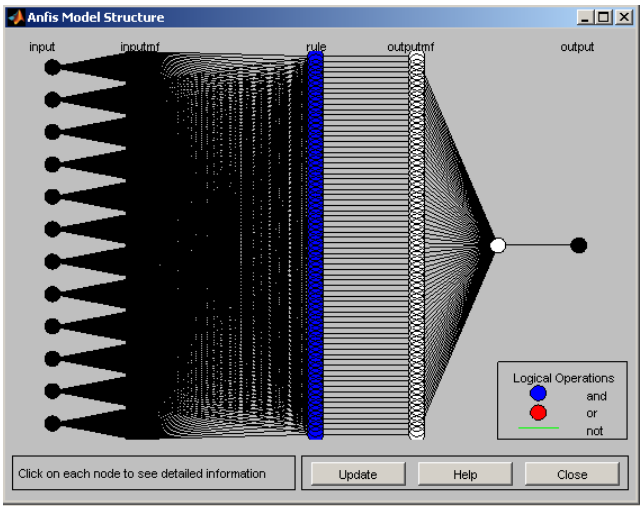


Fig. 5: Anfis model structure (12 inputs, 1 output, and 72 rules)

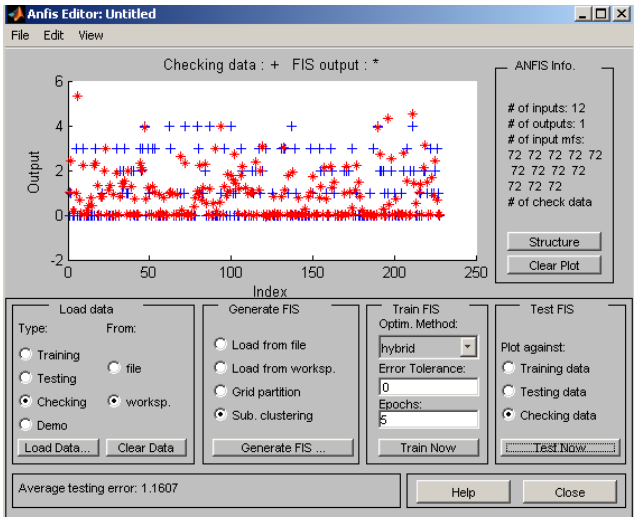


Fig. 6: The verification of the neural networks correct training with checking data (+) and FIS output (*).

The greatest influence in Figure 11 (the smallest error) is for the third input/attribute, *the pain*, but in the Figure 12 the smallest errors between training and checking errors are for the following pairs: *pain-pulse; pain-segST; pain-slopeST; pain- fluoroscopy; pulse-segST; pulse- fluoroscopy; angina-fluoroscopy; segST- fluoroscopy etc etc*.

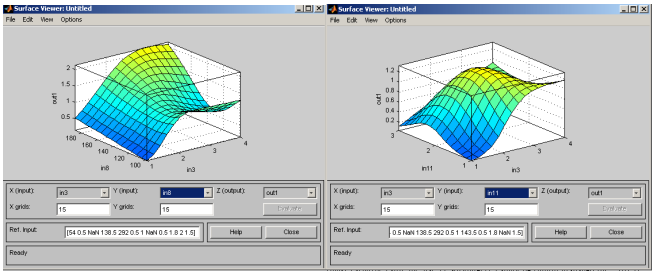


Fig. 7: Surface of *in3* (pain) and *in8* (pulse rate)

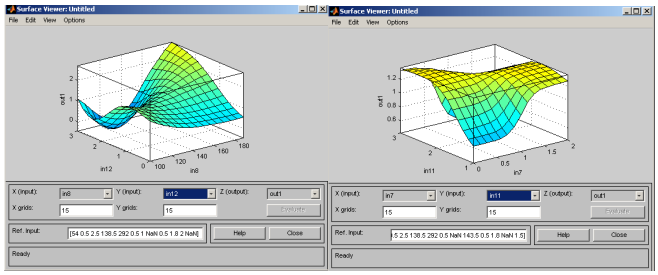


Fig. 8: Surface of *in3* (pain) and *in11* (slope ST segment)

Fig. 9: Surface of *in8* (pulse rate) and *in12* (fluoroscopy)

Fig. 10: Surface of *in7* (ST wave) and *in11* (slope ST segment)

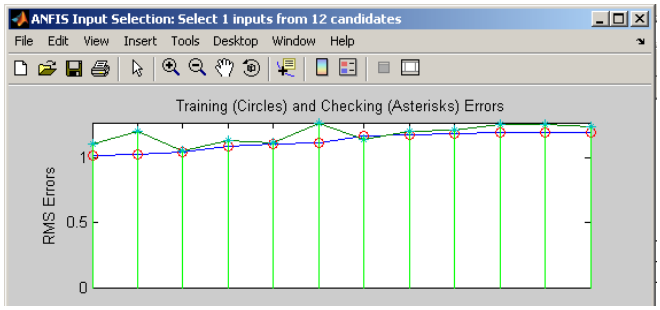


Fig. 11: The Anfis input selection: 1 input from 12 attributes.

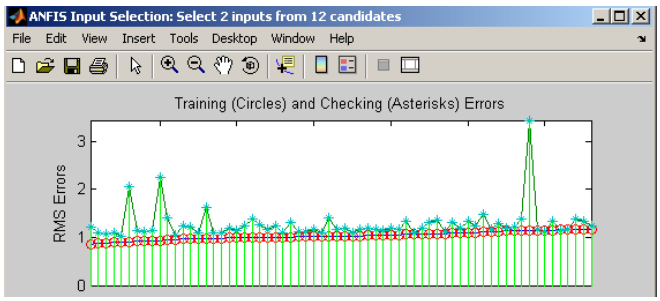


Fig. 12: The 66 combinations of 2 inputs from 12 attributes

In the case of the 220 combinations of 3 inputs from 12 attributes, the greatest influence has the combinations between inputs 1, 10 and 12, and for the 4 inputs, the 1+4+8+10 combinations (the last two, without figures). Although these input/ attribute combinations are the most relevant and shown that a cardiac disease exist however, they not is a guarantee for a good diagnostic. A good (correct) diagnostic is one based on *all* (or more four) inputs/ attributes. `'Exhsrch'` function can do search only for 4 inputs. But, in this case, we have seen that are too much *if-then* rules, and it is necessary a physician. An alternative has been to use some clusters (3, 4, and 5) among the 'good' pairs obtained in the case of C_n^2, Figure 12. Has been clustered both training and checking data, and has been seen the influence of the cluster number on the system performance (errors). Although the rules obtained are good rules, however, not being a physician in our team, they have not been generalized.

IV. CONCLUSIONS

The paper, which is an experiment to see how can be extract fuzzy *"if-then"* rules from data sets with many inputs (here *"The Heart Diseases Data Set"* from [5]), using a hybrid neuro-fuzzy approach. In this aim has been used and tested to see the capabilities of the adaptive neuro-fuzzy system `anfis`, from Matlab. The rule extraction has been based on estimating clusters in the data by *subtractive* clustering method with different ranges (radii) of influence. Although all rules are good rules (not only from technical point of view, but as a good diagnosis), however are too many. Because this, to builds up anfis models for each input or input combination, has been used the `'exhsrch'` function to perform an exhaustive search on selecting 1 to (max) 4 inputs from all input candidates (12 in the HDDS). This function gives us the most influential attribute or combinations of two, three, four attributes (that gives the smallest errors).

To identify and tune/training the membership function parameters of Sugeno type fuzzy inference system from `Anfis` (i.e. to model the given set of I/O data from HDDS), in all our above trials has been used the hybrid method (least-squares and backpropagation gradient descent method).

Some final and general conclusions can be seen in the following. The `Anfis` and `exhsrch` tools are very strong. Although the rules obtained in this case (HDDS) are too many, all are good rules. However, when `Anfis` and `exhsrch` are used in the data bases which imply the people, or (much more) heart diseases as in HDDS, they must be used carefully. In our opinion, is necessary in the research team and an expert physician, to can be possible the rules particularization and generalization.

REFERENCES

[1] K. D. Bollacker, S. Lawrence, and C. Lee Giles, "Discovering relevant scientific Literature on The Web", *IEEE Intelligent Systems*, vol. 15 (2), pp. 42–47, 2000.

[2] P.K. Chan, W. Fan, A. L. Prodromidis and S. J. Stolfo, "Distributed data mining in credit card fraud detection", *IEEE Intelligent Systems,* vol. 14 (6), pp. 67-74, November 1999.

[3] G. H. John, P. Miller and R. Kerber, "Stock selection using rule induction", *IEEE Expert*, pp. 52-58, October 1996.

[4] S. Konias, G. D. Giaglis, G. Gogou, P. D. Bamidis and N. Maglaveras, ***"Uncertainty Rule Generation on a Home Care Database of Heart Failure Patients"***, *Computers in Cardiology*, vol. 30, pp. 765−768, 2003.

[5] "*University of California at Irvine* (UCI) *Machine Learning Repository"*, retrieved May 2010, from http://www.ics.uci.edu/~mlearn/MLRepository.html .

[6] S. Chiu, "Fuzzy Model Identification Based on Cluster Estimation" *Journal of Intelligent & Fuzzy Systems*, vol. 2(3), Sept. 1994.

[7] J. C. Bezdec, "*Pattern Recognition with Fuzzy Objective Function Algorithms*", Plenum Press, New York, 1981.

[8] R. Yager and D. Filev, "Generation of Fuzzy Rules by Mountain Clustering", *Journal of Intelligent & Fuzzy Systems*, vol. 2 (3), pp. 209-219, 1994.

[9] J. -S. R. Jang, "ANFIS: Adaptive Neural-based Fuzzy Inference Systems", *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 23 (3), pp. 665-685, May 1993.

[10] J.-S. R. Jang, and C.-T. Sun, "*Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*", Prentice Hall, 1997.

[11] Mathworks, User's guide of 7th printing for Matlab 7.0.4 (Release 14P2), June 2005.

[12] D. Dubin, "*Fast Interpretation of the Electrocardiography (ECG)"*, Cover Publishing Co. Florida, USA & Ed. Medicala, Bucuresti, 1999.

[13] B. J. Gersh. (Ed), *"Mayo Clinic Heart Book"*, 2nd ed., Mayo Foundation Press, 2000.

Sliding Mode Control of Four Driving-Steering Wheels Autonomous Vehicle

Bogdan Dumitrascu and Adrian Filipescu

***Abstract*—In this paper a lateral motion control using a sliding-mode controller (SMC) for Four Driving-Steering Wheels (4DW/SW) vehicle is presented. The lane centerline following is the main control performance. The dynamic model of the vehicle has been taken into account. Closed-loop robustness to uncertainties is achieved. The lane centerline following by look-ahead techniques is obtained. Closed-loop simulation and real time results with 4DW/SW Seekur robot base prove the efficiency of sliding-mode control. Versus 2DW and 2SW mobile robots control, 4DW/SW sliding-mode control is more stable with smaller lateral error**.

I. INTRODUCTION

For many years, the control of non-holonomic vehicles has been a very active research field. At least two reasons account for this fact. On one hand, wheeled vehicles constitute a major and ever more ubiquitous transportation system. Previously restricted to research laboratories and factories, automated wheeled vehicles are now envisioned in everyday life (e.g. through car-platooning applications or urban transportation services), not to mention the military domain.

For autonomous vehicles, two control tasks arise. The first task, longitudinal control, involves controlling the vehicle speed to maintain a proper spacing between vehicles.

This paper concentrates on the second task, lateral control, which is concerned with automatic steering of vehicles for lane keeping to follow a reference along the lane center.

Steering control of vehicles has been studied since late 1950's. Steering control problem requires addressing two parts: sensing and control [1] - [3]. Steering control approaches can be grouped into look-ahead and look-down systems. Look-ahead systems replicate human driving behavior by measuring the lateral error ahead of the vehicle. A number of research groups have successfully conducted highway speed experiments with look-ahead systems like machine vision and laser. Another approach is the look-down system which measures the lateral displacement at a location within or in the close vicinity of vehicle boundaries, typically straight down the front bumper. Look-ahead systems replicate human driving behavior by measuring the lateral displacement ahead of the vehicle. The look-ahead distance usually is increased with increasing velocity, similar to human behavior. This paper deals with the kinematics and dynamics models and the feedback control of an autonomous wheeled robot named Seekur (Fig.1) from the Mobile Robots Inc [4].

Fig. 1. Seekur - unmanned ground vehicle.

Outdoor robots face all of the same challenges as indoor robots, such as sensing, data processing, locomotion, navigation, and interaction with the surroundings. Outdoor robots, however, are expected to achieve all of these things in much more complex and unstructured environments such as forests, deserts, and even agricultural fields [5]-[8].

Variable structure control (VSC) has been showing to be a robust approach in different applications and has been successfully applied in control problems as diverse as automatic flight control, control of electric motors, regulation in chemical processes, helicopter stability augmentation, space systems and robotics. One particular type of VCS system is the sliding mode control (SMC) methodology [9]. The theory of SMC has been applied to various control systems, since it has been shown that this nonlinear type of control exhibits some excellent properties, such as robustness against large parameter variation and disturbances [10] - [14]. By designing switch functions of state variables or output variables to form sliding surfaces, SMC can guarantee that when trajectories reach the surfaces, the switch functions keep the trajectories on the surfaces, thus yielding the desired system dynamics. The main advantages of using SMC include fast response, good transient and robustness with respect to system uncertainties and external disturbances.

II. LATERAL CONTROL PROBLEM FOR VEHICLES

The control objective in lateral control of vehicles for lane following can be seen as keeping the distance of a point (or

This work of Bogdan Dumitrascu was supported by Project SOP HRD - EFICIENT 61445.

This work of Adrian Filipescu was supported by CNCSIS-UEFISCSU, project number PNII-IDEI 641/2007.

Bogdan Dumitrascu is PhD Student in Control Systems, Faculty of Computer Science, University "Dunarea de Jos" of Galati, Domneasca 47, 800008, Galati, Romania (e-mail: dumi_b20@yahoo.com).

Prof. Adrian Filipescu is with the Control Systems and Industrial Informatics Department, Faculty of Computer Science, University "Dunarea de Jos" of Galati, Domneasca 47, 800008, Galati, Romania (e-mail: adrian.filipescu@ugal.ro).

its vertical projection on the road) from the road centerline zero.

The vehicle model for control design and the control objective is described in the following subsection.

A. Vehicle Model

TABLE I
REFERENCE VEHICLE PARAMETERS

Parameter	Value
Vehicle mass (m)	350 kg
Yaw moment of inertia (I_ψ)	725 kg * m^2
Front axle to CG length (l_f)	0.70 m
Rear axle to CG length (l_r)	0.70 m
Front cornering stiffness (C_f)	17.000 N/rad
Rear cornering stiffness (C_r)	17.000 N/rad

A complex vehicle model includes six degrees of freedom for the vehicle sprung mass, four wheel states, and two engine states. Additionally, the brake, throttle, and steering actuators are modeled as linear first-order systems. The suspension system is modeled as four independent spring-damper systems. From the complex model, we derive the simplified model to design a sliding-mode controller under the following assumptions:

- neglect the roll, pitch, and vertical motion;
- approximate the normal forces acting on tires as static values;
- discount the actuator dynamics (for brake, throttle, and steering).

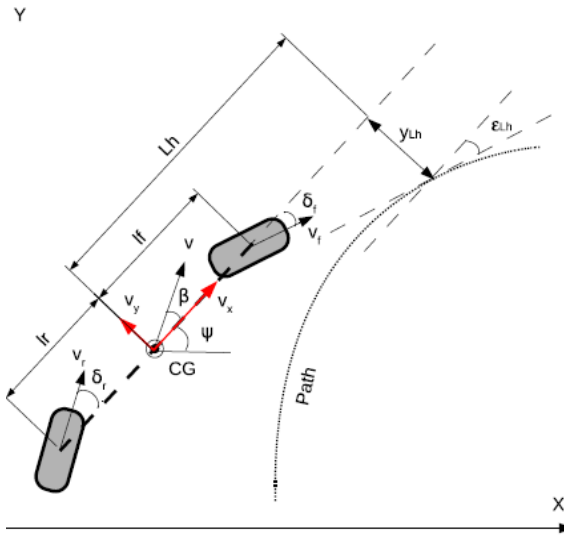


Fig. 2. Linear 2 DOF bicycle mode

For a linear 2 DOF bicycle mode (Fig. 2, when the front and rear steering angles δ_f and δ_r are small, the lateral velocity (v_y) and yaw moment ($r = \dot{\psi}$) are described by the following equation:

$$\begin{bmatrix} \dot{v}_y \\ \dot{r} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} * \begin{bmatrix} v_y \\ r \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} * \begin{bmatrix} \delta_f \\ \delta_r \end{bmatrix} \quad (1)$$

where:

$$a_{11} = -\frac{C_f + C_r}{m * v_x} \qquad a_{12} = \frac{-l_f * C_f + l_r * C_r}{m * v_x} - v_x$$

$$a_{21} = \frac{-l_f * C_f + l_r * C_r}{I_\psi * v_x} \qquad a_{22} = -\frac{l_f^2 * C_f + l_r^2 * C_r}{I_\psi * v_x} \quad (2)$$

$$b_{11} = \frac{C_f}{m} \qquad b_{12} = \frac{C_r}{m}$$

$$b_{21} = \frac{l_f * C_f}{I_\psi} \qquad b_{22} = -\frac{l_r * C_f}{I_\psi}$$

with parameters defined as: m - total vehicle mass, I_ψ - total vehicle inertia about vertical axis at CG [$kg * m^2$], l_f (l_r) - distance of front(rear) axle from CG, C_f (C_r) - front (rear) tire cornering stiffness [N/rad], v_x (v_y) – longitudinal (lateral) velocity [m/s], δ_f (δ_r) - front (rear) wheel steer angle [rad], r - yaw rate [rad/s], CG - centre of gravity (see Table I).

Two special maneuvers, the so-called Zero-side-slip Maneuver and Parallel Steering Maneuver [15], take advantage of the special kinematic characteristic of 4 DW/SW vehicles and are commonly used. In the following, we will show how these two maneuvers can be used in our problem.

A. Zero-side-slip Maneuver - In this maneuver, the sideslip angle is set to zero from the starting point to the ending point when the vehicle moves along the path. The orientation of the vehicle $\psi(t)$ is set to match the tangential angle of the desired path $\psi_d(t)$. This maneuver is desirable in vehicle motion since the vehicle body is always tangent to the path (see Fig. 3A).

B. Parallel Steering Maneuver - Parallel Steering is defined as that both two wheels are always steered at the same angle in the same direction. In this maneuver, two steering angles is set as follows $\delta_f(t) = \delta_r(t)$, $t = 0 \rightarrow t_{fin}$. This implies that the vehicle translates without changing its orientation during the motion. Thus we have $\psi(t) = \psi_0$, $t = 0 \rightarrow t_{fin}$ where ψ_0 is the initial heading angle of the vehicle. This maneuver is very practical in vehicle lanechanging and obstacle avoidance (see Fig. 3B). The rotation of the vehicle is reduced as well, thus improves the vehicle stability at high speed.

In this paper is taken into account first case (case A) where $\delta_r = -\delta_f$.

The model described by Equation (1) is independent of the road reference. To describe the vehicle relative to the road, the vision dynamics of the system is modeled using visual information and road geometry. Figure 2 shows a vehicle following a desired path with road curvature K. In this paper, the vehicle is assumed to be equipped with a set of vision system. The vision system is used to estimate the offset from the centerline y_{Lh} and angle between the road tangent and heading of the vehicle ε_{Lh} at some look-ahead distance Lh where measurement is taken.

The equations capturing the evolution of these measurements due to the motion of the vehicle and changes in the road geometry are:

$$\dot{y}_{Lh} = v_x * \varepsilon_{Lh} - v_y - r * Lh$$
$$\dot{\varepsilon}_{Lh} = v_x * K - r \quad (3)$$

The main characteristics that must rule the lateral control and that differentiate it from trajectory tracking (TT), can be summarized as follows:

- Only the global shape of the path is considered to do the following. The desired trajectory evolution (governed by time) must not play any role in the track as it does in TT.
- The existence of the rigid law in TT means pulling or dragging the robot to reach the reference. On the other hand, in lateral control the reference path can not pull (or drag) the robot: the robot must move independently by some condition (of course, meanwhile a control law must ensure convergence to the path). We must impose a motion in the real system to guarantee that it moves or progresses. In the current mobile robot literature most motion exigencies are applied to mobile robots, so it is usual to have $v_{xd} = ct$ (other authors use $|v_{xd}| \neq 0$) or a velocity profile for v_{xd} is supposed to be given between the initial and final position.
- A direct result of what is explained before is that there is no time exigency in the lateral control. This means that we cannot ensure that the robot will reach a reference point in a predictable period of time.

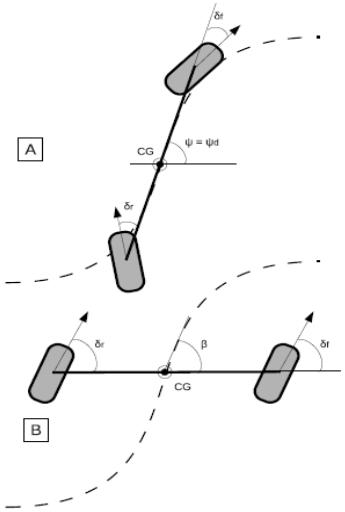


Fig. 3. A. Zero-side-slip maneuver and B. Parallel steering maneuver.

III. SLIDING-MODE CONTROLLER

A Sliding Mode Controller is a Variable Structure Controller (VSC). Basically, a VSC includes several different continuous functions that map plant state to a control surface, and the switching among different functions is determined by plant state that is represented by a switching function.

Gao and Hung [16] proposed a method of reaching mode and reaching law, based on *m*-input *nth*-order systems. In order to assure the attraction of state trajectory onto the switching manifold within the reaching mode, they suggested the control of reaching speed by certain reaching law. The general form of reaching law is

$$\dot{s} = -Q * sgn(s) - P * h(s) \quad (4)$$

where

$Q=diag[q_1,q_2,\ldots,q_m];\ q_i>0$

$sgn(s)=[sgn(s_1),\ sgn(s_2),\ldots,sgn(s_m)]^T$

$P=diag[p_1,p_2,\ldots,p_m];\ p_i>0$

$h(s)=[h_1(s_1),h_2(s_2),\ldots,h_m(s_m)]^T$

$s_i*h_i(s_i)>0;h_i(0)=0$

A practical form of reaching the control law is defined as

$$\dot{s} = -Q * sgn(s) - P * s \quad (5)$$

This reaching law increases the reaching speed when the state is far away from the switching manifold, but reduces the rate when the state is near the manifold. The result is a fast reaching and low chattering reaching mode. In addition, because of the absence of the –Q*sgn(s) term on the righthand side of (4), this reaching law eliminates the chattering.

A new design of sliding surface is proposed such that lateral error, y_{Lh}, and angular error, ε_{Lh}, are internally coupled with each other in a sliding surface leading to convergence of both variables. For that purpose the following sliding surface was proposed:

$$s = \dot{y}_{Lh} + \gamma_y * y_{Lh} + \gamma_0 * sgn(y_{Lh}) * |\varepsilon_{Lh}| \quad (6)$$

here γ_0 and γ_y are positive constant parameters.

If s converges to zero, in steady-state it becomes $\dot{y}_{Lh} = -\gamma_y * y_{Lh} - \gamma_0 * sgn(y_{Lh}) * \varepsilon_{Lh}$. For $y_{Lh} < 0 \Longrightarrow \dot{y}_{Lh} > 0$. For $y_{Lh} > 0 \Longrightarrow \dot{y}_{Lh} < 0$. Finally, it can be known from s that the convergence of y_{Lh} and $\dot{y}_{Lh}$ leads to the convergence of ε_{Lh} to zero.

From the time derivative of (6) and using the reaching law defined in (5) yields:

$$\dot{s} = \ddot{y}_{Lh} + \gamma_y * \dot{y}_{Lh} + \gamma_0 * sgn(y_{Lh}) * sgn(\varepsilon_{Lh}) * \dot{\varepsilon}_{Lh} = \\ = -Q * sgn(s) - P * s \quad (7)$$

From (1), (3), (7) and after some mathematical manipulation, we get the output command of the sliding-mode controller:

$$\delta_{fc} = \frac{Q*sgn(s)+P*s+\gamma_y*\dot{y}_{Lh}+E}{b_{11}+b_{12}+(b_{21}+b_{22})*Lh} \quad (8)$$

where

$$E = \dot{v}_x * \varepsilon_{Lh} + v_x * \dot{\varepsilon}_{Lh} - v_y * (a_{11} + a_{21} * Lh) - \\ -r * (a_{12} + a_{22} * Lh) + \gamma_0 * sgn(y_{Lh} * \varepsilon_{Lh}) * \dot{\varepsilon}_{Lh} \quad (9)$$

IV. SIMULATION RESULTS

In this section, some simulation results are presented to validate the proposed control law. To show the effectiveness of the proposed sliding mode control law numerically, experiments were carried out on the lateral control problem of a 4 DW/SW vehicle. The DW/SW vehicle is assumed to have the same structure as in Fig. 2.

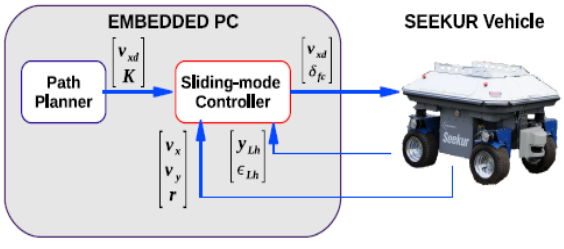


Fig. 4. Sliding-mode control architecture.

The 4 DW/SW vehicle has a two-level control architecture (see Fig. 4). High-level control algorithms (including desired motion generation) are written in C++ and run with a sampling time of Ts = 100 ms on a embedded PC, which

also provides a user interface with real-time visualization and a simulation environment.

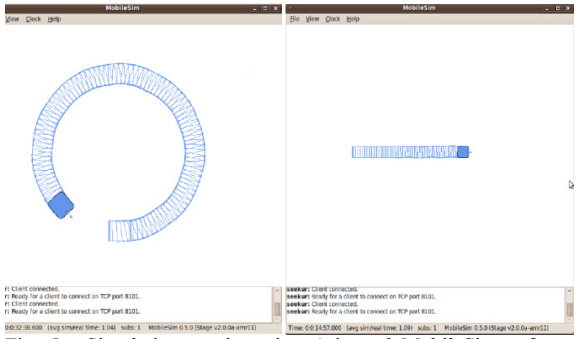


Fig. 5. Simulation results using Aria and MobileSim software (case I and II).

All the simulations were made using the MobileSim. MobileSim is software for simulating MobileRobots' platforms and their environments, for debugging and experimentation with ARIA. The ARIA software can be used to control the mobile robots like Pionner, PatrolBot, PeopleBot, Seekur etc. ARIA (Advanced Robot Interface for Applications) it is an object-oriented Applications Programming Interface (API), written in C++ and intended for the creation of intelligent high-level client-side software.

Two simulation experiments were carried out to evaluate the performance of the sliding mode controller presented in Section III. The first simulation refers to the case of circular path ($v_{xd} = 0.5m/s$ and $K = 0.2$). The look-ahead distance is $Lh = 1m$. In the second simulation the 4 DW/SW robot execute a linear path ($v_{xd} = 0.5m/s$ and $K = 0$).
Fig. 6 shows the trajectory of the 4 DW/SW robot when the robot executes a circular path.

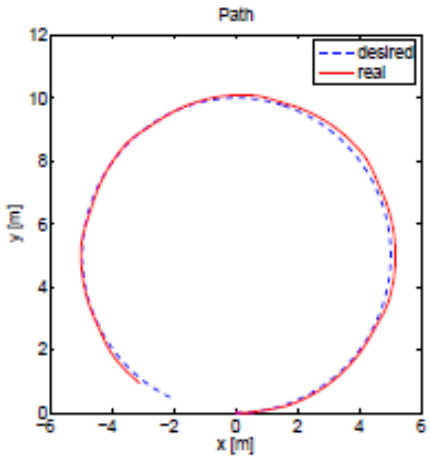


Fig. 6. Simulation I - 4WDS robot made a circular path.

In Fig. 7 the sliding surface s and steering command (δ_f) are shown. Finally, Figs. 8 show the time histories of the offset from the centerline at the look-ahead (y_{Lh}) and the angle between the tangent to the road and the vehicle orientation (ε_{Lh}).

In Fig. 9 the sliding surface s and steering command (δf) are shown when 4 DW/SW vehicle execute a linear path. Figs. 10 show the time histories of the offset from the centerline at the look-ahead (y_{Lh}) and the angle between the tangent to the road and the vehicle orientation (ε_{Lh}).

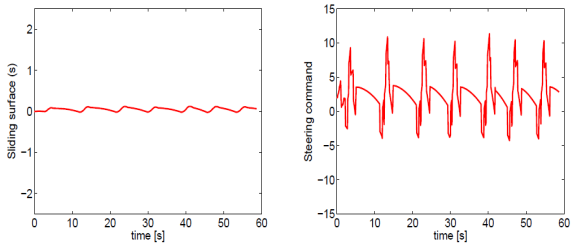


Fig. 7. Simulation I - Sliding surface (s) and the steering command (δ_f).

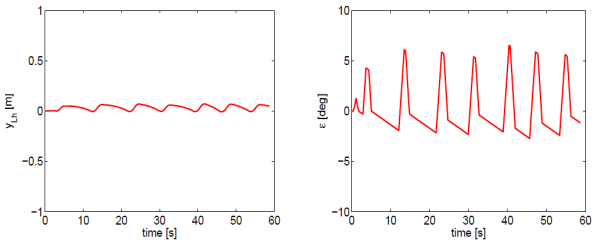


Fig. 8. Simulation I - Variation of the offset from the centerline at the look-ahead (y_{Lh}) and the angle between the tangent to the road and the vehicle orientation (ε_{Lh})

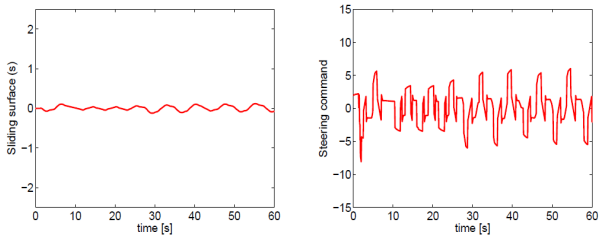


Fig. 9. Simulation II - Sliding surface (s) and the steering command (δ_f).

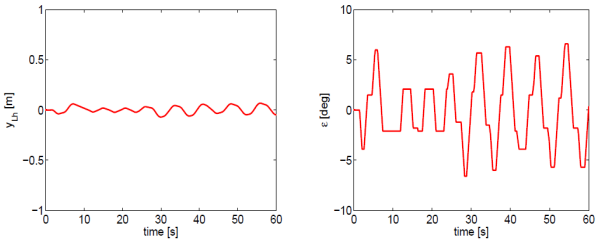


Fig. 10. Simulation II - Variation of the offset from the centerline at the look-ahead (y_{Lh}) and the angle between the tangent to the road and the vehicle orientation (ε_{Lh})

V. REAL TIME RESULTS

In this section, test results are presented to validate the proposed control law. To show the effectiveness of the proposed sliding mode control law, experiments were carried out on the lateral control problem of a Seekur vehicle.

The high-level control algorithms (including desired motion generation) are written in C++ and run with a sampling time of Ts = 100 ms on a PC. A wireless network was set in order to allow data transfer between the robot and the PC. The wireless network used a wireless access point and an universal device server that was connected to the access point and the serial connector of the robot.

An experiment was carried out to evaluate the performance of the sliding mode controller presented in Section III. The test refers to the case of linear path (v_{xd} = 0.3m/s and K = 0).

Fig. 11 shows the trajectory of the 4 DW/SW robot when the robot executes a linear path. In Fig. 12 the sliding surface s and steering command (δ_f) are shown. Finally, Figs. 13 show the time histories of the offset from the centerline at the look-ahead (y_{Lh}) and the angle between the tangent to the road and the vehicle orientation (ε_{Lh}).

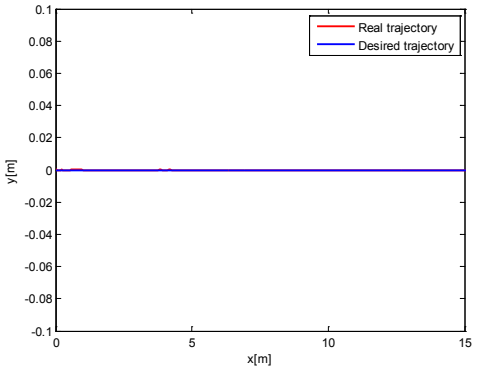


Fig. 11. Real time trajectory of a Seekur robot.

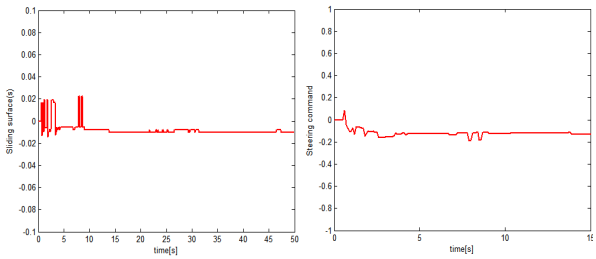


Fig. 12. Simulation I - Sliding surface (s) and the steering command (δ_f).

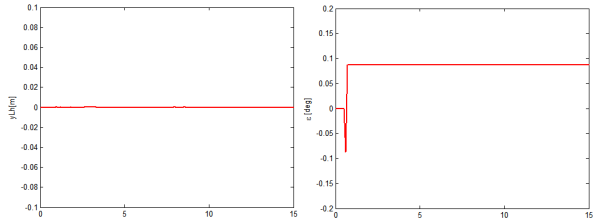


Fig. 13. Simulation II - Variation of the offset from the centerline at the look-ahead (y_{Lh}) and the angle between the tangent to the road and the vehicle orientation (ε_{Lh})

VI. Conclusions

An SM controller based on the look-ahead reference system is employed in the feedback loop of the control system. Simulation example is used to evaluate the sliding-mode algorithm and to show the application of the algorithm in practice. The controller is simply structured and easy to implement. From the simulation and real time results, it is concluded that the proposed strategy achieves the effectiveness of desired performance. In general, the primary objective of this research has been achieved where the proposed control system is able to track the centre of a lane automatically with small error under various conditions.

Future research lines include the experimental validation of our control scheme and the extension of our results to Four Driving-Steering Wheels steering mobile robots on more complex paths.

References

[1] T. Hiraoka, O. Nishihara and H. Kumamoto, *Automatic path-tracking controller of a four-wheel steering vehicle, Vehicle System Dynamics, Taylor & Francis, 47(10), 2009, pp. 1205-1227.*

[2] J-H. She, X. Xin and Y. Ohyama. *Estimation of equivalent input disturbance improves vehicular steering control, IEEE Transactions on Vehicular Technology, 56(6), 2007, pp. 3722-3731.*

[3] A.S. Abdullah, L.K. Hai, N.A.A. Osman and M.Z. Zainon, *Vision based automatic steering control using a PID controller, Jurnal Teknologi, 44(A), 2006, pp. 97114.*

[4] *Seekur - Autonomous All-Weather Robot. Mobile Robots Inc. http://www.mobilerobots.com/ CommSeekur.html.*

[5] T. Huntsberger, H. Aghazarian, Y. Cheng, E. Baumgartner, E. Tunstel, C. Leger, A. Trebi-Ollennu and P. Schenker, *Rover Autonomy for Long Range Navigation and Science Data Acquisition on Planetary Surfaces, Proceedings of the IEEE ICRA '02 Conference, 2002, pp. 3161-3168.*

[6] A. Lacaze, K. Murphy and M. DelGiorno, *Autonomous Mobility for the DEMO III Experimental Unmanned Vehicles, Proceedings of the AUVSI '02 Conference, Orlando, 2002.*

[7] M. Montemerlo, S. Thrun, H. Dahlkamp, D. Stavens, S. Strohband, *Winning the DARPA Grand Challenge with an AI Robot, Proceedings of the AAAI National Conference on Artificial Intelligence, Boston, 2006, pp. 982-987.*

[8] C. Wellington and A. Stentz, *Online Adaptive Rough-Terrain Navigation in Vegetation. Proceedings of the ICRA '04 Conference. Vol. 1, 2004, pp. 96-101.*

[9] V.I. Utkin, *Sliding modes in optimization and control, Springer-Verlag, New York, 1992.*

[10] V.I. Utkin, J. Guldner, and J. Shi, *Sliding mode control in electromechanical systems, Taylor & Francis, London, 1999.*

[11] J.J.E. Slotine and W. Li, *Applied nonlinear control. Prentice-Hall, London, 1991.*

[12] R. Solea, A. Filipescu and G. Stamatescu, *Sliding-mode real-time mobile platform control in the presence of uncertainties, Proceedings of the 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference, Shanghai, 2009, pp. 7747-7752.*

[13] R. Solea, *Sliding mode control applied in trajectory-tracking of WMRs and autonomous vehicles, PhD Thesis, University of Coimbra, Portugal, 2009.*

[14] R. Solea, A. Filipescu, U. Nunes, *Sliding-mode control for trajectorytracking of a wheeled mobile robot in presence of uncertainties, Proceedings of the 7th Asian Control Conference, Hong Kong, 2009, pp. 1701-1706.*

[15] W. Danwei and Qi Feng, *Trajectory planning for a four-wheel steering vehicle, Proceedings of the IEEE International Conference on Robotics and Automation, Seoul, 2001, pp. 3320-3325.*

[16] W. Gao and J. C. Hung, *Variable structure control of nonlinear systems: A new approach, IEEE Transactions on Industrial Electronics, 40(1), 1993, pp. 45 - 55.*

Dynamic Simulator of a Wet Plate Clutch System for Automatic Transmission

Emanuel E. Feru, Daniel I. Patrascu, Corneliu Lazar, *Member, IEEE*

***Abstract*—Automatic transmission became a modern hydraulic system whose essential elements are the clutch and the valve that controls it. In this paper, an analytic model of a wet plate clutch actuated by a pressure reducing valve is developed. Next this model is converted into a Simulink model, with which simulation can be performed in an easy way. To validate the model a test bench, developed by Automotive Continental Romania, was used.**

I. Introduction

THE automatic transmission system was the subject for many other publications [1], [2], [3], [4], where the main idea was to improve the control algorithms or mechanical design to reduce energy losses. Attempts were made to keep the traditional clutch and gear stick untouched but removing the clutch pedal [5]. This was replaced by an actuator controlling the clutch position and therefore the torque transmission. After some precautions were taken, a two-level cascaded feedback design was used to control the vehicle speed with this clutch system. Another type of approach was made by [6] where the automatic transmission is modeled in detail for each component. There, a dynamic model of a power transmission system for gear ratio changed was proposed. Later, [7] developed a mathematical model of a one-way clutch in belt-pulley systems, where a wrap-spring type of clutch is modeled as a nonlinear spring with discontinuous stiffness. In [8] a model is presented that includes the non-linear nature of the diaphragm spring, kinematics of the pedal motion during clutch release, and the dynamics of the driveline and the overall vehicle.

Researches made until now [9], [10], tell that torque can be varied by modifying the clutch position and although they are mostly based on using dry clutches, they give a good input for the modeling of some automatic transmission elements.

Some major advantages of the wet plate clutches can be mentioned: they are built with multiple clutch disks that give a better grip which can be controlled by the pressure inside the clutch. The shifting speed can be considerably reduced by using independent clutch assemblies known as the dual-clutch system. A wet clutch is immersed in a cooling lubricating fluid, which keeps the surfaces clean and gives smoother performances and longer life. Disadvantages are related to the additional energy losses by always actuating a hydraulic pump and the use of high fluid pressure which is unsafe for the environment.

In this paper, a dynamic model of a wet plate clutch based on mathematical equations obtained with the help of physical laws and experimental measurements is developed. A hydraulic valve actuator is considered in order to analyze the general effect of the automatic transmission system. For validating the model and to procure all the experimental data needed, a test bench developed by Automotive Continental Romania was used. What brings new this model described by the mathematical equations is that a new simulator was developed for a non standardized type of pressure reducing valve and wet plate clutch. This simulator can be then used to make diverse simulations in order to understand the behavior of the system, to design complex strategies of control or to test and improve a controller performances already implemented.

The paper is organized as follows. In Section I the modeling of a pressure reducing valve and a wet plate clutch is discussed. In Section II both elements are evaluated and results are discussed by means of simulation in Section III. Finally, conclusions are drawn in Section IV.

II. Problem Formulation

In this section the modeling of a wet plate clutch actuated by a pressure reducing valve is presented. Hydraulic control valves are devices that use mechanical motion to control a source of fluid power and are used as actuators in many control applications for automotive systems. Basically there are three types of control valves: directional control valves, pressure control valves and flow control valves. The type of valve and clutch analyzed in this paper are not standardized, they are designed for a certain application where some performances need to be achieved. In what follows it is presented the dynamics of the valve, then the dynamics of the clutch.

A. Pressure reducing valve model

Schematics of the three way pressure reducing valve used as clutch actuator is presented in Fig. 1.

Manuscript received September 1, 2010. This work was supported in part by the National Centre for Programs management from Romania under the research grant 12100/2008 SICONA.

E. E. Feru is with the Technical University "Gheorghe Asachi" of Iasi, Romania (phone: +40-747-665253; e-mail: emanuel.feru@continental-corporation.com).

D. I. Patrascu is with the Technical University "Gheorghe Asachi" of Iasi, Romania (phone: +40-743-817764; e-mail: daniel.patrascu@continental-corporation.com).

C. Lazar is with the Technical University "Gheorghe Asachi" of Iasi, Romania (phone: +40-232-278680; e-mail: clazar@ac.tuiasi.ro).

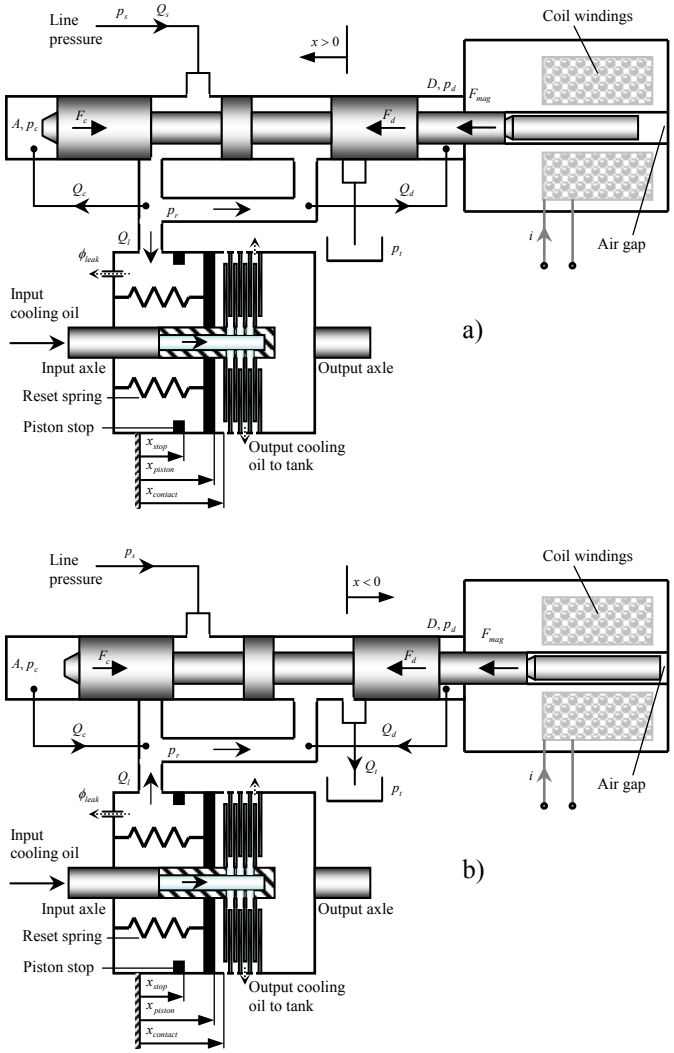


Fig. 1 Schematic draw of the valve connected to the clutch: a) Charging phase; b) Discharging phase

The input for this hydraulic system is represented by the line pressure p_s and current i passing through a solenoid which generates a magnetic force F_{mag}, while the output of the system is represented by the pressure p_r. The physical construction of the valve implies the presence of three pressure variables corresponding to each chamber: p_c, p_r and p_d for the left chamber, the middle chamber and the right chamber respectively.

The hydraulic valve has a self controller by using a mechanical feedback through left and right tubes. Because differences in pressure appear between the middle chamber and the left/right chambers, flows Q_c and Q_d are generated. The feedback force is composed from the force F_c applied on the left sensed pressure chamber, and the force F_d applied on the right sensed pressure chamber. These two forces composed with the magnetic force F_{mag} are used to actuate the spool valve which controls the pressure p_r inside the middle chamber. In the charging phase (Fig. 1, a), the magnetic force given by solenoid current, is grater than the feedback pressure force moving the plunger to the left, connecting the source with the clutch. In the discharging phase (Fig. 1, b), the magnetic force is less than the feedback force implying that plunger will move to the right, connecting the clutch to the tank.

The mathematical connection between electric current trough solenoid and magnetic force generated by the magnetic flux is given in [11].

Feedback pressure force and magnetic force are the most important forces that give the forces balance applied on the valve plunger. Based on Pascal equations which transform the hydraulic pressure into mechanic force we can simulate the feedback pressure by next equation:

$$F_{feedback} = Ap_c - Dp_d \,, \tag{1}$$

where, p_c , p_d are the pressures of the left and right chambers, A is the left plunger area of pressure contact, D is the right plunger area of pressure contact.

Using the magnetic force and feedback force created by the pressure inside the left and right chambers the movement of the plunger, which governs the output pressure, can be approximated with a mass spring damper system given by:

$$F_{mag} - F_{feedback} = M_v\ddot{x} + c\dot{x} + K_e x \,, \tag{2}$$

where, $F_{feedback}$ represents the feedback force, M_v is the spool mass, x represents the plunger displacement, $\dot{x}$ is the plunger speed, $\ddot{x}$ is the plunger acceleration, K_e represents the flow force spring rate and c is the damper coefficient.

To restrict the motion of the plunger between the left and the right bounds a double-side mechanical translational hard stop was implemented:

$$F_{restrict} = \begin{cases} K_p x + D_p \dot{x}, x \geq x_{\max} \\ 0, x_{\min} < x < x_{\max} \\ K_n x + D_n \dot{x}, x \leq x_{\min} \end{cases} , \tag{3}$$

where, $F_{restrict}$ is the dynamic balance force between the plunger and the mechanical limits, $x_{\min}$, $x_{\max}$ are the gap on the right and left side, K_p , K_n represent the contact stiffness at left and right restriction, D_p , D_n are damping coefficient at left and right restriction.

The linearity of pressure drop and flow, a characteristic of all laminar flows, is desirable in many circuits. Capillary tubes are often used to stabilize pressure control valves, as upstream restrictors in hydrostatic bearing. For this reason flow through left and right tubes were considered, for simplicity, to be laminar.

Flows Q_c and Q_d through a tube for both charging and discharging phase are given by [12]:

$$Q_c = \frac{\pi r_c^4}{8\varepsilon L_c}(p_r - p_c), \quad (4)$$

$$Q_d = \frac{\pi r_d^4}{8\varepsilon L_d}(p_r - p_d), \quad (5)$$

where, r_c, r_d are the left tube radius and right tube radius, L_c, L_d represents the passage length of the tubes and ε represents the kinematic viscosity of fluid.

To describe the dynamics given by the left and right chamber the flow continuity equation for each chamber is given by:

$$Q_c = \frac{V_c - Ax}{\beta_e}\frac{dp_c}{dt} - A\frac{dx}{dt}, \quad (6)$$

$$Q_d = \frac{V_d + Dx}{\beta_e}\frac{dp_d}{dt} + D\frac{dx}{dt}, \quad (7)$$

where, V_c, V_d are the sensing chamber volumes for the initial spool displacement and β_e is the effective bulk modulus.

In the (6) and (7) equations the sign for the left and right chamber volume were chosen based on the fact that total volume variation of these chambers depend accordingly to spool displacement. Therefore, a movement of the plunger to the left will decrease the left chamber volume and increase the right chamber volume.

From the (6) and (7) equation the derivative of the pressure sensed in the left and right chamber respectively can be written as follows:

$$\frac{dp_c}{dt} = \frac{\beta_e}{V_c - Ax}\left(Q_c + A\frac{dx}{dt}\right), \quad (8)$$

$$\frac{dp_d}{dt} = \frac{\beta_e}{V_d + Dx}\left(Q_d - D\frac{dx}{dt}\right), \quad (9)$$

An important remark can be made regarding the (6) and (7) equations, that direction of flow Q_C and Q_D are given accordingly to the plunger movement.

The flow behavior for the orifices that connects the clutch to the source and the clutch to the tank was modeled considering variable orifices created by a cylindrical sharp-edged spool. The flow rate through the orifice is proportional to the orifice opening and to the pressure differential across the orifice. The model accounts for the turbulent flow regimes because fluid passes through these orifices with a high speed leading to high Reynolds numbers.

For the charging phase the flow rate from the source to the clutch is determined according to the next equation:

$$Q_s = C_d A_s(x)\sqrt{\frac{2}{\rho}|p_s - p_r|}, \quad (10)$$

$$A_s(x) = bx + A_{leak}, \quad (11)$$

where, C_d is the flow discharge coefficient, $A_s(x)$ is the instantaneous orifice area, *b* represents the orifice slot width, A_{leak} represents the orifice leakage area and ρ is the fluid density. For simplicity, A_{leak} was considered to be 0.

The flow continuity equation at the chamber of the pressure being controlled for the charging phase is represented by:

$$Q_s - Q_c - Q_d - Q_l = \frac{V_s}{\beta_e}\frac{dp_r}{dt}. \quad (12)$$

For the discharging phase the flow rate from the clutch to the tank is obtained in the same manner and given by:

$$Q_t = C_d A_s(x)\sqrt{\frac{2}{\rho}|p_r - p_t|}, \quad (13)$$

where, p_t is the tank pressure.

For this functioning phase of the valve the flow Q_l, in this case called the discharging flow, will be in opposite direction and included in the following equation:

$$Q_l + Q_c + Q_d - Q_t = \frac{V_s}{\beta_e}\frac{dp_r}{dt}. \quad (14)$$

From (12) and (14) equations it can be noticed that flows Q_c and Q_d will be considered as a perturbation for the flow Q_l which depends on the load connected to the output of the valve. In our case the load is a wet plate clutch and flow Q_l will be obtained accordingly to the dynamics given by this hydraulic element.

These equations define the valve and can be used to construct the block diagram in Fig. 2.

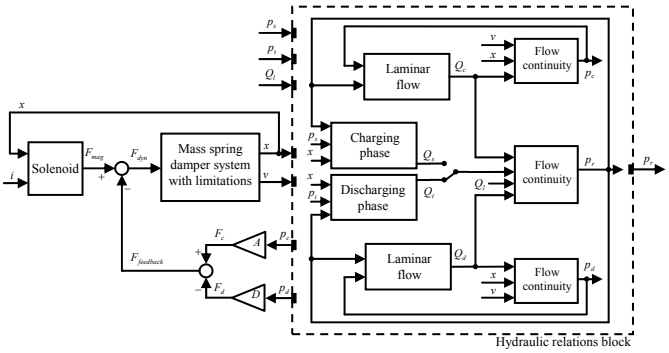


Fig. 2 Nonlinear valve system representation

The solenoid block is considered to be a look-up table, obtained with the help of the experimental measurements, having as input the current and displacement of the plunger and as output the magnetic force. The mass spring damper system is implemented accordingly to the equation (2) introducing the limitations (3) given by the left bound and right bound of the pressure reducing valve. In the charging phase and discharging phase blocks equations (10) and (13) were materialized to obtain the flow which comes from

source and the flow going to the tank, respectively. The flows Q_c and Q_d were computed using relations (4) and (5) in the blocks called laminar flow. Finally, to obtain the pressures p_c, p_d and the output pressure p_r equations (8), (9) and (12), (14) were implemented in the flow continuity blocks. The pressures p_c, p_d and p_r were then used as inputs for the laminar flow blocks and only p_r as input for the charging/discharging phase blocks.

B. Wet plate clutch model

A wet plate clutch is schematically given in Fig. 1. The clutch itself is a chamber with a piston. Behind the piston the clutch plates are fixed on the input axle, respectively on the output axle. The friction surfaces of these plates are covered with an organic material, lubricated with cooling oil, which is brought in through the input axle.

When the clutch closes, considering a step input, four stages can be distinguished for the clutch pressure (Fig. 3):

1) filling of the clutch, without significant pressure increase;
2) pressure increase until the pressure is large enough to exceed the force applied on the piston by the reset spring;
3) movement of the piston towards the clutch plates;
4) pressure rise to the set point value p_{set} after the clutch plates resume contact.

Initially the clutch is considered to be open having no fluid inside the chamber. The piston is pulled back to the piston stops (Fig. 1) by the reset springs with total stiffness k_{reset}. Their elongation in this case is denoted by x_{stop}. The piston itself is assumed to be mass less with a cross sectional area A_{piston} on which the clutch pressure is effective.

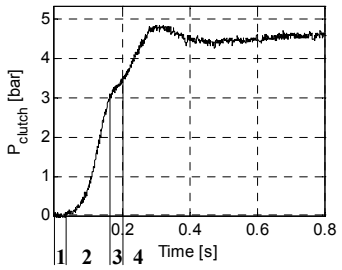


Fig. 3 Step response of the clutch pressure

The piston position x_{stop} can be derived from the equilibrium equation:

$$p_{clutch} A_{piston} + F_{stops} = k_{reset} x_{piston} + F_{friction}, \tag{15}$$

where F_{stops} is the reaction force of the stops on the piston.

As soon as the force on the piston exerted by the clutch pressure becomes larger than the sum of the friction force and the force applied on the piston by the reset springs, the piston starts moving towards the clutch plates. This pressure, denoted by $p_{prestress}$, is given by:

$$p_{prestress} = \frac{k_{reset} x_{stop} + F_{friction}}{A_{piston}}. \tag{16}$$

The plates are pressed together with a compression force $F_{compress}$, expressed by:

$$F_{compress} = (p_{clutch} - p_{contact}) A_{piston}. \tag{17}$$

The flow continuity equation written for our application is:

$$Q(t) - Q_{leak}(t) = \frac{dV(t)}{dt} + \frac{V(t)}{k_s} \frac{dp_{clutch}(t)}{dt}, \tag{18}$$

where $V(t) = V_0 + A_{piston} x_{piston}(t)$ is the volume variation.

After making some calculations in equation (18) we obtain a more convenient form, for the wet plate clutch model:

$$\dot{p}_{clutch}(t) = \frac{k_s}{V_0 + A_{piston} x_{piston}(t)} \left[- A_{piston} \dot{x}_{piston}(t) + Q(t) - Q_{leak}(t) \right] \tag{19}$$

Furthermore, equation (19) can be specialized for each of the four stages mentioned before.

Stage 1, filling of the clutch:

$$\begin{aligned} & V_{oil,0} + \int_{t=t_0}^{t} Q(\tau) d\tau < V_0 \\ & p_{clutch}(t) = \dot{p}_{clutch}(t) = 0 \end{aligned}. \tag{20}$$

Stage 2, pressure increase without piston displacement:

$$\begin{aligned} & 0 \le p_{clutch}(t) < p_{prestress}. \\ & \dot{p}_{clutch}(t) = \frac{k_s}{V_0 + A_{piston} x_{stop}} \left[Q(t) - Q_{leak}(t) \right]. \end{aligned} \tag{21}$$

Stage 3, movement of the piston:

$$p_{prestress} \le p_{clutch}(t) < p_{contact}. \tag{22}$$

With the relations for the piston displacement $x_{piston}(t)$ and speed $\dot{x}_{piston}(t)$:

$$\begin{aligned} & x_{piston}(t) = \frac{A_{piston} p_{clutch}(t) - F_{friction}}{k_{reset}} \approx \frac{A_{piston} p_{clutch}(t)}{k_{reset}}, \\ & \dot{x}_{piston}(t) = \frac{A_{piston}}{k_{reset}} \dot{p}_{clutch}(t), \end{aligned} \tag{23}$$

equation (19) becomes:

$$\dot{p}_{clutch}(t) = \left(\frac{V_0 + A_{piston} x_{piston}(t)}{k_s} + \frac{A_{piston}^2}{k_{reset}} \right)^{-1} \left[Q(t) - Q_{leak}(t) \right]. \tag{24}$$

Stage 4, compression of the clutch plates:

$$p_{clutch}(t) \ge p_{contact}, \tag{25}$$

$$\dot{p}_{clutch}(t) = \frac{k_s}{V_0 + A_{piston} x_{contact}} \left[Q(t) - Q_{leak}(t) \right]. \qquad (26)$$

Equations (20), (21), (22), (24), (25) and (26) represent the wet plate clutch model.

III. Model Validation

The model of the valve actuator and the wet plate clutch were validated and tested using a test bench from Automotive Continental Romania (Fig. 4). Simulation results were recorded and compared with the experimental data provided by this test bench.

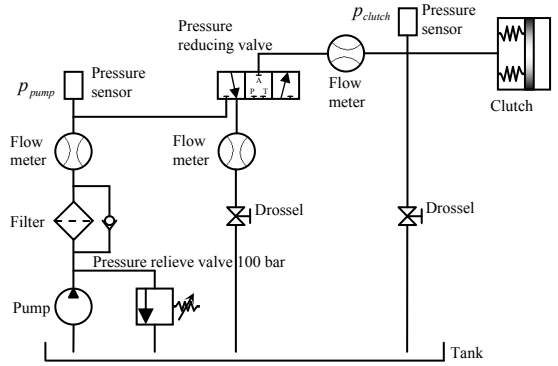


Fig. 4 Simplified test bench representation

The primary function of the test bench is to evaluate the behavior of various hydraulic elements, designed models and control mechanisms. Furthermore it is equipped with several measurement and control devices.

A. Simulator of the valve-clutch system

The simulator of the valve-clutch system, based on the analytic model described in Section II, was constructed in Matlab/Simulink and is implemented regarding the connections between the hydraulic valve and wet plate clutch to simulate the system dynamics.

The simulator (Fig. 5) is composed by the valve and clutch model. In the left of this figure are the inputs of the system, while in the right the output variables are recorded in Matlab memory during simulation. The input of the simulator is represented by the line pressure and the current passing through the coil of the valve which generates a magnetic force F_{mag} applied to one end of the plunger. The magnetic force depends on geometry of the valve and this was hard to consider using mathematical support. For this reason, the magnetic force was measured using a force sensor and the results were used in form of a two dimensional look-up table. A low pass filter was used in order to reduce the noise of the current signal.

The resultant of forces represents the input for the mass spring damper block to obtain the plunger displacement, velocity and acceleration. The output of this block is used in equations (10), (13) to dictate the flow through the valve and in equations (8), (9) for computing the pressure values at the spool end areas.

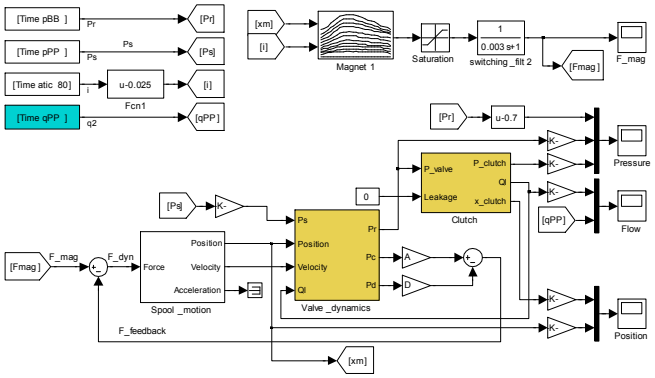


Fig. 5 Simulator representation of the valve-clutch system

The model of the valve contains a switch (Fig. 6) whose function is to commute between the charging phase and discharging phase having as a threshold factor the 0 mm of the plunger displacement which means that all the orifices are closed. Next, the flow Q_c and Q_d were implemented based on the (4), (5) equations and used as a perturbation for the flow Q_l. In the right side of the Fig. 6 the continuity equations (8), (9), and (12), (14) were implemented. The pressure sensed in the left chamber and right chamber creates a feedback force which contributes to the equilibrium of forces.

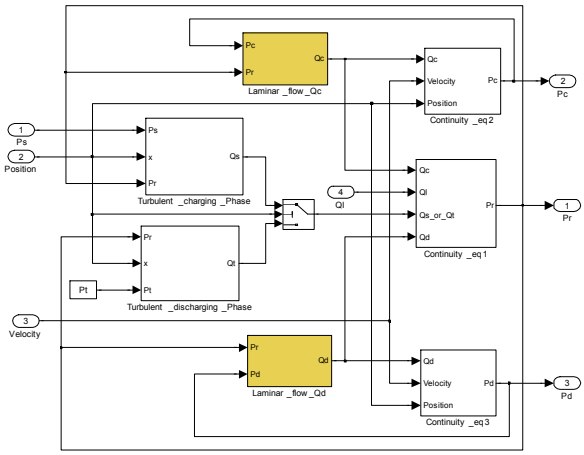


Fig. 6 Simulink block diagram of the valve

The four stages in the pressure response when closing the clutch, as described in the model, are displayed in Fig. 7 as subsystems. Pressure values, like $p_{contact}$ and $p_{prestress}$ are used as switching parameters between the stages. Next the time derivative of the pressure is fed into an integrator to gain the pressure. In this component also the initial pressure can be enclosed. A limitation block is developed to represent the stops. Furthermore, the relation between the pressure and overall compressibility is converted into a lookup table.

It can be noticed that piston movement takes place only in stage three from x_{stop} to $x_{contact}$. Thereby, a pressure controller can be developed based on piston displacement.

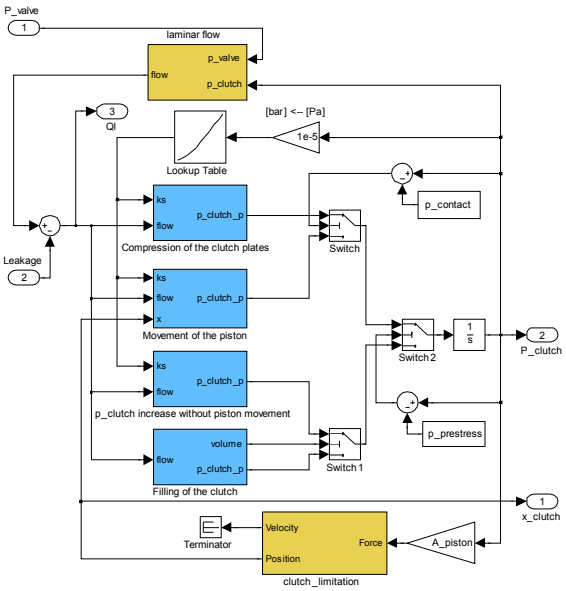


Fig. 7 Simulink block diagram of the wet plate clutch

B. *Simulation results*

In order to validate the model for the solenoid valve actuator and wet plate clutch, a current was applied in coil windings as an input for the simulator described in Section III. In Fig. 8 the input signal is illustrated, represented by a pulse current from 0 to 630 mA used to obtain the magnetic force through the look-up table. The line pressure is set to 10 bars and the obtained output pressure is about 5 bars.

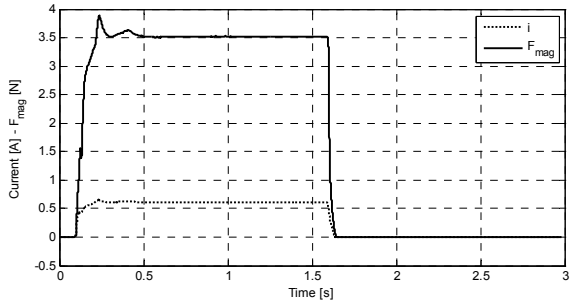


Fig. 8 Current and magnetic force signals

As it can be seen in the first set of signals (Fig. 9, a) the clutch pressure and measured pressure are considerably close, ensuring a small modeling error. Regarding the flow signals (Fig. 9, b) it can be noticed some negative aspects about the experimental response: a persistent noise of reading is present and flow sensor can not read reversed fluid flows. The last set of signals (Fig. 9, c) shows that piston position of the clutch synchronizes smoothly with the clutch pressure in stage three.

Simulations are performed with the valve-clutch system simulator for verifying that model response lie within an acceptable range around the experimental data.

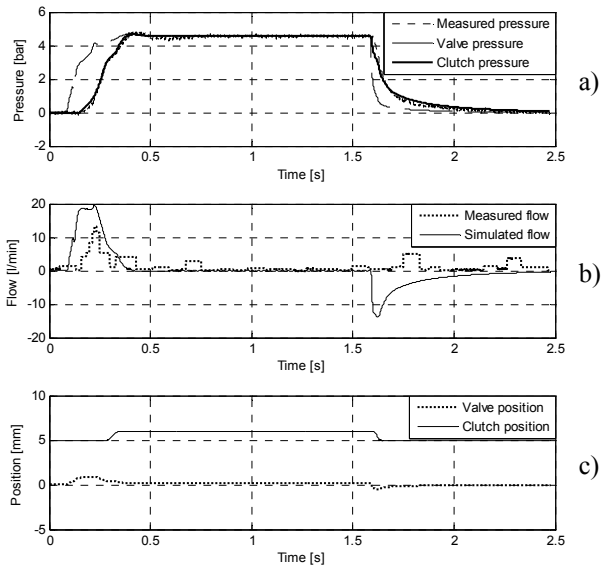


Fig. 9 Simulation responses compared with the experimental measurements

IV. CONCLUSIONS

In this paper a simulator of the valve-clutch system based on mathematical equations, static and transient experiments was developed. The simulator characterizes the dynamics given by the valve, clutch as well as the whole system.

REFERENCES

[1] Liu S. and B. Yao (2008). Coordinative control of energy saving programmable valves. *IEEE Transactions on Control Systems Technology*, **16**, pp. 34-45.

[2] Di Gennaro S., B. Castillo-Toledo and M.D. Di Benedetto (2007). Non-linear control of electromagnetic valves for camless engines. *International Journal of Control*, 80:11, 1796-1813.

[3] Wang Y., T. Megli and M. Haghgooie (2002). Modeling and control of electromechanical valve actuator. *Society of Automotive Engineers*.

[4] Nemeth H. (2004). Nonlinear modeling and control for a mechatronic protection valve. *Ph.D Dissertation*, Budapest, Hungary

[5] J. Lévine and B. Rémond, (2000). Flatness based control of an automatic clutch, *Proc. MTNS-2000*, Perpignan.

[6] Sustersic V., Jovicic N., Babic M., Gordic D. (2003). Principles of automatic transmission modeling and simulation, *Proc. of International Coference on Power transmission '03*, Kragujevac.

[7] Farong Z. (2006). Nonlinear dynamics of one-way clutches and dry friction tensioners in belt-pulley systems, Dissertation Thesis, The Ohio State University.

[8] K. Tripathi and M. D. Agrawal, (2008). Dynamic Modelling of Engagement of Automotive Clutch with Diaphragm Spring, *IE(I) Journal–MC*, volume 88, pp. 10-17.

[9] Chapple, P.J. (2003). *Principles of hydraulic system design*, Coxmoor Publishing Design, Great Britain.

[10] E. Spijker (1994). *Steering and control of a CVT based hybrid transmission for a passenger car*, Wilbro Helmond.

[11] Gibilisco Stan, T.E. (2002). *Teach Yourself Electricity And Electronics*, pp. 134-152, McGraw Hill.

[12] Merritt, H.E (1967). *Hydraulic control systems*, John Wiley & Sons, United States of America.

RoboSmith: Architecture for a Flexible Mini Robot for Multiagent Robotic System

Dan Floroian, Florin Moldoveanu

Abstract- **In this paper is presented an architecture for a flexible mini robot for a multiagent robotic system. In a multiagent system the value of an individual agent is negligible since the goal of the system is essential. Thus, the agents (robots) need to be small, low cost and cooperative. RoboSmith are designed based on these conditions. The proposed architecture divide a robot into functional modules such as locomotion, control, sensors, communication, and actuation. Any mobile robot can be constructed by combining these functional modules for a specific application. An embedded software with dynamic task uploading and multi-tasking abilities is developed in order to create better interface between robots and the command center and among the robots. The dynamic task uploading allows the robots change their behaviors in runtime. The flexibility of the robots is given by facts that the robots can work in multiagent system, as master-slave, or hybrid mode, can be equipped with different modules and possibly be used in other applications such as mobile sensor networks remote sensing, and plant monitoring.**

I Introduction

The main goal of this work is to develop a flexible mini robot to be used in implementation of a multiagent robotic system, and to present the software architecture. The nature of the multiagent systems (even if applied in robotics) brings some limitations and conditions to the design of a reliable platform [1, 5, 11, 13, 14]. Size, cost, and cooperative abilities via specific tools are some of the limitations and conditions. Size and cost limitations are closely related since smaller size means less material thus less cost. With the advances in microelectronics fabrication technologies, the size and cost of the ICs used in the systems went down significantly, which allows the designers to meet the cost limitation. Even though electronic components have gotten smaller significantly, mechanical parts and battery sizes are still reasonably large. Size limitation on a mini robot is imposed by mostly mechanical parts and batteries [3, 6, 9]. The fundamental problem with reducing the size of the mechanical devices is that they either become inefficient or not fully functional when their size is shrunken. For example, when the size of a DC motor or a gear gets smaller, their power and their durability reduce significantly. These limitations have determined our design.

There have been many definitions of a robot in the literature since the beginning of the robotics field in the 1940s. After having been introduced in factories, robots became mobile and smaller with advances made in mechanical and electrical engineering fields. After the mobility of the robots was developed, the artificial intelligence field made its contribution to robotics by making them autonomous and smarter [2, 4, 8, 12].

Multiagent systems represents a relatively new area in computer science and a very new area in robotics, which started to be developed in 1980s but that only in the mid 1990s gained widespread interest [5, 7, 14, 16]. Multiagent systems are compositions of computing elements that possess autonomous action, and which are able to interact among themselves, not only for exchanging messages but also for a more elaborated kind of communication that resembles social activity (cooperation, coordination, negotiation, etc.).

The autonomy of a robot required good communications and sensing skills. These skills can be achieved using multiagent technology, which use agentified components. Also this problem can be approached by using conventional elements for the robot but agentifying the whole robot [1, 12, 16].

Robots need sensors mainly for the reasons. First, sensing is the purpose of the mission. Second, sensing is necessary for survival in the robot's environment: to determine an obstacle on the planned path to the target. Finally, sensing is needed to enable the robots to sense their own configurations and their relationships with the environment. Along with sensing, a robot needs to make decisions (think) to be able to adapt to the environment and to change the environment according to the mission [4, 10, 15].

No machine can survive in an environment without proper feedback from it. Mechanical and sensor errors can easily accumulate and put the robot in a dangerous situation. Thus, cognition is essential for robots' survival. Acting is another essential requirement for the evolution of intelligence. Acting is an ability to act on the environment to survive and accomplish the mission.

Manipulation and mobility are key components of action even though they may not be necessary at the same time for small robots. Most of the time, mobility is enough for small robots to accomplish the mission. This changes the current image of a robot from "one-armed iron-laborer" to "a mobile creature" mostly moved by wheels. In recent years, special attention has been given to robots mostly inspired from the nature (e.g. from human cooperation).

Multiagent society is encouraged by human society. Cooperation between lets them accomplish a complex mission with their rather limited skills. This type of behavior brings the communication component to the picture because robots can cooperate with each other only with effective communication between them. A considerable number of papers have been devoted to these topics [1, 5, 10, 12, 13, 14, 16].

The organization of the paper is as follows. A flexible design architecture based on multiagent technology is described in Section II. Then, in Section III, an modular mini robot is described following the concepts presented above. In Section IV is presented the robotic multiagent system, RoboSmith, which link togheter the mini robots. Finally, Section V gives the conclusions.

II FLEXIBLE DESIGN ARCHITECTURE

There has been a significant amount of research in reconfigurable, modular and flexible robotics in recent years [4, 6, 12]. Most of the research has been on multiple identical modules that construct a single robot.

The proposed architecture has a vertical modularity based on horizontal layers multiagent architecture in which the layers are not identical to one another. It slices a robot into functional abstract layers such as locomotion, control, sensors, communication, and actuation.

These concepts avoid main disadvantages of the horizontal layering (which require control of race conditions over the actuators) because there are a single agent which control actuators. Any mobile robot can be constructed by combining the above layers for a specific application.

A sub-module is a piece of hardware, which accomplishes the functionality of an abstract functionality (and also provide that skill for respective agent), i.e. a wireless communication sub-module for a communication layer.

In our flexible architecture, the robot can be constructed by combining a locomotion layer, sensory layers, an actuator layer, and purpose specific layers. The software flexibility is given by multiagent technology. Figure 1 presents the proposed hardware layers and figure 2 present the software functionality [12]. In the model, each layer can be implemented by the corresponding hardware.

For example, a sensor layer may be an ultrasound sensory board or a proximity sensory board, or perhaps both. The sub-modules are designed such that they have a unique signature and a standard pin connection. The sub-modules can be added at any level and the position does not effect its module's operation.

Since layers can be combined in any order, an application specific robot can be quickly constructed. For example, if a new problem domain requires legs rather than wheels, the wheeled sub-module can be instantly swapped with the legged sub-module. Also, in software application, the wheels agent is replaced with legs agent. This is essential for the flexibility of the applications since agents might be equipped with complementary skills instead of having the same skills.

Programming robotic applications is far from standardized. The primary reason is that each robot is composed of very special hardware designed for a specific goal. The result is that the software also becomes specific. This is very convenient to the multiagent systems which promote reengineering instead of reprogramming. Also the layers architecture can be easily implemented in the multiagent systems. The communications between agents is standardized by FIPA (Foundation of Intelligent Physical Agents) regulations. These facts make multiagent technology very useful in this situation.

A layered architecture is simultaneously reactive and deliberative. The deliberative agents reason and make decisions based on the symbolic representation (model) they have of the external world. These agents need a lot of effort to model the complex entities of the external world. The reactive agents suppose the existence of basic behaviors or sequences of actions that execute concurrently from the lowest level of intelligence. These behaviors are, in turn, used by more complex ones to create more complex levels of intelligence. A layered architecture contain a set of interacting layers in which some are deliberative and others are reactive. In horizontal layering the sensors are directly connected to each existing layer, which also might drive output directly.

The focus of this section moves from the architecture of the flexible mini robot to the architecture that a group of agents create a form. Individual agents are useless in the large majority of situations, because most of the scenarios involve several interacting agents.

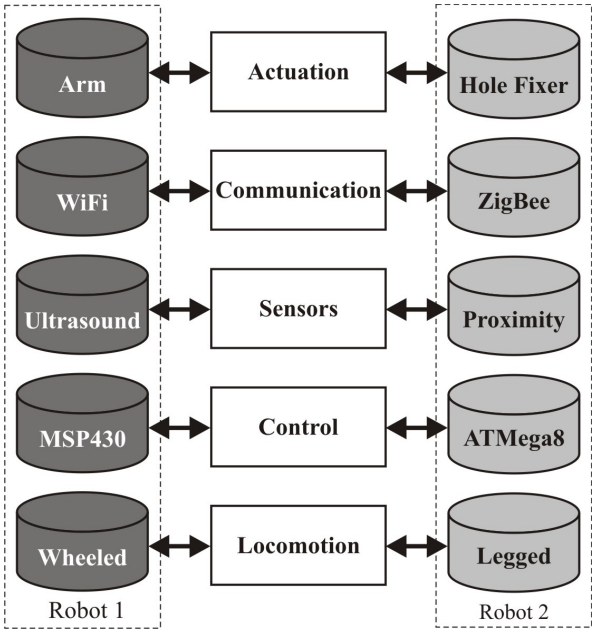


Fig. 1 Flexible hardware structure.

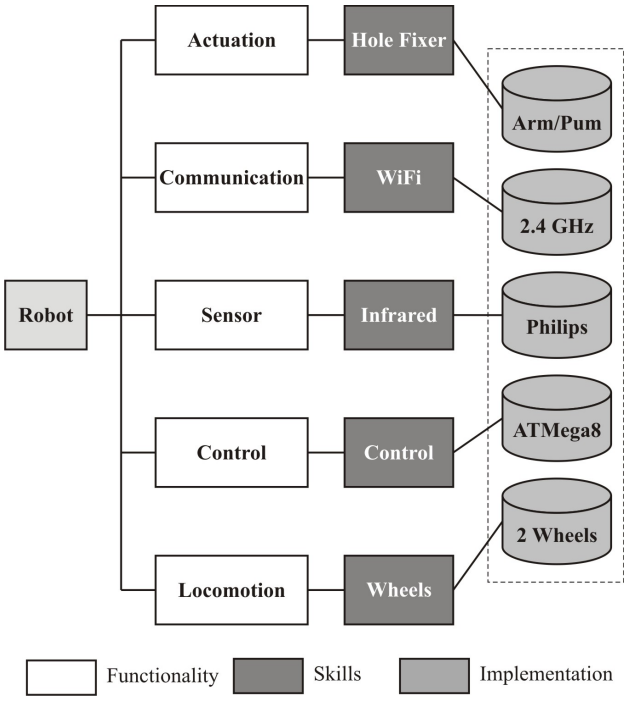


Fig. 2 Flexible software structure.

When dealing with multiagent the important aspects are now how they communicate and how they interact. Communication and interaction are the mechanism that let the community of agents have a more complex behavior than just the sum of their individual behaviors (see Fig. 3).

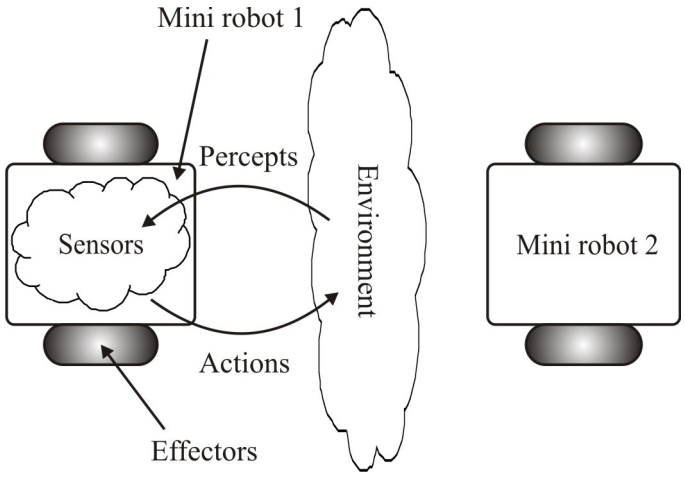


Fig. 3 Agents interactions with the environment.

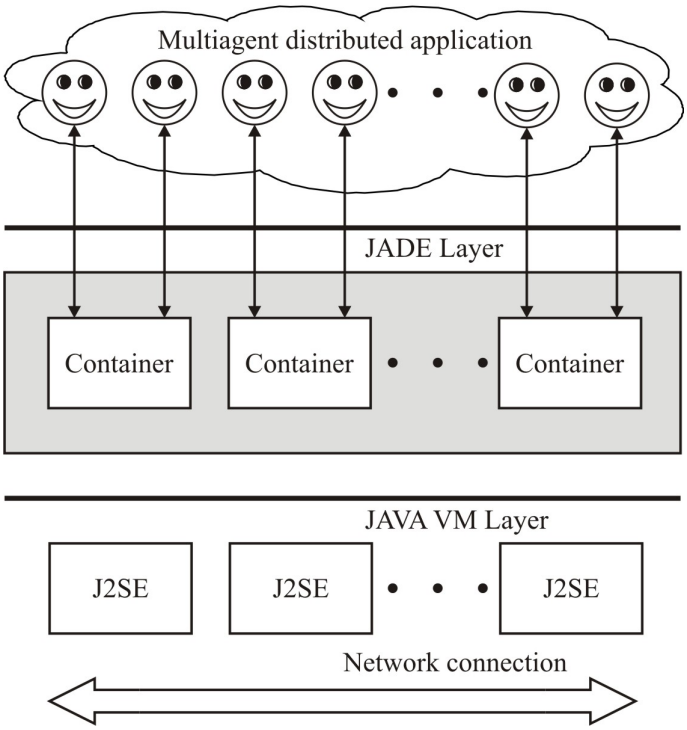


Fig. 4 The JADE architecture.

For implementation of multiagent systems we will use JADE (Java Agent Development Environment) which is a middleware platform intended for the development of distributed multiagent applications based on peer-to-peer communication [17]. JADE includes Java classes to support the development of application agents and the run-time environment that provides the basic services for agents to execute. An instance of the JADE run-time is called a *container*, and the set of all containers is called the *platform*. These platforms provides the layers that hides from agents the complexity of the underlying execution system. This mechanism is depicted in figure 4.

III The Mini Robot

The RoboSmith's robots are flexible mini robots, which take advantage of a layered design approach described in Section II. Even though there are five levels in the hardware and software architectures, the implementations (sub-modules) of the levels can be more than one. In addition, each level may also involve multiple closely related functionalities. The sub-modules are designed and manufactured at Automation Laboratory. This section presents the mini robot and main sub-modules,

Locomotion module has a mechanical base, and locomotion module hardware (sub-module). The base of the robot consists of an aluminum frame, two steppper motors, some gearing, two wheels and associated ball bearings, and the batteries. The base is designed by CAD tools and machined with high precision CNC machines. Figure 5 shows the base with wheels, gears and motors.

A legged version of the base is also in design process as an alternative locomotion to be used in different applications.

The battery selected for the mini robot is an AA form factor NiMH rechargeable cell. Four of these cells connected in series are used in the system. The cells are nominally 1.2 volts each for a system voltage of 4.8 volts.

The two wheels module is very versatile and easily directionable. This mechanism allows the robot to make short turns by moving only a wheel and stoping the other. Another advantage is provide by the sensors. The time delay is not critical because, in this situation, the locomotion module has plenty of time to turn and the control module has plenty of time to make decisions.

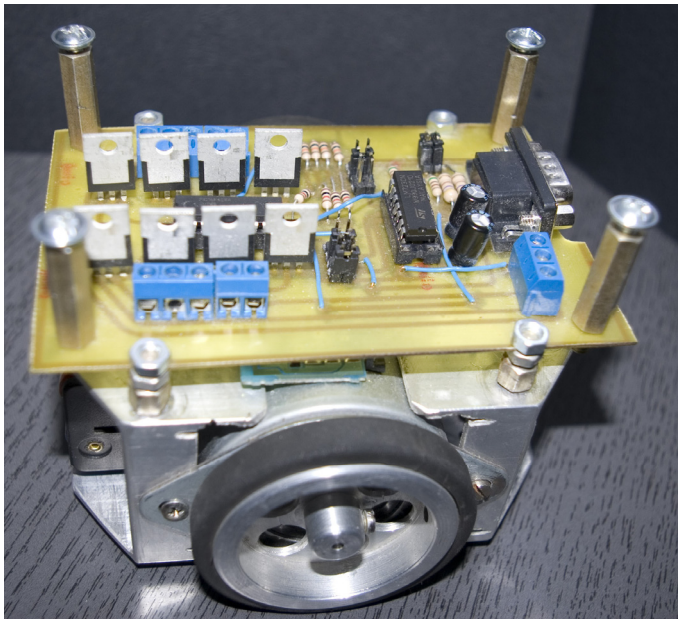


Fig. 5 The base with wheels for flexible mini robot.

The locomotion layer is implemented by first electronic module and the mechanichal base described above. It's most critical layer in the operation of the robot. It contains the circuit for the stepper controller, which provides direction control for the motors and

supplies the high current they require. This module is also manufactured in laboratory and include common components. It also includes the power system, which consists of a DC-DC converter and some passive components.

The power system provides +5 volts for the entire robot, and will accept from 1.5 - 15 volts on its input. This provides plenty of flexibility if a different battery system is put in place. This sub-module also contains the charge circuit, which allows the battery to remain in the robot while its being recharged (see Fig. 6). Also by choosing high capacity elements the robot have more autonomy.

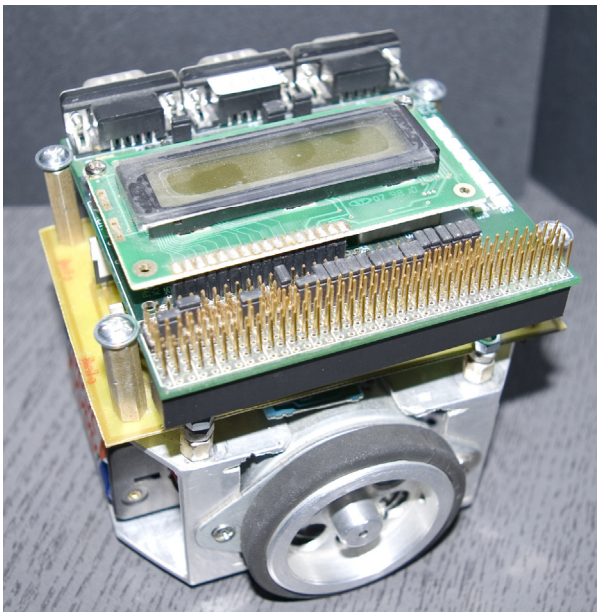


Fig. 6 Implementation of locomotion layer of flexible mini robot.

Over on this sub-module, (as seen in Fig. 7), is the main controller, an ATMega8 microcontroller running at 16MHz. The 8kB flash memory is included on chip and also 512B RAM and 1kB SRAM. All other components are soldered directly to the board. With improved memory architecture, the mini robots are able to run in a flexible architecture.

At the top, the sub-module for communication layer is based on a XBee harware board (serial version) (which is very simmilar to familiar ZigBee modules) [19].

This is necessary for agents interactions and for reporting to the main server unit which is hosted on a PC (see Fig. 8.). Another XBee module (an USB version this time) is connected to a host PC and connect the mini robot to the control program. Also many mini robots (a maximum of 16 is recommended) equipped with XBee module can be interconnected in this manner. The control program will coordinate the messages.

IV The Multiagent Robotic System

The RoboSmith architecture is based on a multiagent robotic system which coordinate the entire community of agents. This section presents the implementation of RoboSmith architecture.

The RoboSmith is a networked organization of mini robots, that have togheter formed a cooperative dynamic network to reach group benefits. RoboSmith is a society of agents (the mini robots) and therefore their interactions are at society level.

The mini robots can be considered intelligent agents because they are proactive, reactive and have social ability. They are proactive since they have goal directed behavior that is seen when the layers involved participate in society with the best possible performance.

Moreover, they can keep working even under environmental coalitions. They are reactive in the sense that they react to changes in the external environment, which are "sensed" through messages.

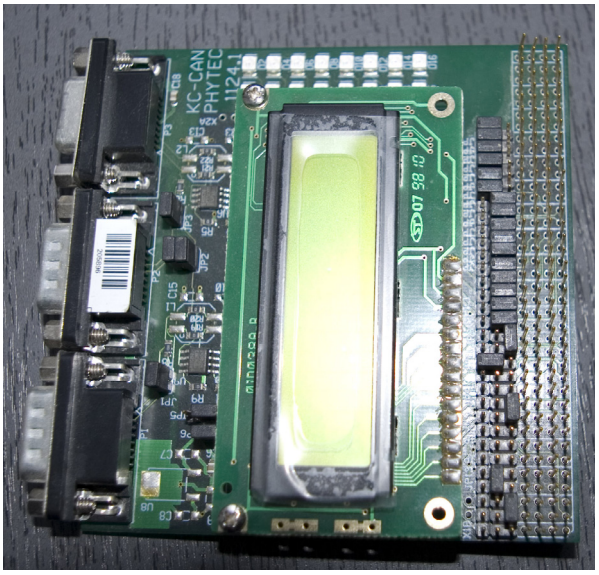


Fig. 7 Mini robot's CPU.

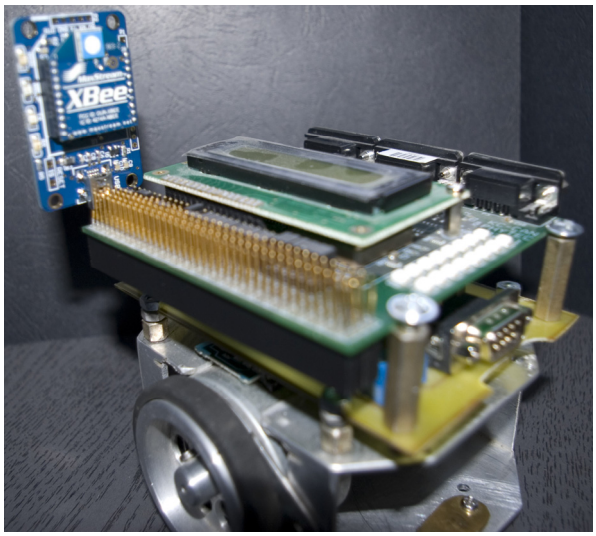


Fig. 8 Implementation of hardware communication layer.

Altough RoboSmith agents are not purely reactive, the importance of messages in their behavior is so relevant that their architecture is more reactive than proactive. Finally, they have social ability because they are able to negotiate and cooperate with the other mini robots.

Communication and interaction among individuals and about a domain can only take place if some conceptualisation of that domain exists. To guarantee a common semantic understanding,

agents must use an appropiate ontology to communicate with their partners. In the case of RoboSmith, all the mini robots need to share some basic concepts, such as skills, requests, services and agent. Therefore all agents of the proposed architecture share a basic global ontology that models the basic referred concepts.

For our purposes, we have adopted the description of an agent as a software program with the capabilities of sensing, computing, and networking associated with the specific skils of the mini robots described above.

This implementation is made in JADE because this development tools is very versatile and could be very well integrated with others development tools (like Protégé-2000 and Java [17,18]).

Also JADE is an open source FIPA compliant Java based software framework for the implementation of multiagent systems. It simplifies the implementation of agent communities by offering runtime and agent programming libraries, as well as tools to manage platform execution and monitoring and debugging activities. These supporting tools are themselves FIPA agents.

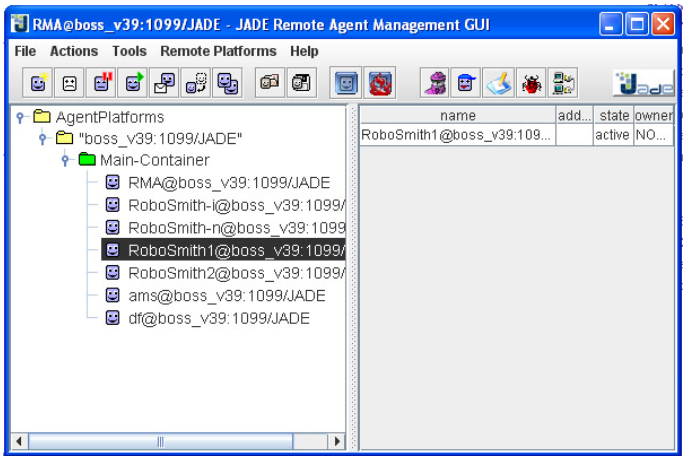


Fig. 9 JADE Implementation of RoboSmith.

JADE offers simultaneously middleware for FIPA compliant multiagent systems, supporting application agents whenever they need to exploit some feature covered by the FIPA standard (message passing, agent life cycle, etc), and a Java framework for developing FIPA compliant agent applications, making FIPA standard assets available to the programmer through Java object-oriented abstractions.

The general management console for a JADE agent platform (RMA – Remote Monitoring Agent), like in figure 9, acquires the information about the platform and executes the GUI (Graphic User Interface) commands to modify the status of the platform (creating new agents, shutting down containers, etc) through the AMS (Agent Management System).

The agent platform can be split between several hosts (provided that there is no firewall between them). Agents are implemented as one Java thread and Java events are used for effective and lightweight communication between agents on the same host. Parallel tasks can be still executed by one agent, and JADE schedules these tasks in a more efficient (and even simpler for the skilled programmer) way than the Java Virtual Machine (VM) does for threads. Several Java VM, called containers in JADE, can coexist in the same agent platform even though they are not running in the same host as the RMA agent.

This means that a RMA can be used to manage a set of VMs distributed across various hosts.

Each container provides a complete run time environment for agent execution and allows several agents to concurrently execute on the same host. The DF (Directory Facilitator), AMS, and RMA agents coexist under the same container (main-container) togheter with the RoboSmith's agentified mini robots, as it is shown in figure 9.

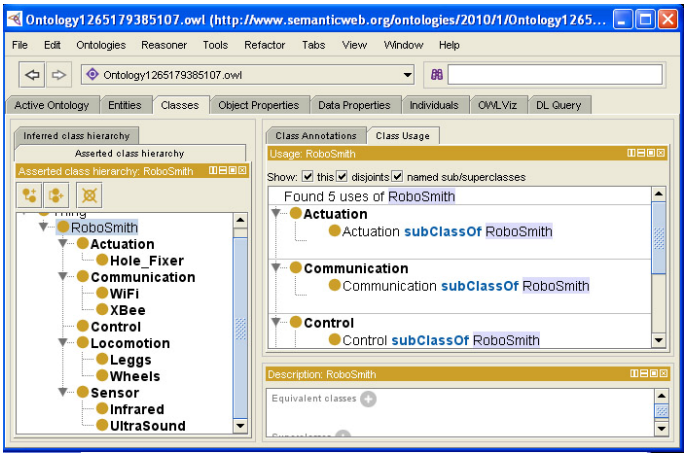


Fig. 10 Defining ontologies for RoboSmith.

To facilitate message reply, which, according to FIPA, must be formed taking into account a set of well formed rules such as setting the appropriate value for the attributes *in-reply-to*, using the same *conversation-id*, etc., the method *createReply*() is defined in the class that defines the ACL (Agent Communication Language) message.

Different types of primitives are also included to facilitate the implementation of content languages other than SL (Standard Languages), which is the default content language defined by FIPA for ACL messages. This facility is made with Protégé 2000 as depicted in Fig. 10.

In Fig. 11 is presented a graphical view of ontology classes which facilitate understanding the fact that from the point of view of the programmer, a JADE agent is simply a Java class that extends the base *agent* class.

It allows inheriting a basic hidden behavior (such as registration, configuration, remote management, etc.), and a basic set of methods that can be called to implement the application tasks of the agent (i.e. send/receive ACL messages).

Moreover, user agents inherit from their Agent superclass some methods to manage agent behaviors. Also this diagram can be represented in UML (Unified Modeling Language) because behaviors are implemented as hierarchy of classes.

The Protégé 2000 connect to RoboSmith JADE Agents by including a Protégé configuration file in the Java compiler. The RoboSmith ontology is divided in many concepts that folloes the class hierarchy defined above.

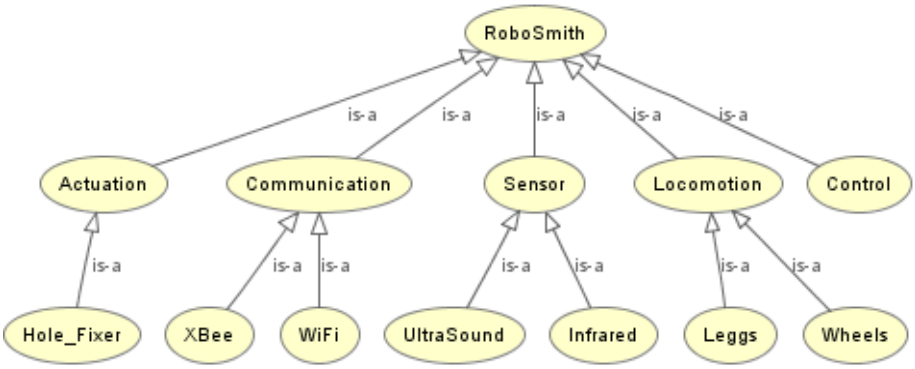


Fig. 11 Classes hierarchy for RoboSmith architecture.

V CONCLUSIONS

In this paper, the exploitation of multiagent technology applied in flexible mini robotic system have been presented. A number of two robots were constructed and implied in multiagent robotic system It is possible to extend this number by adding similar robots.

The results presented confirmed theoretical predictions made during design phase. The proper function of the robotic system is important from a practical point of view since it provides a detailed framework about the design of the control structure and the behaviors tasks. This confirm that multiagent technology can be successfully implemented for control the robotic systems.

REFERENCES

[1] J. Barata, and L.M. Camarinha-Matos, "Multiagent Coalitions of Manufacturing Components for Shop Floor Agility – The CoBaSA Architecture", *Int. J. Networking and Virtual Organisation*, 2(1), pp.50-77, 2003.

[2] B.S. Choi, J.J. Lee, "Localization for a mobile robot based on an ultrasonic sensor using dynamic obstacles", *J. Artificial Life and Robotics*, vol. 12, Springer Japan, pp. 280-283, March, 2008.

[3] V. Denneberg and P. Fromm, "OSCAR. An open software concept for autonomous robots", *Proc. of the 24th Annual Conference of the IEEE Industrial Electronics Society*, pp. 1192-1197, sep. 1998.

[4] L. Feng, J. Borenstein, and H. R. Everett, "*Where am I? Sensors and Methods for Autonomous Mobile Robot Positioning*", vol. 3, The University of Michigan, 1994, pp. 9-10.

[5] D., Floroian, *Multiagent Systems*, Cluj-Napoca (Romania), Ed. Albastra (In Romanian), 2009.

[6] W. Jueyao, Z. Xiaorui, T. Fude, Z. Tao, et al., "Design of a Modular Robotic System for Archaeological Exploration", *Int. Conf. on Robotics and Automation*, Kobe, Japan, pp. 1435-1440, 2009.

[7] T. Komatsu and N. Kuki, "Investigating the Contributing Factors to Make Users React Toward an On-screen Agent as if They are Reacting Toward a Robotic Agent", *18th IEEE Int. Symp. Robot and Human Interactive Communication*, Toyama, Japan, Sept. 2009.

[8] D., Lee, W., Chung, "Discrete-status-based Localization for Indoor Service Robots", *IEEE Trans. Ind. Electron.* Vol. 53, No. 5, Oct. 2006, pg.:1737–1746.

[9] W.H. Lee and A.C. Sanderson, "Dynamics and distributed control of modular robotic systems", *Proc. of the 26th Annual Conference of the IEEE Industrial Electronics Society*, pp. 2479-2484, sep. 2000.

[10] W.J. Opp and F. Sahin, "An Artificial Immune System approach to Mobile Sensor Networks and Mine Detection", *IEEE SMC 2004, the Hauge*, Netherlands, oct. 2004.

[11] E., Petriu, T., Whalen, R., Abielmona, A., Stewart, "Robotic Sensor Agents: A New Generation of Intelligent Agents for Complex Environment Monitoring", *IEEE Instrum. Meas. Mag.*, Vol. 7, No.3, Sep. 2004, pg.:46–51.

[12] F. Sahin, "GroundScouts: Architecture for a Modular Micro Robotic Platform for Swarm Intelligence and Cooperative Robotics", *Int Conf. Syst., Man & Cyb.*, 2004, pp.929-934.

[13] D. Weyns, H. V. D. Parunak, F. Michel, T. Holvoet, and J. Ferber, "Environments for Multiagent Systems State-of-the-Art and Research Challenges", *E4MAS* 2004 *Springer-Verlag*, Berlin, pp. 1–47, 2005.

[14] M., Wooldridge, N.R., Jennings, "Agent Theories, Architectures, and Languages: A Survey", Proc. *ECAI-Workshop on Agent Theories, Architectures and Languages*, Amsterdam (The Netherlands), Aug. 1994, pg.:145–160.

[15] J., Zheng, P., Lorenz, P., Dini, "Guest Editorial: Wireless Sensor Networking", *IEEE Netw.*, Vol. 20, No. 3, May/Jun. 2006, pg.:4–5.

[16] S.A. Suboting and Al. A. Oleinik, "Multiagent Optimization Based on the Bee-Colony Method", *Cybernetics and Systems Analysis*, Vol. 45, No. 2, 2009.

[17] ***, Java Agent Development Framework – JADE, http://jade.tilab.com/

[18] ***, Protégé 2000, http://protege.stanford.edu/

[19] ***, XBee, http://www.digi.com

Custom Interface Elements for Improved Parameter Control in Volume Rendering

Marius Gavrilescu, Muhammad Muddassir Malik, and Eduard Gröller

***Abstract*— In volume visualization interfaces, rendering-related parameters are often manually editable through various controls and interface elements. Most of the time however, these offer little or no beforehand information on the resulting effects that would occur for certain parameter values or across the whole value domain. This makes parameter adjustment a trial and error process. We have developed techniques to anticipate these changes and display them on customized versions of popular interface elements, such as sliders or transfer function editors. Through the use of visualization means such as graphs, color mapping, and various other indicators, the influence of potential parameter changes on the volume rendering output can be assessed before any actual changes are made. This makes it easier for the potential user to work with such interfaces, while receiving feedback on parameter behavior and stability.**

I. Introduction

Volume visualization is a segment of computer graphics which deals with the exploration, classification and on-screen representation of information from three-dimensional, or often multi-dimensional datasets. Such datasets are typically acquired from scanning devices such as Computed Tomography (CT), Magnetic Resonance Imaging (MRI) or Rotational X-ray imaging. Volume visualization techniques are widely used in medical and industrial imaging, where a 3D representation of the available data is often more accessible, suggestive and visually appealing than traditional 2D grayscale slices, and may yield information otherwise hard to spot. Volume data is intuitively composed of atomic elements known as *voxels* (short for *volume pixel*), which are the equivalent in 3D space of the traditional pixels from 2D images. For the purposes of computer processing, voxel-based data is at first discretely sampled, then the samples are traversed and assigned optical properties, after which they are composited and projected into a 2D screen-space to produce an image. The previously mentioned steps are a loose outline of a generic volume rendering approach.

A typical volume rendering application consists of one or several viewports to display the images resulting from rendering the dataset. Multiple controls are used to manipulate the data processing. The complexity of these controls ranges from a simple slider to elaborate transfer function editing interfaces. While there exist numerous efforts to automate or semi-automate the visualization of this type of data [1], [2], [3], many volume rendering applications mostly leave it to the user to adjust the various parameters which control the on-screen outcome. Therefore, in many cases, an image which shows relevant information from a volume dataset is the result of parameter tweaking by means of sliders, interactive graphs, various widgets, and generally speaking, a variety of interface elements.

There is, however, a downside to allowing an extensive degree of manual control. Unless the user is very familiar with the particular dataset under analysis, the adjustment of parameters to obtain the desired results may prove to be a tedious and time-consuming trial-and-error task. Furthermore, while most volume rendering applications allow extensive control over the data, few if any relay feedback to the user as to how a hypothetical change in a parameter value might influence the resulting images. We attempt to reverse this situation through the development of interface elements which provide the user with a-priori knowledge into how a change in the interface control would reflect on the on-screen image. This would also aid in the assessment of parameter behavior and stability across its value-domain.

The paper is structured into several sections. After the introduction, we briefly outline volume rendering in general, and describe the rendering approach used in the paper. Section three deals with the metric used for comparing images resulting from sampled parameter values, and how this metric relates to the human vision system. In the following section, we present a couple of custom interface elements which allow control over their associated parameters, while at the same time automatically displaying information on parameter behavior and stability.

Manuscript revised on 31.08.2010. This work was supported in part by the "BRAIN - An Investment in Intelligence" doctoral Scholarship Program, within the Technical University of Iasi, Romania

The presented work has been partially supported by the Bridge-Project SmartCT and the K-Project ZPT (http://www.3dct.at) of the Austrian Research Promotion Agency (FFG).

Marius Gavrilescu is a PhD student from the Technical University of Iasi, Romania, and collaborates with the Vienna University of Technology, Austria within a joint PhD program (e-mail: mariusgv@cg.tuwien.ac.at, mariusgav42@yahoo.com).

Muhammad Muddassir Malik is with the School of Electrical Engineering and Computer Science, National University of Sciences and Technology, Pakistan (email: mmm@cg.tuwien.ac.at).

Eduard Gröller is associate professor at the Vienna University of Technology, Austria and adjunct professor of computer science at the University of Bergen, Norway (e-mail: groeller@cg.tuwien.ac.at).

We also present an on-screen approach for the dynamic visualization of parameter changes. We conclude by briefly emphasizing the significant aspects of the paper and by providing information on future work.

II. VOLUME RENDERING APPROACH

While a description of volume rendering techniques does not fall within the scope of this paper, we find it necessary to at least outline the basic methodology and point out the elements which are relevant to the contents of other sections.

The images found throughout the paper are produced using direct volume rendering (DVR). Unlike older techniques which employed "proxy geometry" [4] and triangle based surfaces to indirectly outline elements of interest from within the volume, DVR operates on the actual volume data, without the need to use additional geometric primitives. The dataset is typically uploaded into video memory as a 3D texture [5] and sampled discretely. A transfer function maps optical properties to the samples, which are then composited to form the desired image in screen-space. A popular algorithm which encompasses these steps is *ray casting* [4], which has the advantage of exploiting the hardware acceleration capabilities and the parallel architecture of modern graphics processing units (GPUs) [6], [7]. Fig. 1 shows an example of an image rendered via GPU ray casting. The corresponding transfer function is shown below the image.

The transfer function depicted in Fig. 1 maps opacity values to voxels according to their densities [8]. Regions of lower density, located toward the left of the graph in Fig. 1, have a very low opacity, which is reflected in the transparent appearance of the skin in the image rendered above the graph. Similarly, higher density regions such as bone are assigned a much higher opacity and are fully visible. Many volume rendering applications allow manual control over the shape of the transfer function. The control points in Fig. 1, marked with circles, are movable with the mouse and the in-between step-wise components of the function may be linear, cubic or may otherwise have any desired shape. The resulting image changes in real time to reflect the changes in the transfer function. The problem, as previously mentioned, is that such an interface offers little additional information on parameter effect. In other words, the users cannot know what the rendered image will look like if the transfer function is given a certain shape until they actually modify the transfer function. In Section 4, we present our method to address this issue.

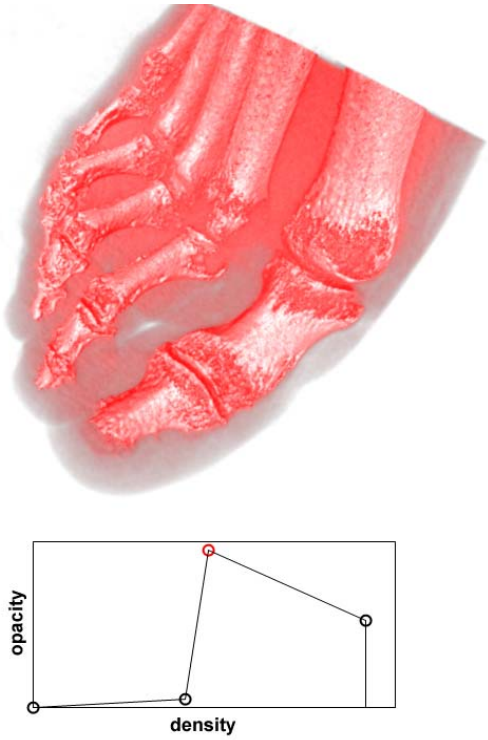


Fig. 1. Volume rendering of a medical dataset and the corresponding opacity transfer function

III. METRICS FOR IMAGE COMPARISON

Our approach to solving the problem of the lack of information in interface controls is mostly an image-oriented one. Given a particular parameter, we sample it across its value-domain and render an image for each sample. By comparing the resulting images we get an idea of how the parameter behaves across its domain, and how different values affect the on-screen outcome. This makes it possible to tailor popular interface controls to also show this parameter behavior, in addition to allowing control over its values. Image comparison is therefore an important piece of the puzzle and the choice in comparison metrics may significantly affect the outcome.

The metrics often involve a pixel-by-pixel comparison of the images, using some type of formula to assess the differences between color or intensity values, followed by an accumulation of these differences in a scalar. Therefore, for each pair of sampled parameter values, we end up with a scalar which shows the difference between using one sampled value versus using the other one. The domain of the parameter would then be characterized by an array of such scalars.

Among the most straightforward and computationally efficient metrics is the absolute mean difference, which essentially involves computing the mean of all pixel differences in the RGB color space, across the two compared images. Wilson et al. [9] provide an introduction

into some of the more common metrics, such as the root mean square (RMS), the signal to noise ratio, as well as their own Δ_g metric, based on the Sobolev norm and the Hausdorff metric. Chan et al. [10] have developed an image comparison method based on the Canny edge detection algorithm. However, such metrics are mathematically defined around pixel differences. They do not take into account aspects pertaining to subjective human perception, though they may incidentally correlate with the human vision system. Furthermore, it is difficult to assess the robustness and efficiency of such metrics, since it is the user who has to relate the information obtained from applying the metrics to actual on-screen changes.

A more straightforward approach to defining a suitable metric is to design bottom-up. We design a metric around the human vision model, considering aspects of the human perception of color and intensity. Efforts in this direction have been made and are well documented in literature [11], [12], [13]. Such metrics could be described as *perception-based image comparison metrics*. They take into account factors such as hue angle, color distance, pixel placement or an estimated viewer distance, among others. As is often the case, there is also a trade-off between the complexity of the metric and the processing speed. Perception-based metrics are typically computationally intensive. As the complexity of the metric increases, performance becomes a significant problem, and the processing of complex metrics at high resolutions and for hundreds of parameter values may take hours even on a relatively powerful machine.

Considering the above, we have developed a metric which takes into account some of the previously mentioned perceptual aspects, while attempting to provide an efficient means of image comparison. The metric is processed in the following steps:

- the general area of the volume in screen space is isolated from the rest of the image, since the background presents no relevant information

- the behavior of the human eye, which only looks at a few details at a time as opposed to the image as a whole, is approximated by analyzing a finite number of random rectangular sub-regions within the image. These regions are then weighed according to their size and color uniformity. This is based on an approach proposed by Matkovic [14].

- a noise removal filter is applied, since changes in noise have little impact on the perceived change in the image. In other words, noise, even when it changes significantly, is still perceived as noise.

- for the purpose of assessing pixel differences, we change to the CIE-Lab color space [15], which is perceptually closer to the human interpretation of color than the RGB model.

- the accumulated difference for corresponding rectangular regions between each pair of images is calculated using the color distance ΔE^*_{ab} [15].

- finally, the amount of variation between each pair of images is computed as the weighted mean of the values from the rectangular regions.

The accuracy and relevance of the metric are difficult to assess, since they are, to a significant degree, subjective matters. Performance-wise, migrating some of the previously mentioned steps to the GPU has shown improvement, though still insufficient for use in real time. However, the trade-off between complexity and speed has so far proven satisfactory on commercial hardware.

IV. Custom Interface Controls

For demonstrative purposes, we consider two basic parameters involved in volume rendering: the step size used when sampling the volume during ray casting and a basic transfer function control which adjusts the threshold of an isosurface and its opacity.

The information regarding the behavior of these parameters across their domain is incorporated into common interface elements by means of information visualization techniques.

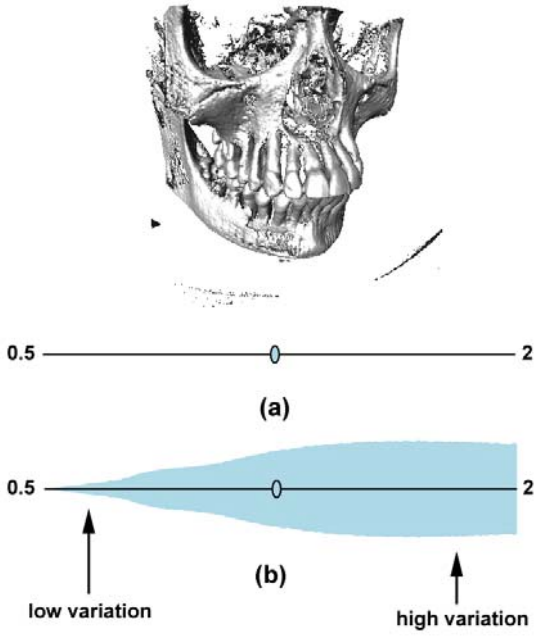


Fig. 2. Rendered dataset with traditional and custom sliders which control the step size. The custom slider also depicts information on parameter behavior

Slider widgets are frequently used for the adjustment of one-dimensional parameters such as the step size. However, a basic slider does not provide any a-priori information regarding the changes that would happen upon

changing the position of the pointer (Fig. 2a). Unless the user is very familiar with the volume under consideration, moving the slider to get the desired result is a matter of trial and error. The slider in Fig. 2b shows the magnitude of change which occurs in the rendered image when the pointer is moved to a neighboring location.

The slider may be further customized as needed, as seen in Fig. 3. Often it might be necessary to more closely inspect a region of the slider where significant changes occur. For this purpose, the specific region can be selected (Fig. 3a) and non-linearly scaled to fill a larger portion of the slider (Fig. 3b). It can take up the entire available length. The borders of the regions keep their initial values, but there is more space along the length of the slider thus allowing more precision in selecting a desired position for the pointer. This process of fine tuning is useful for portions of the slider where there is an abrupt variation in the amount of parameter effect.

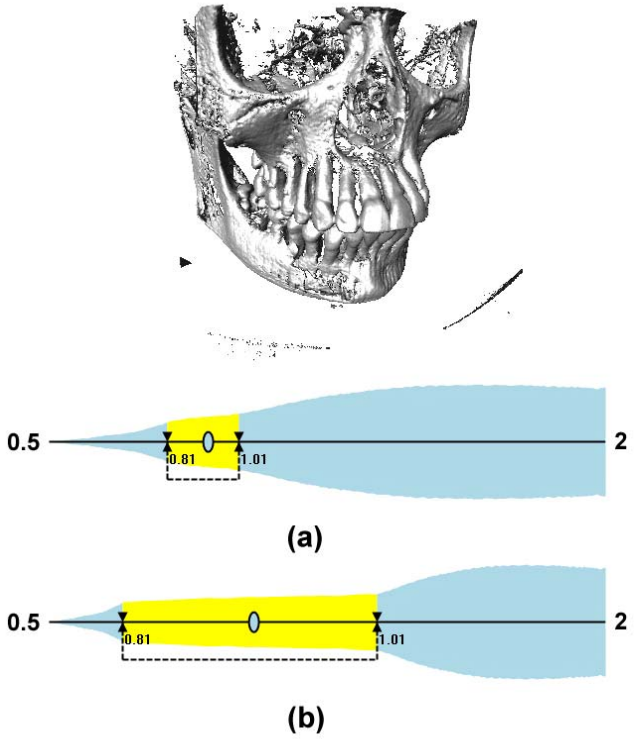


Fig. 3. The selection (a) and non-linear scaling (b) of a slider region for fine tuning purposes

The technique may be fully automated by partitioning the slider into small regions and assigning each region a portion of the slider of a length proportional to the variation taking place within the region. We thus end up with a *perceptually uniform* slider, where the perceived changes in the rendered image are proportional to the distance by which the pointer was moved along the slider.

Performance-wise, we took 300 samples from the domain of the parameter, to generate values for the graph of the slider. We rendered and compared the images obtained for each pair of sampled parameter values. The computations for processing the custom slider and rendering the volume in Fig. 2 took approximately 16.2 seconds on our test machine, an Intel Core I7 with 6 GB of RAM and a GeForce GTX 280 GPU.

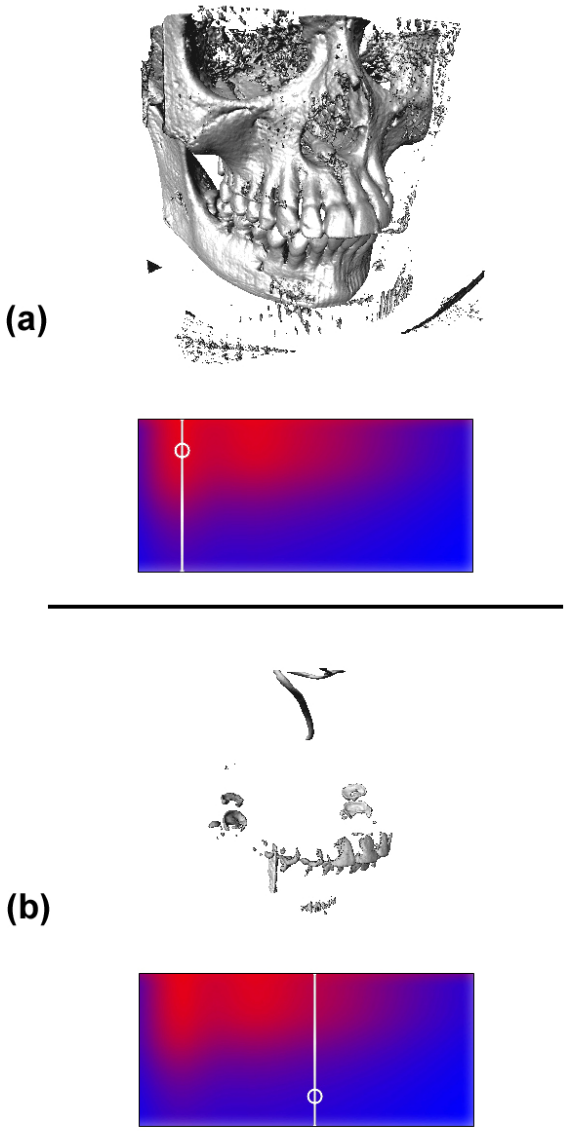


Fig. 4. Custom isosurface control where the magnitude of change is color-coded. (a) and (b) show two positions of the pointer at opposite ends of a region of transition which signifies a variation in parameter stability

The concept of parameter behavior displayed on a custom version of a frequently used interface element also extends to transfer function controls. For this purpose, we consider a simplified version of the editable graph depicted in Fig. 1. The simplified graph allows the adjustment of an isosurface threshold when the pointer is moved horizontally, and the adjustment of the opacity of the isosurface, when the pointer is moved vertically. As

previously, the parameters are sampled and the corresponding images are rendered and compared.

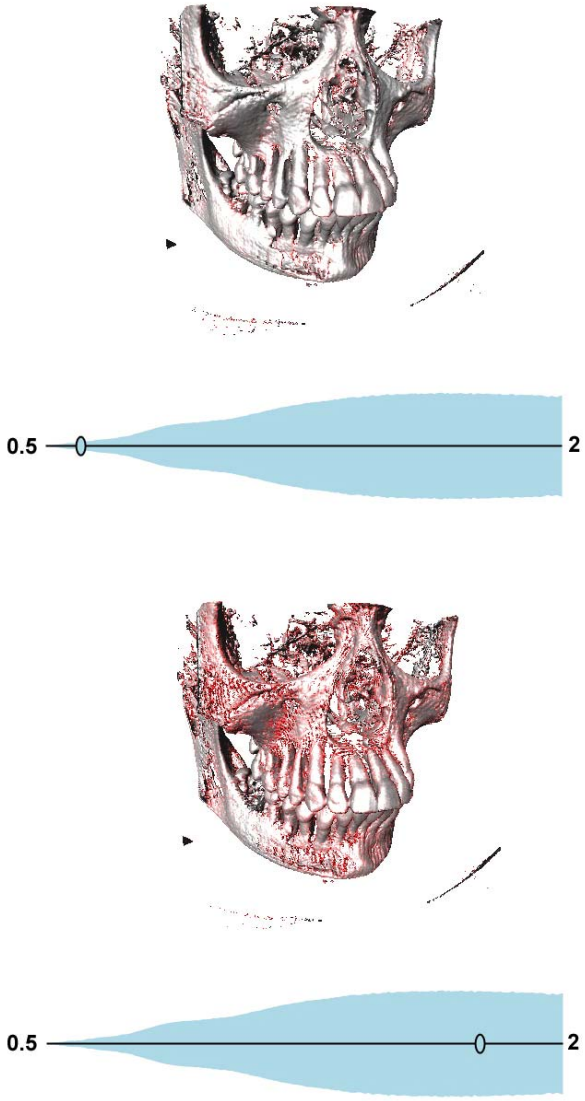


Fig. 5. Changes illustrated on the rendering of the volume for two positions on the step size slider

Fig. 4a and 4b show an isosurface control where the influence of parameter changes is color coded through a red-blue color range. Red areas correspond to positions of the pointer which cause significant changes in the rendered image (shown above the control), whereas blue areas denote regions where the change in parameter values has little effect on the outcome. The purple transition area in-between the red and blue is a region of varying stability, whereby the parameters gain stability as the pointer is moved from the red region toward the blue one.

We took 50 samples on the horizontal axis and 20 on the vertical axis, for a total of 1000, to generate values for color coding. Our test machine took around 49 seconds to process the isosurface control from Fig. 4.

Changes occurring when manipulating an interface control can be shown on the rendered volume itself, as depicted in Fig. 5. The changes are shown using a red coloration of varying intensity. When the pointer is in the low variation region on the slider, little changes are visible. If the pointer is moved to an area showing greater variations, this is reflected in the more frequent intensely red regions on the volume.

However, this approach has the downside of being intrusive and inflicting possibly unwanted changes in the final resulting image. The direct color encoding of image variation may obstruct desired information when superimposed on the volume rendering. One way to avoid this is to use a *viewing lens*, i.e. to restrict the display of changes to a bordered circular region which the user can freely move using the mouse, as illustrated in Fig. 6.

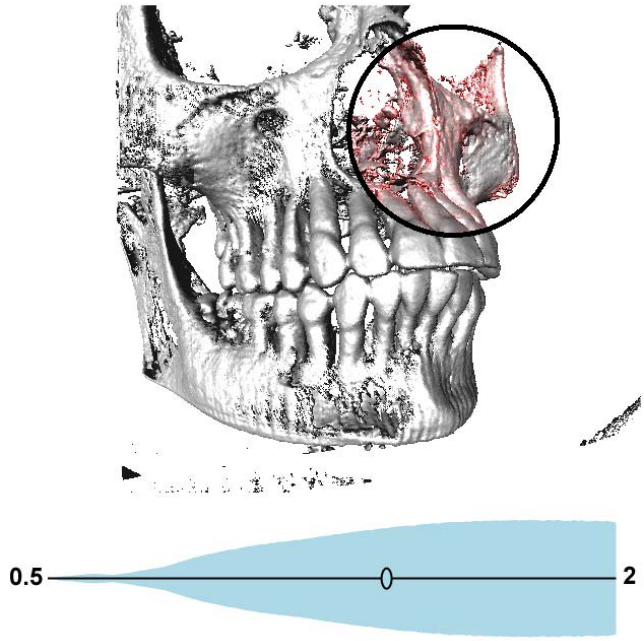


Fig. 6. The viewing lens restricts the display of changes to a mouse-controlled circular region

V. CONCLUSIONS

Interface elements customized to offer information on parameter behavior are meant to make parameter adjustment more efficient and straightforward. The approaches described thus far do not provide an exhaustive analysis of the parameters involved in volume rendering. They do offer a-priori information on the variations which would manifest upon certain changes in the discussed parameters, which may aid in making volume exploration and related interface elements easier to interact with.

There is naturally a lot of room for improvement and extension in this area, mostly in the direction of expanding

the mentioned techniques to more complex contexts, while at the same time working on new information visualization methods for the purposes of parameter analysis. Future development in this direction includes the extension of the concept of customized interface elements to single and multidimensional transfer function controls, while focusing on other potentially more relevant parameters and improving the metrics used for image comparison.

ACKNOWLEDGMENT

Marius Gavrilescu thanks the members of the Vis-Group from the Institute of Computer Graphics and Algorithms of the Vienna University of Technology for their invaluable support and help with research. He would also like to thank Professor Vasile Manta from the Technical University of Iasi, Romania, for his help and supervision.

REFERENCES

[1] J. Zhou, M. Takasuka, "Automatic transfer function generation using contour tree controlled residue flow model and color harmonics", *IEEE Trans. Vis. Comput. Gr.*, vol. 15, no. 6, pp.1482-1488, 2009.

[2] P. Kohlmann, S. Bruckner, A. Kanitsar, and E. Gröller, "LiveSync: Deformed Viewing Spheres for Knowledge-Based Navigation", *IEEE Trans. Vis. Comput. Gr.*, vol. 13, no 6, pp. 1544-1551, October 2007.

[3] F. F. Bernardon, L. K. Ha, S. P. Callahan, J. L. D. Comba, and C.T. Silva, "Interactive Transfer Function Specification for Direct Volume Rendering of Disparate Volumes," SCI Institute Technical Report, No. UUSCI-2007-007, University of Utah, 2007.

[4] M. Hadwiger, J. M. Kniss, C. Rezk-Salama, D. Weiskopf, and K. Engel, *Real-time Volume Graphics*. Wellesley, MA: A K Peters, 2006.

[5] O. Wilson, A. Van Gelder, and J. Wilhelms, "Direct volume rendering via 3D textures", University of California, Santa Cruz, CA, Tech. Rep. UCSC-CRL-94-19, 1994.

[6] A. Kratz, M. Hadwiger, R. Splechtna, A. Fuhrmann, and K. Bühler, "High quality volume rendering of medical data for virtual environments", in *Proc. Int. Conf. Computer Aided Surgery Around the Head (CAS-H 2007)*, Innsbruck, Austria, 2007, pp. 83-85.

[7] I. Viola, A. Kanitsar, and E. Gröller, "GPU-based frequency domain volume rendering", in *Proc. Spring Conference on Computer Graphics (SCCG 2004)*, Budmerice, Slovakia, 2004, pp. 49-58.

[8] G. Kindlmann, "Transfer functions in direct volume rendering: Design, interface, interaction", in *SIGGRAPH 2002 Course Notes*, 2002.

[9] D. L. Wilson, A. J. Baddeley, and R. A. Owens, "A new metric for grey-scale image comparison", *Int. J. Comput. Vision*, vol. 24, no. 1, pp. 5-17, 1997.

[10] W. C. Chan, M. V. Le, and P. D. Le, "A wavelet and Canny based image comparison", in *Proc. 2004 IEEE Conf. Cybernetics and Intelligent Systems*, Singapore, pp. 329-333.

[11] M. Pedersen, and J. Y. Hardeberg, "A New Spatial Hue Angle Metric for Perceptual Image Difference", in *Computational Color Imaging: Second International Workshop (CCIW 2009)*, Saint-Etienne, France, 2009, pp 81-90.

[12] S. I. Titov, "Perceptually Based Image Comparison Method", in *Proc. Int. Conf. Graphicon 2000*, Moscow, Russia, 2000, pp 8-16.

[13] J. P. Farrugia, S. Albin, and B. Peroche, "A perceptual adaptive metric for computer graphics", in *Proc. WSGC2004*, Plzen, Czech Republic, 2004, pp. 49-52.

[14] K. Matcovic, "Tone mapping techniques and color image difference in global illumination", Ph.D. dissertation, Vienna University of Technology, Austria, 1997.

[15] Y. Ohno, "CIE fundamentals for color measurements", in *Proc. IS&T NIP16 International Conference on Digital Printing Technologies*, Vancouver, Canada, 2000, pp. 540-545.

Cell Decomposition-Based Strategy for Planning and Controlling a Car-like Robot

Narcis Ghita, Marius Kloetzer

Abstract – **We address the problem of planning and controlling the motion of a car-like robot, such that a desired position is reached, by evolving in an environment cluttered with obstacles. The developed solution consists of a planning part which is performed offline, followed by a control part. The planning part includes an iterative procedure that integrates cell decomposition methods for generating a coarse angular path, and car-like curvature constraints for generating a smooth implementable trajectory. The control part consists of a simple close loop trajectory following. Illustrative examples that support the developed approach are presented.**

I. Introduction

Navigation of mobile robots is an important robotics problem that continues to receive a lot of attention. The goal of a classical navigation problem for a mobile robot is to find a control strategy such that, by starting from a given initial state, the robot reaches a desired position, by avoiding collisions with existing obstacles [1], [2], [3]. Navigation is usually performed by having a complete or partial map of the environment in which the robot evolves, and information about the current state (localization). Thus, the navigation problem includes specifying, planning and controlling the robot's movement. The research in this area is focused on two main directions. On one hand, some researchers aim to reduce the limited expressivity of the classical navigation task, by developing automated strategies allowing high-level, human-like specifications for the motion task [4], [5], [6]. On the other hand, there is an increased interest in solving the classical navigation problem for robots with complicated dynamics, such as nonholonomic or underactuated systems [7], [8].

The existing solutions for the classical navigation problem can be divided in three categories [9]: methods based on constructing global potential functions (navigation functions), constructing roadmaps for the free configuration space (e.g., visibility graphs or Voronoi diagrams), decomposing the free space into cells (geometric shapes) and creating a graph corresponding to the adjacency relation between cells. All these methods assume that the environment map is known and that the robot is reduced to a fully-actuated point in the free configuration space. Moreover, some of these methods have some additional specific drawbacks, e.g., the potential functions can have local minima, construction of navigation functions without local minima is in general very difficult, while the methods based on visibility graphs result in robot trajectories that are very close to obstacles.

An impressive amount of research on path planning for mobile robots focuses on simple robot dynamics, either fully actuated or underactuated with differential-wheel driven structure (close to unicycle). Although such approaches yield useful results, a more realistic approach from the point of view of existence of large scale mobile robots is to focus on car-like robots. The used math models are usually formulated either in a fixed frame of the environment, or in frames that moves along the desired path of the robot [9].

The goal of this paper is to develop an algorithmic method for constructing a solution to the navigation problem for a car-like robot. We assume that the car-like robot evolves in a planar and bounded environment, cluttered with static and convex polygonal obstacles. The complete map of the environment is assumed to be available.

The solution we propose includes the following main steps: first, by using a cell decomposition method, we find an angular path, consisting from a succession of line segments, which links the start position with the goal position of a specific reference point on the robot. Second, by taking into account the steering capabilities of the car-like robot, we construct a smooth path, consisting of line segments and arcs of circles. Third, by considering the physical dimensions of the robot and the smooth path it should traverse, we use a procedure that iterates the first two steps until the obtained smooth path is correct, in the sense that the robot can follow it and during the motion no obstacle is hit. Finally, by assuming that the position of the robot can be always read, we employ a feedback controller that has the obtained smooth path as a reference trajectory for the robot. The first three steps of our solution form the planning part, and they can be performed offline (before the actual movement). Thus, the computational load is reduced during the real-time control of the car-like robot.

Related work and contributions of the paper:

As in [10], [11], we construct a solution to a navigation problem by using a cell decomposition method. However, our method mainly focuses on planning the motion of car-like robots, while [10], [11] are interested in controller design, and their strategies are conservatively adapted to unicycle robots.

The authors are with The Department of Automatic Control and Applied Informatics at the Technical University "Gheorghe Asachi" of Iasi.

The corresponding author is M. Kloetzer.

The construction of the smooth path from our approach seems related to constructing trajectories for car-like robots by using Dubins curves [12], [13], [14], [15]. However, the focus in [12], [16] is on finding local implementable trajectories the robot can follow, while taking into account the initial and final orientation of the robot as well, but while ignoring any obstacle. We aim in generating an implementable trajectory that guarantees a collision-free movement between obstacles and that includes arcs of circle with different radii, while all arcs from Dubins curves have the same radius.

A path planning and control policy for nonholonomic systems are described in [7], [8] where the environment is no longer decompose in polygonal cells, but the boundary in the local coordinates is a smooth surface. In order to simplify the control, the dimension of environment is increased, by considering the third dimension as the orientation of the robot. Due to complexity issues, this path planning algorithm is mainly designed to work in environments with relatively few obstacles.

For exploring with car-like robots large environments where the map is unavailable before the experiment, [17], [18] proposes an approach named framed quad-tree. Although in [17] the main problem consists in exploring the environment, some advantages of using quad-trees for path planning are stated.

In order to check if the robot does not collide with obstacles, we will use a slightly over approximated path derived from the basic construction proposed by [19]. Unlike [19], we initially generate a path for a specific point on the robot by using a cell decomposition method.

An overview of feedback control techniques for path following car-like robots can be obtained from [20], [21], [22].

The remaining of this article is organized as follows. Section II presents some preliminary material, while Section III proposes a path generating algorithm for car-like robots. In Section IV we briefly discuss the control strategy allowing the motion along the generated path. The theoretical developments are illustrated in Section V by an example using trapezoidal decompositions. The final section comments on the importance of our work and possible future research directions.

II. PRELIMINARIES

For the car-like system we consider the kinematic model given by equation (1):

$$\begin{cases} \dot{x} = v\cos(\theta) \\ \dot{y} = v\sin(\theta) \\ \dot{\theta} = v\tan(\phi)/L \end{cases}, \quad (1)$$

where (x, y) are the Cartesian coordinates in a fixed frame (S) of the reference point P_m, located at mid-distance of the actuated wheels, angle θ characterizes the robot's chassis orientation with respect to frame S, and L is the distance between the rear and front axle. The control inputs are v, which is the vehicle's velocity, ensured by the rear wheels, and ϕ, which is the vehicle's steering wheel angle, due to the front wheels, and measured with respect to the current chassis orientation, as depicted in Fig. 1 a). We will assume a constant velocity v, and a steering angle restricted by physical limits of the car-like robot, $\phi \in [-\phi_{\max}, \phi_{\max}]$. For a detailed development of model (1) we refer the interested reader to [3], [23], while a discussion on when a kinematic model is sufficient and when it is necessary to use a dynamic model for a car-like robot is presented in [9].

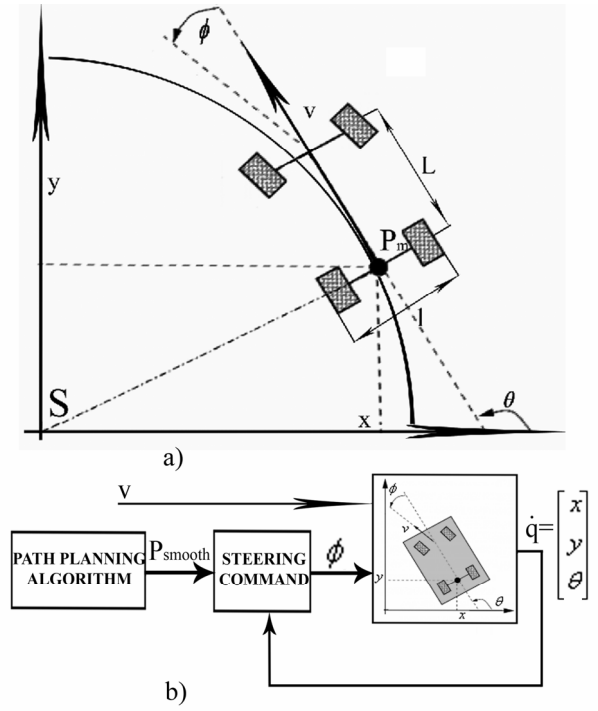


Fig. 1.a) Representation of car-like robot in a static frame S in fig. b) The feedback control architecture

The problem we solve in this paper can be formulated as:

Problem 1: Given a car-like robot with model (1), which can evolve in a bounded environment cluttered with convex polygonal obstacles, a start position (x_0, y_0) and a final position (x_f, y_f), find a control strategy that drives the robot to the goal (final) point while avoiding any collision with obstacles.

We assume that the initial orientation of the robot's chassis can be chosen, while the final orientation is not of interest to us. Furthermore, we will assume that, as any obstacle, the environment is also bounded by a convex polygon, such an assumption being easily accomplished with a bit of conservatism for any arbitrarily-bounded environment. Also, we mention that the assumption of having convex obstacles is not restrictive, since any concave polygonal obstacle can be represented by some overlapping

convex ones. We will denote the obstacles by O_1, …, O_n, and the environment by E.

In solving Problem 1, we will use cell decomposition methods, which are shortly defined in the remainder of this section. A formal treatment of such methods would go beyond the scope of the current paper, and we mention that algorithms performing the mentioned decompositions are available.

A cell decomposition is a partition of the free space (the part of the environment not occupied by obstacles) into polygonal regions of the same type. Our approach can be used for cell decompositions where the cells are convex polygons, typical examples including trapezoidal cells [2], triangular cells [11], [24], polytopal cells [5], or rectangular cells resulted from a quad-tree decomposition algorithm [10], [24]. By using a cell decomposition algorithm, the free space in which a point robot can move is abstracted to a finite graph $G = (C, A)$, where each node $C_i \in C$ corresponds to a cell, and the edges (arcs) from A correspond to adjacency among cells. With a slight abuse of notation, we will use C_i, $i = 1,...,|C|$, to denote either the label of a node from the graph, or the whole polygonal cell corresponding to that node. Furthermore, we assume that A has the form of a symmetric adjacency matrix, $A \in \square^{|C|\times|C|}$, where means there is an arc linking C_i with C_j, with weight (cost) $A_{i,j}$, and $A_{i,j} = 0$ otherwise.

Since the cells we are dealing with are bounded convex polygons, we mention that such a region can be described in two equivalent ways: as the convex hull of its vertices (usually the vertices result from a cell decomposition algorithm), or as the intersection of a set of closed half-planes (form that is used for example in LP problems). These two representations are denoted by V-representation and H-representation, respectively.

III. Path Generating

For solving Problem 1, we aim in generating a smooth trajectory accomplishing the imposed task. As described in Section I, this smooth trajectory will be obtained from an iterative procedure that first generates an angular trajectory, based on cell decomposition, than smoothens it and tests it against intersections with obstacles. Before describing the iterative procedure (subsection III.C), we will first focus on constructing an angular path (subsection III.A) and a smooth path (subsection III.B).

A. Angular path

by performing a cell decomposition of the free space from environment E (as mentioned in Section II), and we assume that a graph $G = (C, A)$ is obtained. For simplicity, we assume that all arcs have unitary cost, although some approaches might compute weights by considering the size of the car-like robot, or the energy spent for moving between adjacent cells. Next, we identify the cells including the start and the final positions, and we denote them by C_0 and C_f, respectively.

By performing a search on graph G, with start node C_0 and goal node C_f, we obtain a path showing the succession of cellular regions the robot should follow. Since we consider unitary costs on arcs of G, if the path was found by a minimum-cost search algorithm (e.g. Dijkstra), it includes the minimum number of cells. We simply construct the angular path (P_{ang}) by choosing a single point from each cell from the path in G, and by linking these points in the order the corresponding cells should be traversed. The point chosen from each cell can be either the centroid of the cell (e.g., for polytopal decompositions), or the centroid of the common line segment shared by the current cell with the next one, excepting, of course, the last cell from the path (e.g. in the case of trapezoidal and rectangular decompositions). Since we are dealing with convex cells, this choice of intermediary points guarantees that the angular path is contained in the sequence of regions defined by the cells from the path.

We finish the construction of the angular path by adding the starting point (x_s, y_s) at the beginning of the path, and the ending point (x_f, y_f) at the end. We denote the points defining the angular path P_{ang} by $(p_1, p_2, ..., p_N)$, where $p_i = (x_i, y_i) \in \square^2$, $i = 1,...,N$, and $p_1 = (x_s, y_s)$, $p_N = (x_f, y_f)$, $p_1, p_2 \in C_0$, $p_{N-1}, p_N \in C_f$. The approach we use for constructing P_{ang} is common in navigation planning that uses cell decompositions, and we refer the interested reader to [2] for a more detailed treatment of the trapezoidal-cell case.

B. Smooth path

As stated before, P_{ang} is the union of the line segments $[p_i, p_{i+1}]$, $i = 1,...,N-1$, and can be described in Cartesian coordinates of the fixed frame S by:

$$P_{ang} = \bigcup_{i=1}^{N-1} \{(x, y) \in [x_i, x_{i+1}] \times [y_i, y_{i+1}] \,|\, a_i x + b_i = y\} \qquad (2)$$

P_{ang} is in general a non-smooth trajectory, because the line segments defining it have might have different slopes. Since the car-like described by (1) can only follow smooth trajectories, we aim to generate a smooth path P_{smooth} by starting from P_{ang}. The idea of generating P_{smooth} resembles Dubins' curves [12], which interconnect circle arcs and line segments such that a desired position and orientation are reached. Our approach differs in the following sense: we already have an angular path that gives us the orientation of the intermediary line segments, and we smoothen it by using circle arcs with different radii.

In the following, we denote the orientation (slope) of each segment $[p_i, p_{i+1}]$, $i = 1,...,N-1$, by θ_i, where

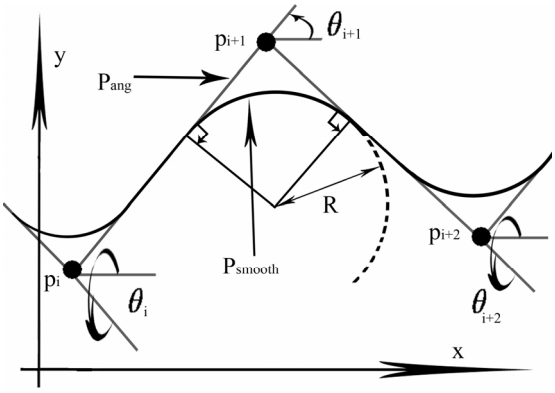


Fig. 2. P_{ang} and P_{smooth} described in section III A and III B

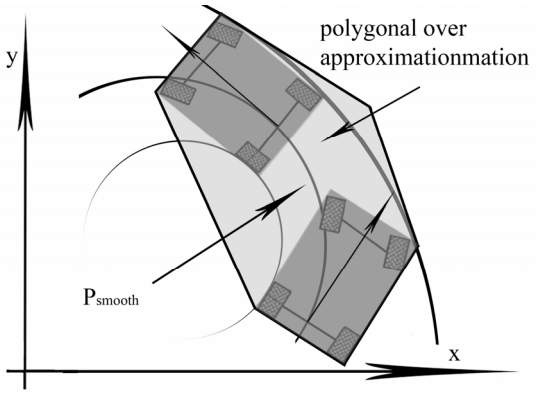


Fig. 3. Polygonal over approximation of the area swept by the robot when following a circle arc

$\theta_i = atan(a_i)$. Let us assume that at a certain time moment the robot is along a line segment $[p_i, p_{i+1}]$ (its reference point is on $[p_i, p_{i+1}]$, and its orientation is θ_i). For being able to move the robot such that it reaches any position from $[p_{i+1}, p_{i+2}]$ and orientation θ_{i+1}, we interconnect the line segments $[p_i, p_{i+1}]$ and $[p_{i+1}, p_{i+2}]$ by an arc of circle of radius *R*, tangent to both segments, as illustrated by Fig. 2. In choosing the value of *R* we make a tradeoff between the following two aspects: on one hand, the minimum value of *R* is imposed by the maximum steering angle $\phi_{\max}$, and such a minimum radius would keep P_{smooth} as close as possible to P_{ang}. On the other hand, the smaller the radius is, the larger the discontinuity will be in control input ϕ, when steering the robot from a line segment to the arc of circle.

P_{smooth} is generated by iterating the above idea for all successive line segments from P_{ang}. Due to space constraints, we do not include the algorithm used for generating P_{smooth}. However, we mention that for some P_{ang} we cannot generate P_{smooth}, simply because the minimum radius imposed by $\phi_{\max}$ might not be small enough to interconnect short line segments. In such a case, the algorithm used for generating P_{smooth} returns a set *I* of pairs of indices for line segments from P_{ang} that cannot be smoothly interconnected.

C. Feasible path

We say that an obtained P_{smooth} is *feasible* if no obstacle is hit while the robot moves along it. In order to check the feasibility of P_{smooth} found as in subsection III.B, we first consider that the initial orientation of the robot chassis is chosen to be θ_1. Then, while the robot moves forward along the part of the line segment $[p_1, p_2]$ included in P_{smooth} it sweeps a rectangle with a side equal to *l*. Checking the possible intersections of this rectangle with obstacles O_1, …, O_n reduces to checking the existence of a solution for a linear programming (LP) problem, where the constrained set is given by the polytopal obstacles and the rectangle swept by the robot, and a trivial cost is assumed. For checking possible intersections with obstacles while the robot follows a circle arc from P_{smooth}, we over approximate the area swept by the robot with a polyhedron, as shown in Fig. 3, and then we solve a similar LP problem. Although this over approximation increases the conservativeness of our approach, solving an LP is computationally attractive than numerically solving a system of nonlinear equations with constrained solutions. The over approximation of the curved swept area is similar with the one used in [19], the difference consisting in the construction and the shape of the considered polyhedron.

The feasibility of P_{smooth} is checked by iterating the above approach for all its line segments and circle arcs. When some elements of P_{smooth} yield intersections with obstacles, their corresponding pairs of indices for line segments from P_{ang} are stored in a set *I*, and the current path is unfeasible.

So far we have described all the individual procedures for constructing P_{ang}, P_{smooth} and for testing if P_{smooth} is correct from the point of view of not yielding collisions with obstacles. All these strategies are incorporated in Algorithm 1, which ends by either returning a correct path, or by not finding any feasible one, case when we deem Problem 1 unfeasible.

Algorithm 1 – *Find a feasible path*
Perform cell decomposition and obtain $G = (C, A)$
WHILE TRUE
  Find P_{ang} as described in subsection III.A
  IF P_{ang} not found
    break /* Problem 1 is unfeasible */
  ENDIF
  Construct P_{smooth} from P_{ang}
  Check feasibility of P_{smooth}
  IF P_{smooth} was not created or is not feasible $A_{i,j} = 0,\ A_{j,i} = 0$ for any $\{i, j\} \in I$
  ELSE
    break/* P_{smooth} is a feasible trajectory*/
  ENDIF
ENDWHILE

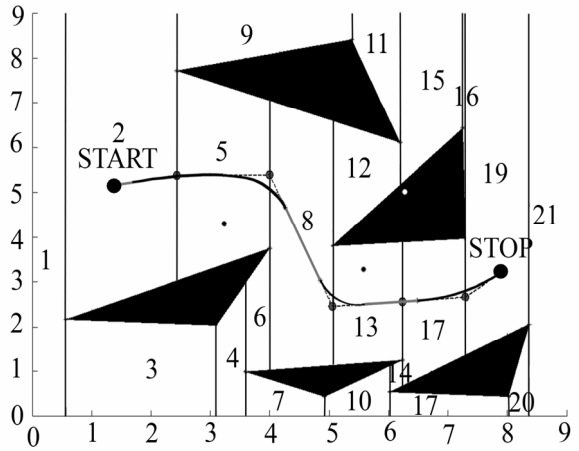


Fig. 4. P_{ang} and P_{smooth} obtained with trapezoidal cell decomposition

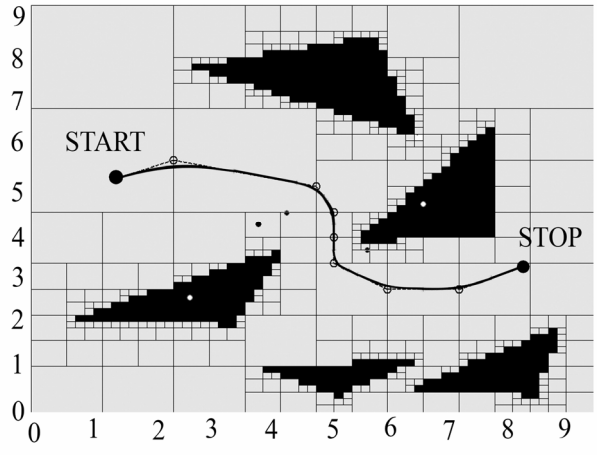


Fig. 5. P_{ang} and P_{smooth} obtained with quad-tree cell decomposition

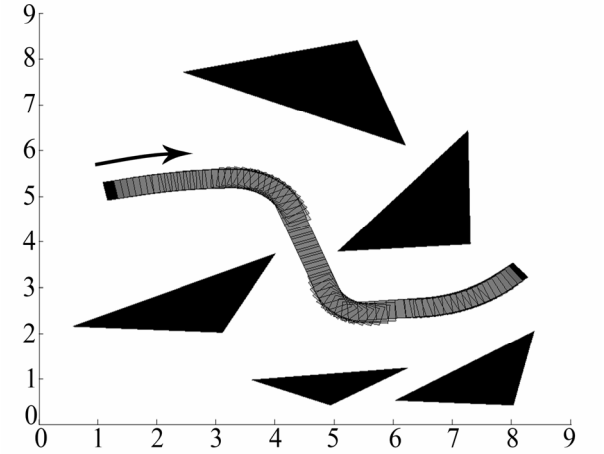


Fig. 6. Robot motion in the environment when the trapezoidal cell decomposition is used

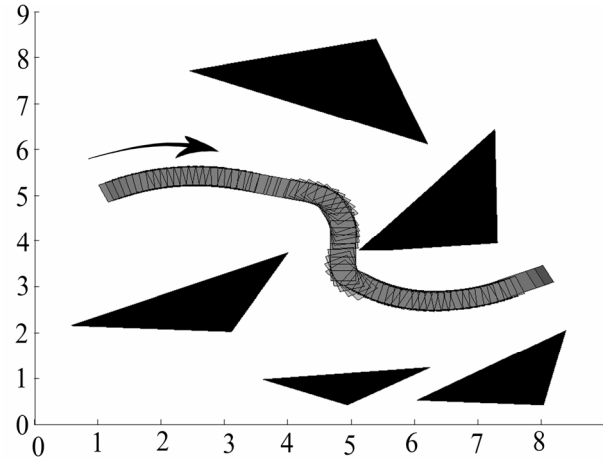


Fig. 7. Robot motion in the environment when the quad-tree cell decomposition is used

Algorithm 1 iteratively finds an angular path, a smooth path and tests the feasibility of the smooth path. If the current path is feasible, it will be used as reference trajectory in Section IV. Otherwise, links from set *I* (that prevented the construction of P_{smooth}, or that induced unfeasibility of the current path) are removed from graph *G*, and a new angular path is searched. Since we remove an arc from the graph *G* at each iterative step, Algorithm 1 is guaranteed to terminate in a finite number of steps (if a feasible path cannot be found, at some point P_{ang} does not exist and the procedure stops).

IV. Control Strategy

The control strategy is based on a small variation linear model, obtained from the linearization of nonlinear system (1) around a generic point (x, y, θ) of the path P_{smooth} (meaning the two coordinates, supplemented with information about angle). The generic points are found by assuming a constant velocity *v* when moving along P_{smooth}, and the angle is given by the tangent in the current coordinates at P_{smooth}. The small variation linear model is further discretized by using a sampling period chosen in accordance with value of *v*. The feedback law providing the control input ϕ is designed by assigning eigenvalues able to ensure a reasonable decrease of the state variables (i.e., the variations from the nominal point). At each sampling time, the nominal point is considered as the space origin for the discrete-time form of the linearized model.

V. Simulation Results

To illustrate the proposed planning strategy, we developed a simulation study in Matlab. The example we focused on is depicted in Fig. 4, where a trapezoidal decomposition is performed. We have chosen this decomposition after a number of tests for trapezoidal, rectangular, triangular and polytopal cells. The advantage offered by the trapezoidal decomposition consists in the algorithm's simplicity and the reduced number of resulting cells. For example, Fig. 5 represents a comparison term corresponding to the decomposition of the same environment by using rectangular cells resulted from a quad-tree approach. As a consequence, P_{ang} and P_{smooth} shown in Fig. 4 allow a robust implementation for the path following strategy. The advantage of the trapezoidal cells commented above does not reflect a general recipe, since for other environments, one can obtain trapezoids that are too thin.

The generated P_{smooth} was used in the control strategy briefly described by Section IV, and the trace of the car-like robot motion is illustrated in Fig. 6 - for the trapezoidal decomposition and, respectively, in Fig. 7 – for the quad-tree approach. A comparative analysis of figures 6 and 7 points out a supplementary advantage (not commented above) of the path generated with trapezoidal cell decomposition. Generally speaking, by considering this path and any two neighboring obstacles with the path located in between, the points of the path are placed at approximately equal distances from the two obstacles. Thus, adequate lateral margins are ensured for the motion of the robot (whose width cannot be neglected). This property is ensured by the construction procedure selecting the mid points of the edges, and it may not hold true for the quad-tree cell decomposition. Figures 6 and 7 exhibit visible differences for the robot motion in the descending part of the two sigmoid-type "tubes" that keep the moving robot inside. In fig. 7 the motion is very close to an obstacle (meaning an increased risk of collision), whereas in fig.6 the motion progresses at reasonable distances from all the obstacles (with low risk of collision). Actually the differences between figures 6 and 7 are predictable by examining figures 4 and 5, since the angular path in fig. 5 is not equally distanced from the adjacent obstacles.

VI. Conclusion

This paper proposed a solution for planning and controlling a car-like robot in planar environments cluttered with obstacles. The novelty refers to the development of an iterative procedure that returns a smooth path based on cell decomposition methods and connections inspired by Dubins' curves. Although the approach we propose is not complete, since a solution might not be found even if one exists, it has the advantage of low computational complexity and easy implementation.

The proposed solution deserves further investigations in extending the problem to more complex, human-like robot specifications.

Acknowledgements

The first author acknowledges the support of the EURODOC (Doctoral Scholarships for research performance at European level) project, financed by the European Social Fund and the Romanian Government. The second author acknowledges the support of the CNCSIS-UEFISCSU project PN-II-RU PD code 333/2010.

Both authors are indebted to Professor Octavian Pastravanu for useful discussions on various topics of this paper.

References

[1] Latombe, J., *Robot Motion Planning*, Kluwer Academic Publishers, 1991

[2] Choset, H., Lynch, K. M., Hutchinson, S., Kantor, G., Burgard, W., Kavraki, L. E., and Thrun, S. *Principles of Robot Motion: Theory, Algorithms, and Implementations*, 2005, MITPress, Boston.

[3] La Valle, S. M. *Planning algorithms* Cambridge University Press, Cambridge, UK, 2006.

[4] C. Belta, A. Bicchi, M. Egerstedt, E. Frazzoli, E. Klavins, and G. J. Pappas, "Symbolic Planning and Control of Robot Motion," *IEEE Robotics and Automation Magazine - special issue on Grand Challenges of Robotics*, vol. 14, no. 1, pp. 61-71, 2007

[5] Kloetzer, M. and Belta, C., "A fully automated framework for control of linear systems from temporal logic specifications," *IEEE Transactions on Automatic Control*, 2008b, 53(1): pp. 287–297.

[6] Fainekos, G. E., Kress-Gazit, H., and Pappas, G. J., "Hybrid controllers for path planning: a temporal logic approach, " *In IEEE Conference on Decision And Control*, pages4885–4890, Seville, Spain, 2005

[7] Conner, D. C., Rizzi, A. A., and Choset, H., "Integrated Planning and Control for Convex–bodied Nonholonomic systems using Local Feedback Control Policies," *In: Robotics: Science and Systems II*, Philadelphia, PA, 2006

[8] David C. Conner, Hadas Kress-Gazit, Howie Choset, Alfred A. Rizzi, and George J. Pappas, "Valet Parking Without a Valet," *Proc. of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, CA, USA, Oct 29 - Nov 2, 2007 pp 572-577.

[9] B. Siciliano, O. Khatib – *Springer Handbook of Robotics*, Springer-Verlag, 2008, pp. 9-62, 799-527.

[10] Kloetzer, M. and Belta, C., "A framework for automatic deployment of robots in 2d and 3d environments," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp 953–958, 2006, Beijing, China.

[11] Belta, C., Isler, V., and Pappas, G. J., "Discrete abstractions for robot planning and control in polygonal environments, " *IEEE Transactions on Robotics*, 2005, 21(5),pp 864-874

[12] L. E. Dubins, "On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents," Amer. J. Math., vol. 79, pp. 497–516, 1957.

[13] A. Scheuer and Ch. Laugier, "Planning Sub-Optimal and Continuous-Curvature Paths for Car-Like Robots," *IEEE – RSJ Int. Conf. on Intelligent Robots and Systems*, Victoria BC (CA), October, 1998.

[14] Tzu-Chen Liang, Jing-Sin Liu, Gau-Tin Hung, Yau-Zen Chang, "Practical and flexible path planning for car-like mobile, robot using maximal-curvature cubic spiral," Robotics and Autonomous Systems, vol. 52, pp. 312–335, June 2005, Available: www.directscience.com.

[15] F. Lamiraux and J. -P. Lamond, "Smooth motion planning for Car-Like Vehicles," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 4, August 2001, pp. 498-502.

[16] J. A. Reeds and R.A.Shepp, "Optimal paths for a car that goes both forward and backward," Pacific J. Math., vol. 145, no. 2, pp. 367–393, 1990

[17] Shahram Rezaei, Jose Guivant, Eduardo M. Nebo, "Car-Like Robot Path Following in Large Unstructured Environments," , *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pt. 3, pp. 2468-73, v 3, 2003.

[18] A. Yahja, S. Singth and A. Stentz, Efficient, " On-Line Path Planner for Outdoor Mobile Robots, " *Robotics and Autonomous Systems*, 32(2): pp. 129-143, 2000.

[19] A. Scheuer and Th. Fraichard, "Continuous-Curvature Path Planning for Multiple Car-Like Vehicles," *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. September,pp. 8-12, Grenoble, France, 1997.

[20] Tua Agustinus Tamba, Bonghee Hong, and Keum-Shik Hong, "A Path Following Control of an Unmanned Autonomous Forklift," *International Journal of Control, Automation, and Systems*, Springer, 2009, Available : http://www.springer.com/12555.

[21] J. P. Laumond *Robot Motion Planning and Control*, Springer-Verlag, Toulouse, pp171- 253, 1998.

[22] Patricia Mellodge, "Feedback Control for a Path Following Robotic Car," M.S. thesis, Faculty of the Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 2002.

[23] T. D. Gillespie - *Fundamentals of Vehicle Dynamics* - Society of Automotive Engineers, 1992.

[24] DannyZ. Chen, Robert J. Szczerba, and JohnJ. Uhran, Jr., "A Framed-Quadtree Approach for Determining Euclidean Shortest Paths in a 2-D Environment, " *IEEE Transactions on Robotics and Automation,* Vol. 13, no.5, pp 668-681, 1997

Real-Time Rendering and Animation of Vegetation

Ralf Habel
Institute of Computer Graphics and Algorithms
Vienna University of Technology

Abstract— **Vegetation in all its different forms is almost always part of a scenery, be it fully natural or urban. Even in completely cultivated areas or indoor scenes, though not very dominant, potted plants or alley trees and patches of grass are usually part of a surrounding. Rendering and animating vegetation is substantially different from rendering and animating geometry with less geometric complexity such as houses, manufactured products or other objects consisting of largely connected surfaces.**

In this paper we will discuss several challenges posed by vegetation in real-time applications such as computer games and virtual reality applications and show efficient solutions to the problems.

I. Introduction

Vegetation in computer graphics can be roughly categorized into the field of modeling the growth of a plant by generating its geometry, and the field of modelling the appearance and behavior of plants in an environment. Though real plants all basically use the same processes to grow, a plethora of methods can be applied to generate plants at various ages, ranging from fractals [1], L-Systems [2] and procedural approaches [3] to full simulations of ecosystems [4], among others. To display and to animate this generated geometry interactively, specialized representations, lighting and shading techniques together with animation or simulation methods are applied to incorporate the non-geometric attributes of vegetation. Of course, both fields are strongly connected since the environment impacts the growth of a plant [5]. Also, geometric representation and lighting or shading techniques are heavily dependent on each other since geometric attributes need to be transported by the representation in order to have them available for shading.

Though the generation of plants has received more attention than other aspects of vegetation, only the combination of accurate geometry, appearance and dynamic behavior results in a convincing result. Especially under real-time conditions, all facets of displaying vegetation pose significant problems, which makes interactive rendering and animation of vegetation one of the biggest challenges in real-time graphics.

II. Challenges

The term vegetation is a broad term, covering structures such as lawns up to complete landscapes covered with a forest. Many computer games and virtual reality applications are already very realistic, though most lack a realistic display of plants and trees due to their inherent complexity. Especially for trees and grass, many standard acceleration and simplification methods cannot be applied. This results in severe compromises in the realism of their appearance compared to other parts of a scene. There are several reasons why vegetation is more difficult to display than other objects:

A. Light Interaction

Vegetation is not only complex in geometry, also the light interaction of leaves or grass blades is highly intricate. A leaf for example usually consists of different layers and is strongly structured, which has a profound impact on both the reflectance and translucency of leaves, an integral part of the light interaction of vegetation. Additionally, many leaves differ not only between species but also in their light transport on the front and back, depending on the nature of the surface, and no general assumptions can be made.

B. Geometric Complexity

Concerning trees and treelike plants, it is possible to use a full geometry representation on current hardware, though only a limited amount of polygons can be spent on each branch and leaf depending on the corresponding size and shape, also limiting the number of branches to a few thousand and the number of leaves to a few ten thousand.

A tree in full geometry representation poses challenges to create realistic animations under the given real-time constraint since every branch and leaf is perceived as a separate part and thus needs to be treated separately. The structure of a tree consists of a complex hierarchy of branches to which leaves are attached, all of which interact with a turbulent wind field, and every part of the tree must react consistently to wind in order to achieve a realistic and convincing animation of a complete tree.

III. Leaf Rendering

The rendering of leaves in commercial applications such as games and virtual reality simulations is usually avoided completely by representing trees as billboards or billboard clouds. This means that there are no separate leaves and the textures used for the billboard are rendered with standard methods, not taking care of special attributes of leaves. This can already produce somewhat good results if the textures are generated so they benefit the appearance of leaves [6]. But this approach does not reproduce the behavior of leaves in light and can therefore provide only very limited realistic results.

Leaves have a very complex interaction with light and only few assumptions can be made since there is a large

variety of leaves. They differ not only in shape and color, but also in surface attributes, ranging from highly glossy surfaces due to thick wax layers to completely diffuse surfaces due to micro hairs. Also, leaves usually show very different light interactions on the adaxial and abaxial side. But the most defining attribute of leaves differentiating them from other surfaces is their translucency, which becomes very apparent in direct sunlight when seen from the unlit side (see Figure 1).

Fig. 1. Leaves in sunlight.

Another research area where light-leaf interaction is important is remote sensing, which is usually done by satellite or radar. In order to derive values such as vegetation covering of a landscape, health of plants, water containment of plants, etc., from measurements, accurate models of reflectance, translucency and general light transport inside plants or canopies are required to extrapolate such data. Though those models are targeted to derive biophysical and agricultural properties, they can also be applied to computer graphics. An extensive overview of optical properties in the context of remote sensing can be found in [7].

A realistic leaf can not be modeled using standard methods due to the intricate light-leaf interaction, and specialized methods have to be applied to render convincing vegetation since an important part of the appearance is dominated by the scattering of light inside a leaf. Real-time graphics only tries to model the appearance of objects so fully accurate models that predict the light transport are not required and using measured data to reproduce the appearance without an exact knowledge of the internals is sufficient to display highly realistic results.

Scattering of light is a wide field in computer graphics, ranging from scattering in gaseous structures such as clouds or fog to scattering in fluid and solid material such as milk, marble, skin or leaves. In fluid and solid materials which reside inside a non-scattering medium, usually air, the scattering can be described with a BSSRDF [8]. Compared to a BSDF, the incident light can be at a different position than the exitant light, making a BSSRDF an 8-dimensional non-local function. This high dimensionality poses a computational problem which can only be solved exactly by path tracing. Practical methods reduce the dimensionality, compromising on the accuracy of the solution or deriving analytical expressions for special cases.

Concerning real-time rendering, subsurface scattering is an active research area with many results. Examples are skin subsurface scattering [9], scattering in more general lighting conditions [10] or deformable models [11]. Although this field can be seen as a complete sub-area of real-time rendering, only a few publications propose techniques that specifically deal with realistic leaf rendering.

Many properties of a leaf such as local thickness, optical density or internal structure have an essential impact on its appearance. These values are usually not generated synthetically but measured, so data sets have to be created that a model can be fitted or verified to.

A. Measurements

A realistic result can be achieved by measuring, since nature contains many small imperfections which are automatically captured. In this case, the surface of a leaf is fully reproduced so all structures on its surface, including any bending and bulging are incorporated, resulting in a very realistic appearance. The acquisition setup allows generating high-resolution maps (smaller than $1mm$) using a very simple process and off-the-shelf scanning hardware, so even the smallest details, which have an essential impact on both the reflectance and translucence, are captured.

The devices used are a 3D scanner operating at an effective resolution of 0.1 mm (Minolta VI-910), a digital camera (Canon EOS 20D) with fixed exposure time, two 1,000 Watt light sources with large box diffusers, and an easy to construct fixing frame for the leaf. The large diffusers are used to approximate hemispherical illumination, which is required for capturing the albedo, removing any directionality in the illumination.

The leaf is sampled by first taking a 3D scan, then the scanner is replaced with a diffuser and the albedo is recorded using the camera. To capture the translucency, the front diffuser is switched off and a picture is taken with the back diffuser on. The same procedure is done for the back side of the leaf.

For postprocessing, standard tools are applied. Maya was used to create a simplified mesh and to generate highly detailed normal maps [12] and displacement maps for both sides. Further, the thickness map is generated by subtracting the displacement maps, normalized to a user-defined maximum thickness which can also be measured directly on the leaf. The normal maps are not bound to a specific geometry but can be mapped to different geometric levels of detail (see Figure 2). Figure 3 shows a complete data set generated using this measurement method. In comparison to the acquisition setup by Wang et al. [13], the per-pixel BRDF or BTDF data is not captured, requiring a custom-built linear light source device in order to measure both sides. Spatially varying roughness or specular intensity according to measurements cannot be encoded, though hand-produced modulations of BRDF and BSDF parameters are still possible. On the other

Fig. 2. The scanned geometry, normal-mapped simplified geometry and the normal map on a quad patch. The highlights have been exaggerated for visualization purposes.

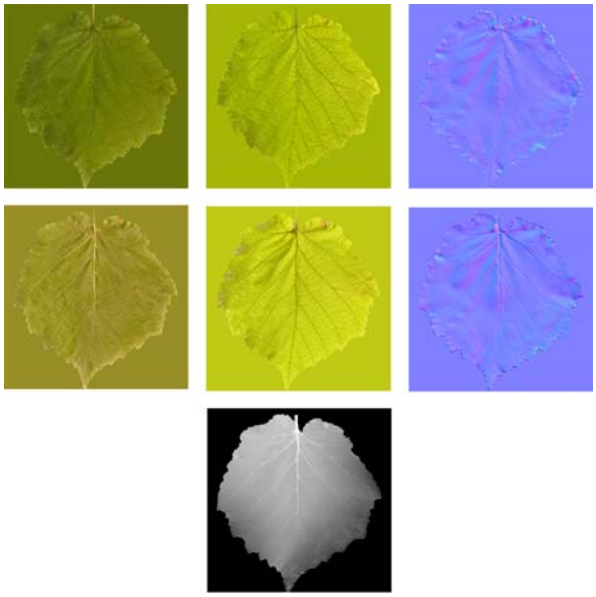


Fig. 3. A complete data set of a leaf, consisting of albedo (left), translucency (middle) and normal map (right) for both sides and a thickness map (bottom).

hand, high-resolution normal maps are created, which causes highlights to be placed more accurately according to the high-frequency structure of the leaf. This is not the case for the method proposed by Wang et al. [13].

B. Reflectance

The structure of a leaf is mostly perceived in its specular reflectance properties due to direct sunlight illumination, revealing its high-frequency structures, which are the most prominent features in the illumination and need to be modeled correctly to achieve a realistic leaf rendering. There is a huge variety of leaf BRDFs, ranging from velvet-like due to micro-hairs to highly specular caused by a thick waxy layer. In most cases, the front of a leaf has broad specularity whereas the back of a leaf is diffuse.

Following Bousquet et al. [14] and Wang et al. [13], the Cook-Torrance shading model [15] is used for the front side of the leaf. A simple diffuse model is applied to the back side of the shown leaves. This is not a general limitation, the reflectance attributes of each measured leaf should be examined to match them for a faithful reproduction. Other BRDFs such as Blinn [16] or Schlick [17] are also good choices depending on the physical accuracy required or other factors such as editability or parameter tuning.

As for the parameters of the Cook-Torrance BRDF, the measured and fitted specular coefficients of Bousquet et al. [14] are used. Their measurements define a range of $n = 1.2 - 1.7$ for the effective refractive index and $\sigma = 0.078 - 0.5$ for the roughness, covering highly specular leaves (e.g. *Laurel*) to nearly diffuse specular lobes (e.g. *Hazel*). Figure 4 shows leaves with different parameter configuration.

Fig. 4. Quad patches shaded with highly specular (top) and almost diffuse (bottom) reflectance, and with a directional light at steep (left) and grazing angle (right).

C. Translucency

One of the main insights of the shown technique is that while subsurface scattering has only negligible impact on the appearance of the light-facing side of a leaf, it is the dominant factor for the opposite side. Figure 5 demonstrates the difference between a simple, yet state-of-the-art translucency model based on a diffuse BTDF, and a BSSRDF approach. As opposed to the reflective part of the leaf, where high-frequency features are conveniently modeled using a normal map, the same approach should be taken to model the high-frequency surface variations in the translucent part compared to a simple diffuse shading model using the geometric normal of the leaf. By including these variations, depending on the incident light angle, the leaf appears either smooth at steep angles or shows the influence of the high-frequency details from bulges and veins at grazing light angles (see Figure 5).

The main features taken into account by the BSSRDF model are self shadowing of the leaf before the light pen-

Fig. 5. Physically based leaf translucency (top) with light at different angles from steep (left) to grazing angles (right) in comparison to the standard diffuse translucency model (bottom).

etrates into the leaf interior, variations in leaf thickness, and variations of the reflectance properties over the light-facing leaf surface. These effects lead to variations in the amount of light entering the medium and scattering towards a specific point to be shaded on the opposite leaf surface. Note that the presented model is local to a leaf, and therefore light variations due to shadows from other leaves or similar only modify the resulting radiance, but do not enter into the subsurface scattering computations. These effects can be handled using standard real-time shadow algorithms.

A combination of both reflectance and translucency, showing the consistency of both can bee seen in Figure 6.

Fig. 6. Leaf on a tree showing fully consistent reflectance and translucency at grazing light directions.

IV. Tree Animation

Generally, animating a tree involves a number of components. First, a wind model describes the characteristics of the wind-tree interaction which is coupled with a dynamic system of some form that describes the reaction of branches and leaves to the applied wind force. Usually, the dynamic system incorporates a structural model to define the hierarchical organization. Structural elements then define how the results of the dynamic system affect the geometry of the tree, the bending of branches for example.

The prevalent structural element model for trees is a skinned or rigid skeletal joint system analogous to rigid or smooth skinning of characters([18], [19], [20], [21], [22], [23]). Many interactive methods simply avoid the problem of bending by not considering any form of deformation ([20], [21]). To achieve deformation, a skeletal structure is used to segment single branches in order to model the smooth bending of a branch. With segmented branches, a principal problem occurs since a high number of joints are needed to get convincing results. Additionally, if leaves are represented separately, each leaf requires its own joint.

Another approach to model the deformation of branches is to use a structural mechanics model ([24], [25], [26]). Structural mechanics is the computation of deformations, deflections, and internal forces or stresses within structures, either for design or for performance evaluation of existing structures. As this is the basis of many engineering sciences, a number of different and highly developed methods exist, though the simpler methods are sufficient to accurately model a branch, as the goal is to achieve a correct appearance and thus strong simplifications can be introduced without compromising the realistic appearance of a deformation.

A. Wind Model

To avoid a full fluid dynamics simulation, a stochastic approach from wind engineering can be applied as proposed by Shinya et al. [25] which is also used by Zhang et al. [19]. Instead of solving the Navier-Stokes equation, wind is modeled as a velocity field with longitudinal, lateral and vertical components. Each fluctuating component of the resulting velocity vectors is modeled by a stationary Gaussian stochastic process. The spatial-temporal properties of the components in the frequency domain are represented by the Cross-Power Spectral Density Matrix to model the coherence of the fluctuations where the FFT (Fast Fourier Transform) delivers the velocity field. Since the fluid simulation is replaced by an FFT, this approach is much faster and still can deliver realistic and physically based wind fields, because turbulent wind can be modeled accurately through a stochastic process. Stam [24] applies a similar model by filtering uncorrelated random velocity vectors for each branch in the frequency domain to achieve a correlation of loads on nearby branches.

B. Heuristic Animation

Heuristic models do not try to solve the animation problem by means of a simulation, but rather try to emulate the appearance of vegetation movement as efficiently as possible ([21], [27]) using noise functions to drive the animation.

This usually does not require dedicated structural elements, structural models or elaborate calculations but still can deliver acceptable animation. The simplest approach is to modulate the position of a vertex by a noise function, disregarding any correlation to the geometric structure. Additional weights, which define the strength and direction of

the displacement, can introduce user-defined constraints and a high controllability, which makes this approach very artist friendly and can also emulate the bending of branches on a coarse level [27].

The key element for real-time performance is to *localize* all computations in a vertex shader, leading to so-called vertex displacement. This is often used to animate simple vegetation represented as billboards (e.g., grass) or billboard clouds (e.g. simple tree models). For full geometry models, this is not straight-forward.

The general approach is to expose all relevant information of the tree structure, i.e. the hierarchy of the structural model, to every vertex. Thus, the hierarchical deformations of all parent branches can be explicitly performed inside the vertex shader and no information needs to be propagated at runtime. This can be achieved by assigning each vertex an index into a texture that holds all necessary information. This means that every vertex within a branch has the same index. Additionally, as sub-branches can emanate at any position on its parent branch, the relations of each vertex to all parent hierarchy levels are required. To have this data exposed to each vertex, the normalized local coordinate $x \in [0..1]$ of the vertex is precalculated as a scalar per vertex, where x is along the principal axis of a branch. Also, the x-values at parent-branch connections are calculated and propagated down the branch hierarchy (see Figure 7). These values

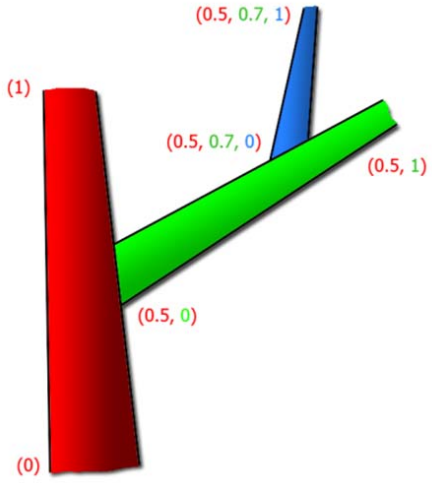


Fig. 7. $\vec{w}$ distribution of branches in a tree.

are stored with each vertex in addition to its own x-value. The shown trees have 4 hierarchy levels, so each vertex has a vector $\vec{w}$ of 4 values associated in addition to the branch index. A problem that occurs with hierarchical vertex displacement is that the used deformation model that defines the structural elements needs to be able to correctly transform the local coordinate axes between hierarchy levels, so tangent and normal transformations need to be available in order to transform local coordinate systems as well.

V. Beam Model

The model used for describing the geometry and physics of a branch as a beam determines how realistic branches swaying in the wind appear to the viewer. A common approximation for realistic animation systems is to model the beam as an elastic cylinder (uniform beam) using structural mechanics, and describe the deformation due to a uniform traversal force using a polynomial deflection function depending on the basic physical properties of the beam. However, uniform beams are not a good approximation for tree branches, as branches are not uniform beams but thin out (taper) at their free end, which has an essential impact on the bending behavior. This taper leads to the effect that tips are much more flexible than thicker parts, which is not accounted for in a uniform beam model. Also, the length needs to be taken care of to achieve a convincing deformation.

VI. Synthesizing Branch Motion

A tree interacting with wind is a highly complex dynamic system that is difficult to solve through numerical simulation while upholding the restrictions of hierarchical vertex displacement, not allowing access to previous states. To simplify the dynamic system, it is treated as an uncoupled system of harmonic oscillators per branch. The basis of this simplification is that an observer cannot judge the correctness of the response function of the highly complex dynamical system because the wind itself is not visible. Thus, the characteristics of the animation are mainly determined by the frequencies and amplitudes of branches and not by their exact functional values.

To avoid explicit integration of the equations of motion, *spectral methods* similar to [26] are used to synthesize branch motion to drive the branch deformation. The principal idea is to generate noise functions that obey the same frequency distributions as empirically observed data or results of a full simulation.

VII. Conclusion

The presented method allows efficiently animating and rendering highly detailed trees with a massive amount of branches and leaves in high quality. With the stochastic approach, there are no considerable costs and enough resources remain free for other calculations such as shading the tree or other parts of a natural scene.

The shown set of methods are confined to a vertex shader using hierarchical vertex displacement, leveraging the performance of GPUs and also making it easy to integrate into existing frameworks.

In summary, the presented methods provide a simple and efficient way for high quality animation. There are no elaborate precomputations required and all parameters can be changed interactively, both on a high level such as the damping coefficient of a level, and on a very low level such as the physical properties of an individual branch.

VIII. Acknowledgements

This research was funded by the Austrian Science Fund (FWF) under contract no. P21130-N13.

Fig. 8. Detailed animated tree.

REFERENCES

[1] P. E. Oppenheimer, "Real time design and animation of fractal plants and trees," in *SIGGRAPH '86: Proceedings of the 13th annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM, 1986, pp. 55–64.

[2] P. Prusinkiewicz, A. Lindenmayer, and J. Hanan, "Development models of herbaceous plants for computer imagery purposes," *SIGGRAPH Comput. Graph.*, vol. 22, no. 4, pp. 141–150, 1988.

[3] P. de Reffye, C. Edelin, J. Françon, M. Jaeger, and C. Puech, "Plant models faithful to botanical structure and development," in *SIGGRAPH '88: Proceedings of the 15th annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM, 1988, pp. 151–158.

[4] O. Deussen, P. Hanrahan, B. Lintermann, R. Měch, M. Pharr, and P. Prusinkiewicz, "Realistic modeling and rendering of plant ecosystems," in *SIGGRAPH '98: Proceedings of the 25th annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM, 1998, pp. 275–286.

[5] W. V. Haevre, F. D. Fiore, P. Bekaert, and F. V. Reeth, "A ray density estimation approach to take into account environment illumination in plant growth simulation," in *SCCG '04: Proceedings of the 20th spring conference on Computer graphics*. New York, NY, USA: ACM, 2004, pp. 121–131.

[6] A. Kharlamov, "Next-generation speedtree rendering," in *GPU Gems 3*, H. Nguyen, Ed. Addison Wesley, July 2007, ch. 4.

[7] S. L. U. Stephane Jacquemoud, "Leaf optical properties: A state of the art." in *8th Int. Symp. Physical Measurements & Signatures in Remote Sensing*, 2001, pp. 223–232.

[8] F. E. Nicodemus, J. C. Richmond, J. J. Hsia, I. W. Ginsberg, and T. Limperis, *Geometrical considerations and nomenclature for reflectance*. USA: Jones and Bartlett Publishers, Inc., 1977.

[9] D. L. Eugene d'Eon, "Advanced techniques for realistic real-time skin rendering," in *GPU Gems 3*, H. Nguyen, Ed. Addison Wesley, July 2007, ch. 14.

[10] R. Wang, J. Tran, and D. Luebke, "All-frequency interactive relighting of translucent objects with single and multiple scattering," in *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers*. New York, NY, USA: ACM Press, 2005, pp. 1202–1207.

[11] T. Mertens, J. Kautz, P. Bekaert, F. V. Reeth, and H.-P. Seidel, "Efficient rendering of local subsurface scattering," in *PG '03: Proceedings of the 11th Pacific Conference on Computer Graphics and Applications*. Washington, DC, USA: IEEE Computer Society, 2003, p. 51.

[12] J. Cohen, M. Olano, and D. Manocha, "Appearance-preserving simplification," in *SIGGRAPH '98: Proceedings of the 25th annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM Press, 1998, pp. 115–122.

[13] L. Wang, W. Wang, J. Dorsey, X. Yang, B. Guo, and H.-Y. Shum, "Real-time rendering of plant leaves," in *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers*. New York, NY, USA: ACM Press, 2005, pp. 712–719.

[14] L. Bousquet, S. Lacherade, S. Jacquemoud, and I. Moya, "Leaf BRDF measurements and model for specular and diffuse components differentiation," *Remote Sensing of Environment*, vol. 98, pp. 201–211, 2005.

[15] R. L. Cook and K. E. Torrance, "A reflectance model for computer graphics," *ACM Trans. Graph.*, vol. 1, no. 1, pp. 7–24, 1982.

[16] J. F. Blinn, "Models of light reflection for computer synthesized pictures," *SIGGRAPH Comput. Graph.*, vol. 11, no. 2, pp. 192–198, 1977.

[17] C. Schlick, "An inexpensive brdf model for physically-based rendering," *Computer Graphics Forum*, vol. 13, pp. 233–246, 1994.

[18] Y. Akagi and K. Kitajima, "Computer animation of swaying trees based on physical simulation." *Computers and Graphics*, vol. 30, no. 4, pp. 529–539, 2006.

[19] L. Zhang, C. Song, Q. Tan, W. Chen, and Q. Peng, "Quasi-physical simulation of large-scale dynamic forest scenes," in *Computer Graphics International*, 2006, pp. 735–742.

[20] T. Sakaguchi and J. Ohya, "Modeling and animation of botanical trees for interactive virtual environments," in *VRST '99: Proceedings of the ACM symposium on Virtual reality software and technology*. New York, NY, USA: ACM, 1999, pp. 139–146.

[21] O. Shin, T. Fujimoto, M. Tamura, K. Muraoka, K. Fujita, and N. Chiba, "1/fβ noise-based real-time animation of trees swaying in wind fields," in *Computer Graphics International*, 2003, pp. 52–59.

[22] R. Zioma, "Gpu-generated procedural wind animations for trees," in *GPU Gems 3*, H. Nguyen, Ed. Addison Wesley, July 2007, ch. 6.

[23] F. D. F. William Van Haevre and F. V. Reeth, "Physically-based driven tree animations," *Eurographics Workshop on Natural Phenomena*, pp. 75–82, 2006.

[24] J. Stam, "Stochastic dynamics: Simulating the effects of turbulence on flexible structures," *Computer Graphics Forum*, vol. 16, no. 3, pp. C159–C164, 1997. [Online]. Available: citeseer.ist.psu.edu/stam97stochastic.html

[25] M. Shinya and A. Fournier, "Stochastic motion-motion under the influence of wind," *Comput. Graph. Forum*, vol. 11, no. 3, pp. 119–128, 1992.

[26] Y.-Y. Chuang, D. B. Goldman, K. C. Zheng, B. Curless, D. H. Salesin, and R. Szeliski, "Animating pictures with stochastic motion textures," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 853–860, 2005.

[27] T. Sousa, "Vegetation procedural animation and shading in crysis," in *GPU Gems 3*, H. Nguyen, Ed. Addison Wesley, July 2007, ch. 16.

Single volume reconstruction from multiple MRI images

Paul Herghelegiu, Vasile Manta

Abstract— **Magnetic Resonance Imaging (MRI) is a medical imaging technique used especially for the visualization of soft tissues of the human body. For a better understanding of a particular organ, applications that help physicians in the diagnostication process usually use a segmentation method to separate the organ, followed by a three-dimensional reconstruction process. In this paper, we present a new method for combining the medical data obtained along multiple scanning plans of the same patient using MRI data. Our method is based on the ray-casting algorithm and uses the features of graphical processing unit, for obtaining a real time result. By using this method, we avoid the heavy computational process of rotating, scaling and translating the three-dimensional textures used in the visualization process in order to align them at the same coordinate system.**

I. Introduction

MEDICAL Imaging is the domain of medicine that deals with creating images of the human body for their use in clinical purposes (diagnose or disease examination). Currently, the purpose of the images obtained from Magnetic Resonance Imaging (MRI) and Computed Tomography (CT) scans is no longer restricted to simple inspection of individual slices. The interest has expanded to the development of new applications that helps physicians in many clinical practices such as surgical planning, fracture or tumor detection.

MRI is a medical imaging technique used in radiology to visualize especially soft tissues. MRI scans are mainly used in neurological (brain, spinal cord), musculoskeletal, cardiovascular, and oncological (cancer) imaging. Unlike the CT scanning that can provide images only along the axial plane, an MRI scan can provide images in any plane. Most commonly, an MRI scans along three planes, sagittal, coronal and axial, resulting three sets of images. Physicians usually have to inspect images form all three planes in order to determine an accurate diagnosis. Because the visualization of a volume offers a better perspective of the human body, the images obtained from an MRI scan are frequently reconstructed into a 3D volume. Numerous methods of reconstructing successive slices obtained for an medical scan into a volume have been developed: a method of reconstructing the pancreas and its surrounding using a previous saved model[1], the 3D reconstruction of the individual phrenic nerve [2]. Techniques that combine data from multiple types of medical scans have also been used, and are referred to as *multimodal visualization techniques*. A such multimodal technique for planning neuro-surgical interventions have been developed by S. Reissenberg et al. [3]. A method that uses data from magnetic resonance angiography (MRA) and from classical MRI in order to obtain a better segmentation of the brain vessels have been presented by N. Passat et al. [4]. A combination of MRI and magnetoencephalography (MEG) data for the study of repetition suppression have been introduced by R. Vidyasagar [5]. Most recent approaches use the features of parallel computing offered by Graphics Processing Unit (GPU) programming, allowing real time visualization. This approach of using the GPU is becoming more commonly used in medical imaging area in order to better understand the human anatomy and for providing support for the educational side of medicine. A new technique for efficient, real time visualization of multimodal cardiac data using consumer graphics hardware combining data from CT, MRI and others scanning techniques have been presented by D. Levin et al. [6]. The programmability of graphic cards provides the background needed in order to merge and display multimodal data [8]. Hong et al. [9] used aligned 3D texture for multimodality rendering. Because an MRI scan provides three sets of images (one for each plane), after applying the reconstruction process three volumes are obtained.

In order to emphasize certain organs of the human body, a segmentation method that will separate the image into two distinct regions (one containing points of interest and one containing other points belonging to the image) had to be used. We focused upon the spinal cord. Until now, different unsupervised methods of image segmentation have been used [10]. In general, segmentation methods are based on local properties of images, such as discontinuities or similarities.

In this paper, we present a method of combining reconstructed volumes from sets of images obtained directly from an MRI scan (one for each plane) and from sets of corresponding segmented images, providing a unique volume. Our method is based on the ray-casting algorithm and uses the features of OpenGL Shading Language (GLSL) in order to obtain real time results. Using the information about the exact position of each MRI slice we have

Manuscript received May 15, 2010.

P. Herghelegiu is with the Faculty of Automatic Control and Computer Engineering, Technical University "Gheorghe Asachi" Iasi, Str. Prof. dr. doc. Dimitrie Mangeron, nr. 27, Iasi, cod 700050, Romania (e-mail: pherghelegiu@cs.tuiasi.ro).

V. Manta is with the Faculty of Automatic Control and Computer Engineering, Technical University "Gheorghe Asachi" Iasi, Str. Prof. dr. doc. Dimitrie Mangeron, nr. 27, Iasi, cod 700050, Romania (e-mail: vmanta@cs.tuiasi.ro).

computed the exact 3D position of each volume, and we have obtained a single comprehensible volume that includes data from all reconstructed volumes.

This paper is organized as follows. Chapter 2 describes the steps followed in order to obtain a single volume from multiple reconstructed volumes. Chapter 3 reports the conclusions of this paper.

II. NEW METHOD FOR OBTAINING A SINGLE VOLUME

A. Reconstructing MRI slices and corresponding segmented images

In order to separate the data corresponding to the spinal cord from the MRI images, we used Region Growing as a segmentation method, method that requires minor user intervention in establishing the seed points around which the region corresponding to the spinal cord will develop. To correct any holes that can appear in the segmented region after the process of segmentation, an iterative binary hole filling algorithm was applied. We used the Iterative Hole Filling Filter that is part of Insight Segmentation and Registration Toolkit (ITK). We applied this algorithm on every image obtained from an MRI scan, so we have obtained a set of segmented images for every set of slices obtained form the MRI scan.

We considered two sets of images form the same MRI scan: a set of images resulting from an axial scan, and the second one resulting from a sagittal scan. By applying the segmentation method we obtained the corresponding segmented images. After reconstructing and combining these sets of images, we obtained an unique volume. By applying this technique on both sets of data, two volumes with the segmented spinal cord have resulted, as can be seen in Figure 1.

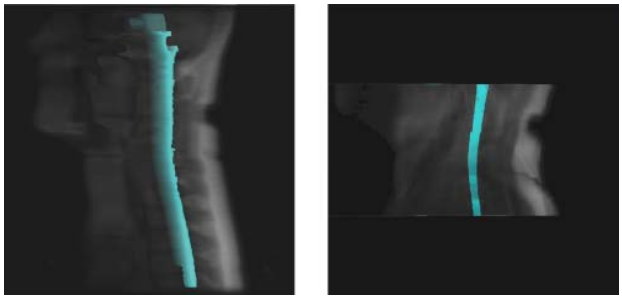
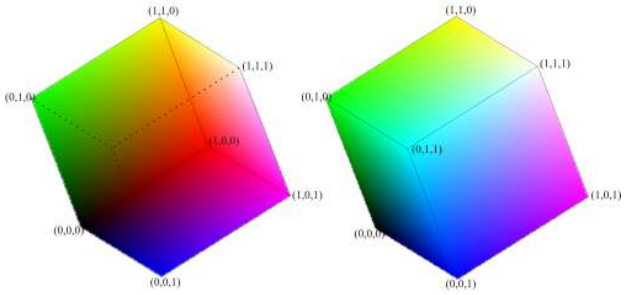


Figure 1. The reconstructed volumes from the sagittal plane images (on the left) and from the axial plane images (on the right)

B. GLSL method for obtaining a unique volume from multiple reconstructed volumes

Our approach for combining two volumes into a unique volume is based on the ray-casting algorithm which involves computing the color of every point belonging to the output image by casting a ray along the viewing direction into the volume that has to be displayed. Ray-casting algorithm uses an RGB cube for computing the direction of the ray into the volume. This is done by rendering a cube where the colors represent coordinates. We used the OpenGL feature to render the front or the back faces of an object. By subtracting the back-face from the front we obtain a direction vector for each pixel. This direction vector represents the direction of the ray. Both images with the front and back faces of the same RGB cube can be seen in Figure 2.



(a) (b)

Figure 2. Front (a) and back (b) faces of an RGB cube used for displaying a single 3D volume

We applied this algorithm for both volumes, resulting two ray directions which will be used in our algorithm for combining the volumes.

In order to align the coordinates of the second volume to the ones of the first volume we used the three basic transformations: scale, rotate, translate functions provided by OpenGL. We considered the coordinates of the first volume as main coordinates, and we applied transformations onto the second volume to align it to the main coordinates.

C. Combining multiple MRI data into a single volume

For obtaining a better perspective over one reconstructed volume, some information from the other volumes obtained after the MRI scan can be added. Figure 3 presents an area in the sagittal reconstructed volume that can be corrected by adding information from the other reconstructed segmented volumes. We obtain the presented unrealistic result because of the OpenGL interpolation process that is automatically applied between slices, in order to obtain the 3D texture used in the visualization process. By combining data from multiple unaligned scans into a single volume, the inspection of multiple volumes or slices can be avoided.

In order to combine the reconstructed volumes obtained from an MRI scan using the method presented above, we need to know the exact position of each slice. This position of each MRI slice is given by the coordinates data embedded in the image (pixel spacing, image orientation, image position, slice thickness). Pixel spacing provides the physical distance in the patient between the center of each pixel, specified by a numeric pair - adjacent row spacing (delimiter) and adjacent column spacing in millimeters;

image orientation specifies the direction cosines of the first row and the first column with respect to the patient; image position specifies the (x, y, z) coordinates of the upper left hand corner of the image; slice thickness will provide the normal slice thickness, in millimeters.

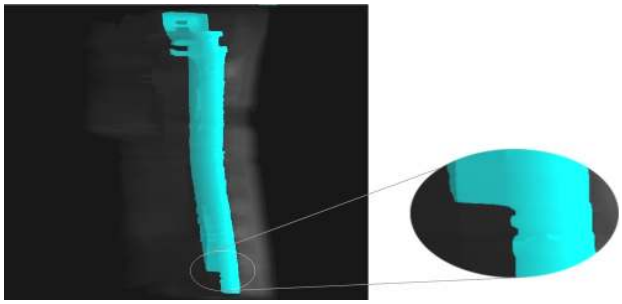


Figure 3. Section in the sagittal reconstructed volume that can be improved by adding data from other MRI scans.

We consider two successive slices used in the reconstruction process of the sagittal volume (named Sag_1, Sag_2), and two successive slices used for reconstructing the axial volume (named Ax_1, Ax_2). The data used from the MRI image used for establishing the position of each individual slice is presented in Table 1: (x, y, z) represents the upper left corner coordinates of the slice, $(\cos r_x, \cos r_y, \cos r_z)$ and $(\cos c_x, \cos c_y, \cos c_z)$ represents the direction cosines of the first row and respectively the first column of the slice.

TABLE 1
POSITION AND DIRECTION COSINES FOR TWO SAGITTAL SLICES AND FOR TWO AXIAL SLICES

Slice	Sag_1	Sag_2	Ax_1	Ax_2
x	-4.146341	0.353659	-109.646339	-109.646339
y	-120.0	-120.0	-98.079422	-98.079422
z	257.658539	257.658539	77.038040	81.486374
$\cos r_x$	0.0	0.0	1.0	1.0
$\cos r_y$	1.0	1.0	0.0	0.0
$\cos r_z$	0.0	0.0	0.0	0.0
$\cos c_x$	0.0	0.0	0.0	0.0
$\cos c_y$	0.0	0.0	0.988519	0.988519
$\cos c_z$	-1.0	-1.0	0.151096	0.151096

In order to align the two volumes to the same coordinate system, we have to apply a series of transformations on the axial volume. As can be observed in Figure 4, in order to align the system coordinate of the axial volume to the axes of the sagittal volume, two rotations are needed, by 90 degrees about the x axis, followed a rotation with -90 degrees about the z axis.

For implementing these rotations in OpenGL, a correspondence between the coordinate system used in OpenGL and the coordinate system of the patient has been defined as follows: $Ox_o \Leftrightarrow Oy_p$, $Oy_o \Leftrightarrow Oz_p$ and $Oz_o \Leftrightarrow Ox_p$.

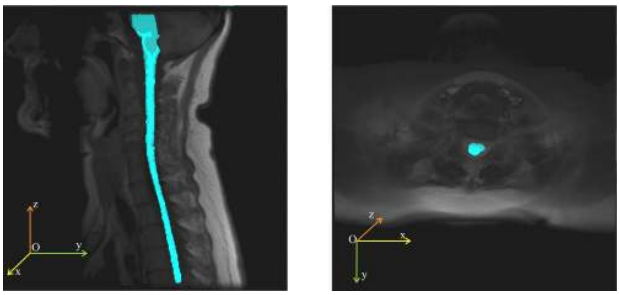


Figure 4. Patient coordinate axes for the reconstructed volumes

In the OpenGL coordinate system, the rotations needed for the alignment of the axial volume to the sagittal volume are:

- Rotation about the Oz_o axis with 90 degrees, described by the following rotation matrix:

$$R_{z_o} = \begin{pmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix};$$

- Rotation about the Oy_o axis with -90 degrees, described by the following rotation matrix:

$$R_{y_o} = \begin{pmatrix} 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

After aligning the two coordinate system, he have to apply another rotation for obtaining the direction given by the first column of the axial images. This rotation is about the Ox_o axis with an angle of $\arccos(0.988519) = 8.690482$ degrees. This angle is embedded in the MRI image as be seen in Table 1. The final position of the axial volume after applying all the rotations described above is presented in Figure 5.

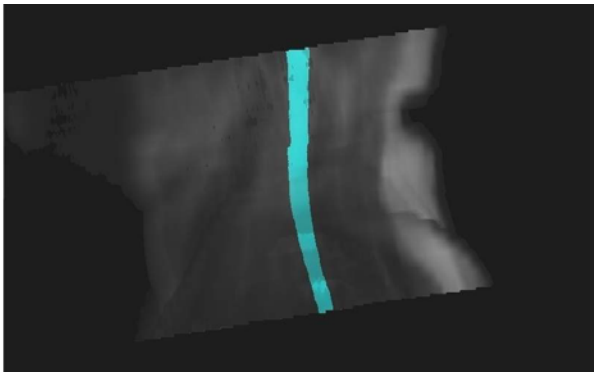


Figure 5. Reconstructed axial volume aligned to the corresponding position in the coordinate system of the patient

After applying this algorithm for aligning the two coordinate systems, we combine the two volumes, sagittal and axial, into one single volume. The unique volume obtained from the two reconstructed volumes is presented in Figure 6.

Figure 6. The final volume obtained from combining data from the sagittal and axial reconstructed volumes

III. CONCLUSION

In this paper we presented a method of combining data from multiple MRI scans. We used a semi-automated segmentation method for the delimitation of a specific organ of the human body (spinal cord) and we used a straight forward method for reconstructing a series of images into a 3D volume. Based on a ray-casting algorithm, we introduced a method for combining multiple volumes with different space orientation. For aligning the volumes at the same coordinate system, we implemented the three basic transformations (scale, rotate, translate) onto the RGB cubes used by the ray-casting algorithm. By applying this technique, we minimize the time consumed for computing the transformations on the 3D textures, textures that correspond to every reconstructed volume that have a contribution on the unique final volume. For the alignment of the two coordinate systems of the reconstructed volumes, we rotated the axial volume twice, once about the z axis and once about the y axis. The final direction of the axial volume was given by the direction cosines embedded in the MRI image.

REFERENCES

[1] Yun Jin et al, "Three-dimensional reconstruction next term of the pancreas and its surrounding structures," Computerized Medical Imaging and Graphics Volume 32, Issue 4, June 2008, pp 277-283

[2] Boris Schmidt MD et al. Three-dimensional reconstruction of the anatomic course of the right phrenic nerve in humans by pace mapping. Heart Rhythm Volume 5, Issue 8, August 2008, pp 1120-1126

[3] S. Reissberg et al. Multimodalnext term brain tumour diagnostics for the individual planning of neuro-surgical interventions in the area of eloquent brain areas – Diagnostic possibilities by means of modern previous termMRInext term devices (1.5 T) Clinical Neurophysiology Volume 120, Issue 1, January 2009, pp e71

[4] N. Passat et al, Watershed and multimodalnext term data for brain vessel segmentation: Application to the superior sagittal sinus. Image and Vision Computing Volume 25, Issue 4, April 2007, pp 512-521 International Symposium on Mathematical Morphology 2005

[5] Rishma Vidyasagar et al. A multimodalnext term brain imaging study of repetition suppression in the human visual cortex. NeuroImage Volume 49, Issue 2, 15 January 2010, pp 1612-1621

[6] Techniques for efficient, real-time, 3D visualization of multi-modality cardiac data using consumer graphics hardware Computerized Medical Imaging and Graphics, Volume 29, Issue 6, September 2005, pp 463-475 David Levin, Usaf Aladl, Guido Germano, Piotr Slomka

[7] Multimodal volume rendering with 3D textures Computers & Graphics, Volume 32, Issue 4, August 2008, pp 412-419 Pascual Abellán, Dani Tost

[8] Hardwiger M, Kniss M, Rezk-Salama K, Weiskopf D. In: Engel K, Peters AK, editors. Real-time volume graphics. 2006.

[9] Hong H, Bae J, Kye H, Shin YG. Efficient multimodalty volume fusion using graphics hardware. In: ICCS 2005, Lecture Notes in Computer Science, vol. 3516, 2005. pp. 842–45.

[10] Zhanga, H. et al, Image segmentation evaluation: A survey of unsupervised methods. Computer Vision and Image Understanding, Vol. 110, pp. 260–280, 2008

Linked Neighborhood Pattern-Sensitive Faults in Random-Access Memories. A Fault Coverage Evaluation

Cristina Huzum, and Petru Caşcaval

***Abstract*— A fault coverage evaluation regarding to a linked neighborhood pattern sensitive faults model (NPSFs) in $N \times 1$ random-access memories is presented. The most important published tests dedicated to the NPSF model have been considered in this simulation study. Even though these tests cover all the simple faults, simulation results show that only the longer of them are able to detect all linked NPSFs.**

***Key words*: Memory Testing, Functional Fault Model, Static Faults, Linked Neighborhood Pattern-Sensitive Faults.**

I. Introduction

RAPID increase of density in the integrated circuits has an immediate effect upon memory testing. On one hand, the capacity of random-access memories chips enhances, thus increasing the test time and cost; on the other hand, the density of memory circuits grows, therefore more failure modes and faults need to be taken into account in order to obtain a good quality product. Accordingly, there are two conflicting constraints that need to be dealt with when considering a test algorithm: reducing the number of memory operations in order to permit large capacity memories to be tested in an appropriate period of time and covering a larger variety of memory faults [1], [2].

As a result of the increasing coupling effect triggered by the growing density of memory circuits, the pattern-sensitive fault (PSF) is becoming an important fault model [3], [4], [5]. The PSF model is a type of coupling fault, with several aggressor cells (4, 9 etc.) and only one victim cell. In this work, the neighborhood PSF (NPSF) has been considered. This is a particular PSF, in which the aggressor cells are located in the physical neighborhood of the victim cell. The NPSF model was first defined by J.P. Hayes in 1980 [3]. He also devised a memory test for this model [3]. Soon after that, D.S. Suk and M. Reddy have proposed a new memory test [6] based on a bipartite method. This test divides the memory cells into two partitions and applies a sequence of transitions to cover all possible victim-aggressor combinations. Unfortunately, for the memory chips currently used, the test proposed by Suk and Reddy needs a long time to perform. In 2002, other more efficient march tests were given by Cheng, Tsai, and Wu, namely: CM-79N and March-100N [7]. In another paper, written in 2008, Julie, Wan Zuha and Sidek use a modified version of March-100N for diagnosis of SRAM [8]. In all these papers the authors have limited themselves only to the class of simple faults. In this work we focused on the problem of testing the linked neighborhood pattern-sensitive faults.

The remainder of this paper is organized as follows. Section II introduces some notations and definitions, and Section III defines the set of fault primitives for the neighborhood pattern-sensitive fault model. Section IV presents the memory tests we have considered for the simulation study. Section V presents experimental results regarding the ability of these important published tests dedicated to the NPSF model to cover linked NPSFs. Some conclusions concerning this work are drawn in Section VI.

Manuscript received: March 30, 2010.

Cristina Huzum is with the Faculty of Automatic Control and Computer Engineering , "Gheorghe Asachi" Technical University of Iaşi, Romania (e-mail: cristinahuzum@ yahoo.com).

Petru Caşcaval is with the Faculty of Automatic Control and Computer Engineering , "Gheorghe Asachi" Technical University of Iaşi, Romania (e-mail: cascaval@ cs.tuiasi.ro).

II. Notations, Definitions and Fault Classifications

An operation sequence that results in a difference between the observed and the expected memory behaviour is called a *sensitizing operation sequence* (*S*). The observed memory behaviour that deviates from the expected one is called *faulty behaviour* (*F*). In order to specify a certain fault, one has to specify the *S*, together with the corresponding faulty behaviour *F*, and the read result (*R*) of *S* in case it is a read operation. The combination of *S*, *F* and *R* for a given memory failure is called a *Fault Primitive* (FP), and is usually denoted as $< S / F / R >$. The concept of FPs allows for establishing a complete framework of all memory faults. Some classifications of FPs can be made based on different and independent factors of *S* [9].

a) Depending on the number of simultaneous operations required in the *S*, FPs are classified into *single-port* and *multi-port* faults.

- *Single-ports faults:* These are FPs that require at the most one port in order to sensitize a fault. Note that single-port faults can be sensitized in single-port as well as in multi-port memories.
- *Multi-port faults:* These are FPs that can only sensitize a fault by performing two or more operations simultaneously via different ports.

b) Depending on the number of simultaneous operations required in the *S*, FPs are classified into *static faults* and *dynamic faults*.

- *Static faults:* These are FPs which sensitize a fault by performing at most one operation in the memory (#O=0 or #O=1);

- *Dynamic faults*: These are FPs that perform more than one operation sequentially in order to sensitize a fault (#O > 1).

c) Depending on the way FPs manifest themselves, they can be divided into *simple faults* and *linked faults*.

- *Simple faults:* These are faults which cannot be influenced by another fault. That means that the behaviour of a simple fault cannot change the behaviour of another one; therefore masking cannot occur.
- *Linked faults:* These are faults that do influence the behaviour of each other. That means that the behaviour of a certain fault can change the behaviour of another one such that masking can occur. Note that linked faults consist of two or more simple faults.

In this work, single-port, static faults are considered. From here on, the term 'fault' refers to a single-port, static, simple fault and 'linked fault' means single-port, static, linked fault.

The following notations are usually used to describe operations on RAMs :

- ↑ denotes an up transition due to a certain sensitizing operation.
- ↓ denotes a down transition due to a certain sensitizing operation.

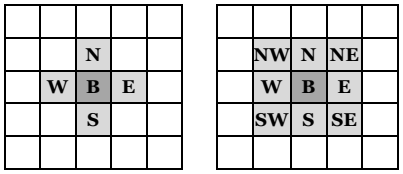
III. The Linked Neighborhood Pattern-Sensitive Fault Model

RAM faults can also be divided into *single-cell* and *multi-cell* faults. Single-cell faults consist of FPs involving a single cell, while multi-cell faults consists of FPs involving more than one cell. In this work, we consider a particular class of multi-cell faults (also called *coupling faults*), namely the pattern sensitive faults (PSF). The PSF is a coupling fault, which affects the content of a memory cell (called the *victim cell* or the *base cell*), or the ability to change its content, when other memory cells (called *aggressor cells*) have certain patterns. It is unnecessary and unrealistic to consider all possible patterns of all the memory cells, therefore simplified models of neighborhood pattern sensitive faults (NPSF) were introduced. In these models, the aggressor cells are limited to the physical neighborhood of the victim cell. Depending on the number of aggressor cells, NPSF can be divided into several types, but only two of those are more common: Type-1 NPSF that has four aggressor cells and Type-2 NPSF that has eight aggressor cells as illustrated in Fig. 1a and Fig. 1b, respectively [10].

Like in the most previous works, in this paper only Type-1 NPSF is studied, because it is more practical for the type of memory we have considered and less concerning when it comes to complexity.

Due to the features of the NPSF model, the general notation for a FP is particularized, thus in the rest of this paper a FP is denoted as (*N W E S*; *B* / *Bf*) [7], where:

- *N*, *W*, *E*, *S* describes the sensitizing value or operation in the aggressor cells (placed as presented in Fig. 1a);



a *b*

Fig. 1 – Common types of neighborhood pattern sensitive faults:
a – Type-1 NPSF; *b* – Type-2 NPSF.

- *B* describes the correct value or transition in the base cell;
- *Bf* shows the faulty value or transition of the base cell.

Note that *N*, *W*, *E*, *S*, *B* and $Bf \in \{0, 1, \uparrow, \downarrow\}$.

Depending on the behaviour of the fault, the NPSF can be divided into three classes [10], namely:

- Static NPSF (SNPSF): the base cell is forced to a certain value when the aggressor cells have a certain pattern. An example of a static NPSF is FP_1=<0100; 0/1>, where the base cell is forced to 1 when the aggressor cells have the pattern 0100.
- Passive NPSF (PNPSF) reflects the impossibility of the base cell to execute a transition due to the appearance of a certain pattern in the aggressor cells. An example of a PNPSFs is FP_2=<1100; ↓/1>, where the base cell cannot switch from 1 to 0 because the aggressor cells have the pattern 1100.
- Active NPSF (ANPSF): a certain transition in one of the aggressor cells forces the victim cell to change its state when the other aggressor cells (also called *enabling cells*) have a certain pattern. An example of this class of faults is FP_3=<10↑0; 0/1>, where a transition in the *E* cell causes the base cell to flip from 0 to 1 when the *N*, *W* and *S* cells have the pattern 100.

The model of NPSF we have considered can be entirely described by the set of FPs presented in Table I. There are 192 fault primitives: 32 SNPSFs, 32 PNPSFs, and 128 ANPSFs.

The linked neighborhood pattern-sensitive faults are NPSFs that influence the behaviour of each other, such as masking can occur. Therefore, they are more difficult to detect. A linked fault consists of two ore more FPs with contrary effects on the same victim (base) cell. For example, take a NPSF fault in which an up transition into cell *W* changes the state of cell *B* from 1 to 0, when the enabling cells have the pattern 100, whereas an up transition into cell *S* changes the state of cell *B* from 0 to 1, when the enabling cells have the pattern 011. This is a linked fault that can be modeled by two FPs,

FP_1=<1↑00;1/ 0>, and FP_2=<011↑;0/1>.

TABLE I
LIST OF NPSF PRIMITIVES

Fault primitives		Fault type
$<x\ y\ z\ t;\ 0\ /\ 1>$ $<x\ y\ z\ t;\ 1\ /\ 0>$	$x, y, z, t \in \{0, 1\}$	SNPSF
$<x\ y\ z\ t;\ \uparrow\ /\ 0>$ $<x\ y\ z\ t;\ \downarrow\ /\ 1>$	$x, y, z, t \in \{0, 1\}$	PNPSF
$<x\ y\ z\ \uparrow;\ 0\ /\ 1>$ $<x\ y\ z\ \downarrow;\ 0\ /\ 1>$ $<x\ y\ z\ \uparrow;\ 1\ /\ 0>$ $<x\ y\ z\ \downarrow;\ 1\ /\ 0>$ $<x\ y\ \uparrow\ z;\ 0\ /\ 1>$ $<x\ y\ \downarrow\ z;\ 0\ /\ 1>$ $<x\ y\ \uparrow\ z;\ 1\ /\ 0>$ $<x\ y\ \downarrow\ z;\ 1\ /\ 0>$ $<x\ \uparrow\ y\ z;\ 0\ /\ 1>$ $<x\ \downarrow\ y\ z;\ 0\ /\ 1>$ $<x\ \uparrow\ y\ z;\ 1\ /\ 0>$ $<x\ \downarrow\ y\ z;\ 1\ /\ 0>$ $<\uparrow\ x\ y\ z;\ 0\ /\ 1>$ $<\downarrow\ x\ y\ z;\ 0\ /\ 1>$ $<\uparrow\ x\ y\ z;\ 1\ /\ 0>$ $<\downarrow\ x\ y\ z;\ 1\ /\ 0>$	$x, y, z \in \{0, 1\}$	ANPSF

IV. MEMORY TESTS FOR NPSFS

In order to describe the memory tests we have considered for the simulation study, first some notations regarding the march tests are given. Usually, a complete march test is delimited by '{…}' bracket pair, while a march element is delimited by the '(…)' bracket pair. March elements are separated by semicolons, and the operations within a march element are separated by commas. Note that all operations of a march element are performed at a certain address, before proceeding to the next address. The whole memory is checked homogeneously in either one of two orders: ascending address order ($\Uparrow$) or descending address order ($\Downarrow$). When the address order is not relevant, the symbol $\Updownarrow$ is used. Multibackground march tests are march tests that run under several different data backgrounds [11], [12], [13]. In this case, the $w0$ and $r0$ operations are substituted with the *wa* and *ra* operations, respectively, where *a* is the value in the background. Also, $w1$ and $r1$ are substituted with *wb* and *rb*, respectively, where *b* is the complement of *a*.

The most important published test algorithms dedicated to the classical model of NPSF are presented as follows:

a) The test given by Suk and Reddy [6] (*SR* from here on) is a non-march test algorithm that divides the memory cells into two partitions and applies a sequence of transitions to cover all possible victim-aggressor combinations. The length of this test is 165*N*.

b) March-100N

This march memory test, given by Cheng, Tsai, and Wu [7], uses eight different data backgrounds and for each of them applies the following march sequence:

$\{\ \Updownarrow (wa); \Uparrow (ra, wb, wa); \Uparrow (ra, wb); \Uparrow (rb, wa, wb); \Uparrow (rb, wa); \Updownarrow (ra)\ \}$.

The data backgrounds used for the memory initialisation (denoted by BG1, BG2, ..., BG8) are presented in Fig. 2.

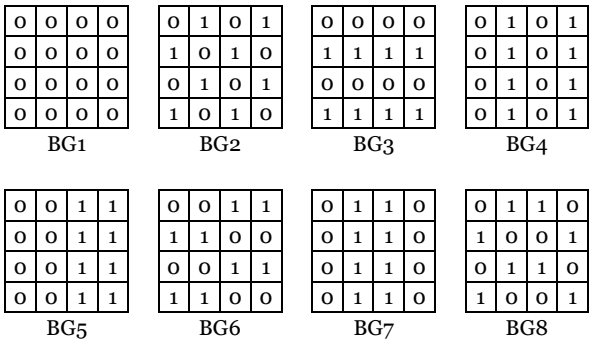


Fig. 2 – The 8 backgrounds for March-100N.

Additionally, only for BG1 the test applies the march sequence $\{\ \Downarrow (ra, wb); \Downarrow (rb, wa)\}$.

c) CM-79N

This memory test, also given by Cheng, Tsai, and Wu, [7], uses sixteen different data backgrounds and for each of them applies the following march sequence:

$\{\ \Updownarrow (wa); \Uparrow (ra, wb, rb, wa); \Updownarrow (ra)\ \}$.

The data backgrounds used for the memory initialisation (denoted by BG1, BG2, ..., BG16) are presented in Fig. 3.

As specified in [7], some redundant operations can be removed out of the test. Thus, the initial write operation of the march test is applied only on the cells that must be changed (note that every two successive backgrounds have exactly four different cells, so instead of writing nine cells, the test will write only the four cells that are different). Also, the last read operation skips over the cells that are not changed, because the first read operation for the next

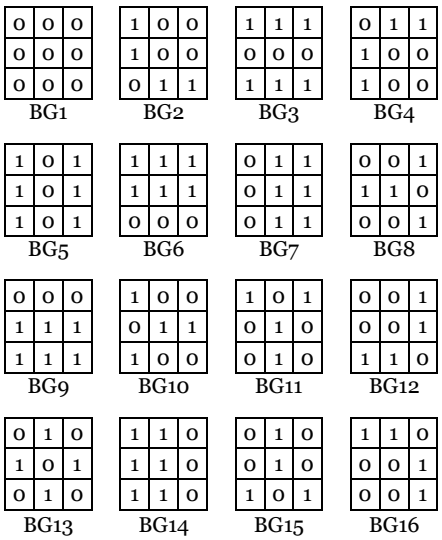


Fig. 3 – The 16 backgrounds for CM-96N march test.

background can do that. This happens with every background change. Therefore, the test length ($96N$) is reduced with $15 \times (5N/9 + 5N/9) = 50N/3$ operations. Consequently, the length of test CM-79N is $79\frac{1}{3}N$.

V. Fault coverage evaluation for linked NPSFs

For the simulation study, the linked faults consisting of two simple faults have been considered. There are 96 NPSFs that flip the base cell from 0 to 1 and 96 that flip it from 1 to 0. Consequently, a total of $96 \times 96 = 9216$ linked faults have been considered for the coverage evaluation by the memory tests presented in the previous section.

The simulation results show that the tests *SR* and CM-100N are able to cover entirely the model of linked NPSFs. Concerning the CM-79N test, our study demonstrates that a lot of linked NPSFs can not be detected by this march memory test. The linked faults undetected by CM-79N are presented as follows.

Taking into account that the patterns of the backgrounds used by this test are composed of 3×3 cells, for the simulation study, the memory cells have been divided into nine mutually disjoint subclasses. These are denoted B1, B2, …, B9, depending on their location. Let r and c be the row address and the column address, respectively, of a memory cell. The cell belongs to a certain subclass according to the following formulas:

- B1 – c % 3 = 0 and r % 3 = 0
- B2 – c % 3 = 0 and r % 3 = 1
- B3 – c % 3 = 0 and r % 3 = 2
- B4 – c % 3 = 1 and r % 3 = 0
- B5 – c % 3 = 1 and r % 3 = 1
- B6 – c % 3 = 1 and r % 3 = 2
- B7 – c % 3 = 2 and r % 3 = 0
- B8 – c % 3 = 2 and r % 3 = 1
- B9 – c % 3 = 2 and r % 3 = 2.

For a memory array with 8 rows and 8 columns, these nine subclasses of cells are illustrated in Fig. 4.

Due to the shape and the dimensions of the CM-79N backgrounds, every memory cell that belongs to the same subclass will support the same opperations during the test. Moreover, if two base cells belong to the same subclass, their agressor cells will support the same initializations. Hence, for the simulation study only nine locations (one for each subclass) have been considered for the base cell.

Table II presents all the linked NPSFs that are not detected by the test CM-79N. To simplify the writing in Table II, the '<' and '>' symbols usually used to denote a fault primitive have been neglected.

	0	1	2	3	4	5	6	7
0	B1	B4	B7	B1	B4	B7	B1	B4
1	B2	B5	B8	B2	B5	B8	B2	B5
2	B3	B6	B9	B3	B6	B9	B3	B6
3	B1	B4	B7	B1	B4	B7	B1	B4
4	B2	B5	B8	B2	B5	B8	B2	B5
5	B3	B6	B9	B3	B6	B9	B3	B6
6	B1	B4	B7	B1	B4	B7	B1	B4
7	B2	B5	B8	B2	B5	B8	B2	B5

Fig. 4 – The cell subclasses for an 8×8 memory chip array.

TABLE II
List of undetected Linked NPSFs for CM-79N

Sub class	Undetected linked faults			
B1	↑010;1/0	↑111;1/0	↑011;0/1	↑110;0/1
	↓010;0/1	↓111;0/1	↓011;1/0	↓110;1/0
	0↑01;1/0	0↑10;1/0	1↑00;1/0	1↑11;1/0
	0↓01;0/1	0↓10;0/1	1↓00;0/1	1↓11;0/1
	0↑00;0/1	0↑11;0/1	1↑01;0/1	1↑10;0/1
	0↓00;1/0	0↓11;1/0	1↓01;1/0	1↓10;1/0
	10↑0;1/0	01↑1;0/1	10↑1;0/1	11↑0;0/1
	10↓0;0/1	01↓1;1/0	10↓1;1/0	11↓0;1/0
	00↑1;1/0	010↑;1/0	100↑;1/0	111↑;1/0
	00↓1;0/1	010↓;0/1	100↓;0/1	111↓;0/1
	000↑;0/1	011↑;0/1	101↑;0/1	110↑;0/1
	000↓;1/0	011↓;1/0	101↓;1/0	110↓;1/0
	001↑;1/0			
	001↓;0/1			
B2	↑000;0/1	↑101;0/1	↑110;0/1	↑111;1/0
	↓000;1/0	↓101;1/0	↓110;1/0	↓111;0/1
	0↑00;0/1	0↑11;0/1	1↑01;0/1	1↑10;0/1
	0↓00;1/0	0↓11;1/0	1↓01;1/0	1↓10;1/0
	0↑01;1/0	1↑00;1/0	00↑0;0/1	10↑0;0/1
	0↓01;0/1	1↓00;0/1	00↓0;1/0	01↓0;1/0
	10↑0;0/1	10↑1;0/1	01↑1;0/1	11↑0;0/1
	010↑;1/0	10↓1;1/0	01↓1;1/0	11↓0;1/0
	00↑1;1/0	01↓0;1/0	000↑;0/1	101↑;0/1
	00↓1;0/1	101↑;0/1	000↓;1/0	010↑;1/0
	101↑;0/1	110↑;0/1	011↑;0/1	100↑;1/0
	101↓;1/0	110↓;1/0	011↓;1/0	100↓;0/1
	001↑;1/0	010↑;1/0	111↑;1/0	
	001↓;0/1	010↓;0/1	111↓;0/1	
B3	↑101;0/1	↑010;1/0	↑100;1/0	0↑00;0/1
	↓101;1/0	↓010;0/1	↓100;0/1	0↓00;1/0
	0↑11;0/1	1↑01;0/1	1↑10;0/1	0↑01;1/0
	0↓11;1/0	1↓01;1/0	1↓10;1/0	0↓01;0/1
	0↑10;1/0	1↑11;1/0	00↑1;1/0	10↑0;1/0
	0↓10;0/1	1↓11;0/1	00↓1;0/1	10↓0;0/1
	01↑0;1/0	01↑1;1/0	01↑1;1/0	11↑1;1/0
	01↓0;0/1	10↓1;0/1	100↓;0/1	11↓1;0/1
	00↓0;1/0	10↓1;0/1	000↓;0/1	000↑;0/1
	111↑;1/0	011↓;1/0	111↑;1/0	000↓;1/0
	101↑;0/1	110↑;0/1	011↑;0/1	100↑;1/0
	101↓;1/0	110↓;1/0	011↓;1/0	100↓;0/1
	001↑;1/0	010↑;1/0	111↑;1/0	100↓;0/1
	001↓;0/1	010↓;0/1	111↓;0/1	011↓;1/0

TABLE II (CONTINUED)
LIST OF UNDETECTED LINKED NPSFS FOR CM-79N

Sub class	Undetected linked faults			
B4	↑010;1/0	↑100;1/0	0↑00;0/1	0↑11;0/1
	↓010;0/1	↓100;0/1	0↓00;1/0	0↓11;1/0
	1↑01;0/1	1↑10;0/1	0↑10;1/0	00↑0;0/1
	1↓01;1/0	1↓10;1/0	0↓10;0/1	00↓0;1/0
	10↑1;0/1	01↑1;0/1	11↑0;0/1	00↑1;1/0
	10↓1;1/0	01↓1;1/0	11↓0;1/0	00↓1;0/1
	10↑0;1/0	01↑0;1/0	11↑1;1/0	000↑;0/1
	10↓0;0/1	01↓0;0/1	11↓1;0/1	000↓;1/0
	101↑;0/1	110↑;0/1	011↑;0/1	100↑;1/0
	101↓;1/0	110↓;1/0	011↓;1/0	100↓;0/1
	001↑;1/0	010↑;1/0	111↑;1/0	
	001↓;0/1	010↓;0/1	111↓;0/1	
B5	↑000;0/1	↑110;0/1	↑001;1/0	↑010;1/0
	↓000;1/0	↓110;1/0	↓001;0/1	↓010;0/1
	↑111;1/0	0↑01;1/0	0↑10;1/0	1↑00;1/0
	↓111;0/1	0↓01;0/1	0↓10;0/1	1↓00;0/1
	1↑11;1/0	10↑1;0/1	11↑0;0/1	00↑1;1/0
	1↓11;0/1	10↓1;1/0	11↓0;1/0	00↓1;0/1
	10↑0;1/0	11↑1;1/0	000↑;0/1	101↑;0/1
	10↓0;0/1	11↓1;0/1	000↓;1/0	101↓;1/0
	110↑;0/1	011↑;0/1	100↑;1/0	001↑;1/0
	110↓;1/0	011↓;1/0	100↓;0/1	001↓;0/1
	010↑;1/0	111↑;1/0	100↓;0/1	
	010↓;0/1	111↓;0/1	110↓;1/0	
B6	↑011;0/1	↑101;0/1	↑010;1/0	↑100;1/0
	↓011;1/0	↓101;1/0	↓010;0/1	↓100;0/1
	0↑11;0/1	1↑01;0/1	1↑10;0/1	0↑01;1/0
	0↓11;1/0	1↓01;1/0	1↓10;1/0	0↓01;0/1
	1↑00;1/0	1↑11;1/0	00↑0;0/1	10↑1;0/1
	1↓00;0/1	1↓11;0/1	00↓0;1/0	10↓1;1/0
	11↑0;0/1	00↑1;1/0	01↑0;1/0	11↑1;1/0
	11↓0;1/0	00↓1;0/1	01↓0;0/1	11↓1;0/1
	000↑;0/1	101↑;0/1	110↑;0/1	011↑;0/1
	000↓;1/0	101↓;1/0	110↓;1/0	011↓;1/0
	100↑;1/0	001↑;1/0	010↑;1/0	111↑;1/0
	100↓;0/1	001↓;0/1	010↓;0/1	111↓;0/1
B7	↑101;0/1	↑010;1/0	0↑00;0/1	0↑11;0/1
	↓101;1/0	↓010;0/1	0↓00;1/0	0↓11;1/0
	1↑10;0/1	0↑01;1/0	0↑10;1/0	1↑00;1/0
	1↓10;1/0	0↓01;0/1	0↓10;0/1	1↓00;0/1
	1↑11;1/0	10↑1;0/1	00↑1;1/0	01↑0;1/0
	1↓11;0/1	10↓1;1/0	00↓1;0/1	01↓0;0/1
	11↑1;1/0	01↓0;0/1	000↑;0/1	101↑;0/1
	11↓1;0/1	11↓0;1/0	000↓;1/0	101↓;1/0
	110↑;0/1	011↑;0/1	100↑;1/0	001↑;1/0
	110↓;1/0	011↓;1/0	100↓;0/1	001↓;0/1
	010↑;1/0	111↑;1/0		
	010↓;0/1	111↓;0/1		
B8	↑101;0/1	↑110;0/1	↑001;1/0	↑010;1/0
	↓101;1/0	↓110;1/0	↓001;0/1	↓010;0/1
	↑100;1/0	↑111;1/0	0↑00;0/1	0↑11;0/1
	↓100;0/1	↓111;0/1	0↓00;1/0	0↓11;1/0
	1↑01;0/1	1↑10;0/1	0↑01;1/0	0↑10;1/0
	1↓01;1/0	1↓10;1/0	0↓01;0/1	0↓10;0/1
	1↑00;1/0	00↑1;1/0	10↑0;1/0	11↑1;1/0
	1↓00;0/1	00↓1;0/1	10↓0;0/1	11↓1;0/1
	000↑;0/1	101↑;0/1	110↑;0/1	011↑;0/1
	000↓;1/0	101↓;1/0	110↓;1/0	011↓;1/0
	100↑;1/0	001↑;1/0	010↑;1/0	111↑;1/0
	100↓;0/1	001↓;0/1	010↓;0/1	111↓;0/1
B9	↑000;0/1	↑011;0/1	↑110;0/1	↑111;1/0
	↓000;1/0	↓011;1/0	↓110;1/0	↓111;0/1
	0↑00;0/1	0↑11;0/1	1↑01;0/1	1↑10;0/1
	0↓00;1/0	0↓11;1/0	1↓01;1/0	1↓10;1/0
	0↑01;1/0	0↑10;1/0	1↑00;1/0	1↑11;1/0
	0↓01;0/1	0↓10;0/1	1↓00;0/1	1↓11;0/1
	00↑0;0/1	10↑1;0/1	01↑1;0/1	11↑0;0/1
	00↓0;1/0	10↓1;1/0	01↓1;1/0	11↓0;1/0
	00↑1;1/0	10↑0;1/0	000↑;0/1	101↑;0/1
	00↓1;0/1	10↓0;0/1	000↓;1/0	101↓;1/0
	110↑;0/1	011↑;0/1	100↑;1/0	001↑;1/0
	110↓;1/0	011↓;1/0	100↓;0/1	001↓;0/1
	010↑;1/0	111↑;1/0		
	010↓;0/1	111↓;0/1		

VI. CONCLUSIONS

Many of the works dedicated to the NPSF model take into account only the simple fault model. However, a more realistic model that needs to be considered is the linked NPSF. This paper presents the fault coverage of the linked faults by the most important published tests dedicated to the NPSF model. The results show that only the longer algorithms are able to cover the whole model of linked NPSFs. Also, the simulation study leads to a new opportunity: to create a new memory test, shorter than March-100N, able to detect all NPSFs as well as linked NPSFs. This memory test will be the subject of an upcoming paper.

REFERENCES

[1] R. D. Adams, "High performance memory testing: design principles, fault modelling and self-test", *Kluwer Academic Publishers*, Norwell, USA, 2003.

[2] S. Hamdioui, "Testing static random access memories: defects, fault models and test patterns", *Kluwer Academic Publishers*, Norwell, USA, 2004.

[3] J. P. Hayes, "Testing memories for single-cell pattern-sensitive fault", *IEEE Trans. Comput.*, vol. 29, pp. 249-254, 1980.

[4] D. C. Kang, S. B. Cho, "An efficient build-in self-test algorithm for neighborhood pattern sensitive faults in high-density memories", in *Proc. 4th Korea-Russia Int. Symp. Science and Technology*, vol. 2, pp. 218-223, 2000.

[5] B. F. Cockburn, "Deterministic tests for detecting scrambled pattern-sensitive faults in RAMs", *Proc. IEEE Int. Workshop Memory Technology, Design and Testing (MTDT)*, San Jose, CA, pp. 117-122, 1995.

[6] D. S. Suk ,M. Reddy, "Test procedures for a class of pattern-sensitive faults in semiconductor random-access memories", *IEEE Trans. Comput.*, vol. 29, pp. 419–429, 1980.

[7] K. L. Cheng, M. F. Tsai, C. W. Wu, "Neighborhood pattern-sensitive fault testing and diagnostics for random-access memories", *IEEE Trans. On CAD*, vol. 21, no. 11, pp. 1328-1336, 2002.

[8] R. R. Julie, W. H. Wan Zuha, R. M. Sidek, "12N test procedure for NPSF testing and diagnosis for SRAMs", *Proc. IEEE Int. Conf. on Semiconductor Electronics*, pp. 430-435, 2008.

[9] S. Hamdioui, A. J. van de Goor, M. Rodgers, "March SS: A test for all static simple RAM faults", *Proc. of IEEE Int'l Workshop on Memory Technology, Design and Testing*, Isle of Bendor, France, pp. 95-100, 2002.

[10] A. J. van de Goor, "Testing semiconductor memories: theory and practice, *Wiley*, Chichester, U.K., 1991.

[11] V. Yarmolik, Y. Klimets, S. Demidenko, "March PS(23N) test for DRAM pattern-sensitive faults", in *Proc. Seventh IEEE Asian Test Symp.(ATS)*, Singapore, pp. 354-357, Dec. 1998.

[12] K.L.Cheng, M. F. Tsai, C. W.Wu, "Efficient neighborhood pattern-sensitive fault test algorithms for semiconductor memories", in *Proc. IEEE VLSI Test Symp (VTS)*, Maria Del Rey, CA, pp. 225-237 Apr.2001.

[13] V. Yarmolik, I. Mrozek, "MultiBackground memory testing", in *Proc. MIXDES 14th International Conference*, Ciechocinek, Poland, June 2000.

Control of the microalgae photosynthetic growth in a torus photobioreactor

George Ifrim*, Marian Barbu*, Mariana Titica**, Lionel Boillereaux**, Sergiu Caraman*

***Abstract*—The microalgae have the ability to use CO_2 as carbon source and, together with the solar energy, to biosynthesize various components, generating O_2. They have a huge potential in various industrial applications such as the production of therapeutic and industrial metabolites, biofuels and environmental applications. The aim of this paper is to control the photosynthetic growth of the microalga *Chlamydomonas reinhardtii*, in a continuous torus shape photobioreactor. The control strategy was to maintain the biomass concentration constant into the photobioreactor using the dilution rate as control variable. The main disturbance was the incident light flux. Two control laws were designed and analyzed in simulation, to wit a linear control algorithm and a nonlinear one. The linear controller was synthesized in a PI structure which was further tested in simulation at variable setpoints and incident light fluxes. The nonlinear controller was conceived in order to minimize the inconveniences encountered at the PI controller. The simulation conditions were identical for both controller types.**

I. Introduction

The Earth's atmosphere before Life was rich in CO_2 and CH_4. In these hostile conditions the first microorganisms – cyanobacteria – emerged 3.5 billion years ago. For eons, they were the sole photosynthesizers which contributed with oxygen to the reducing and anaerobic atmosphere. The first eukaryote appeared only 1.5 – 2.2 billion years ago, so far no living organism ever grew larger than a single cell [1]. The first photosynthetic eukaryotes – microalgae – played an essential role in the formation of the actual breathable atmosphere. The microalge have the ability to use CO_2 as carbon source and, together with the solar energy, to biosynthesize various components for the cell, at the same time generating O_2 as a residue. The photosynthesis process underpins all these by starting with the tool for harvesting light, the chlorophyll.

The microalgae stirred up the interest of scientists and industrialists due to their huge potential. Species like *Arthrospira* and *Chlorella* are widely used in human and animal nutrition [2]. These organisms are also used in various applications such as the production of therapeutic and industrial metabolites as the long chain polyunsaturated fatty acids, pigments, polycarbohydrates, vitamins or various biological active compounds [3]. However, the microalgae can be also used in environmental applications due to their capability to fix the carbon dioxide and certain heavy metals during growth – wastewater treatment and the greenhouse gases reduction – and to produce energy without emission of greenhouse gases – biofuels production [4]. The researches for new technologies will allow us to use and produce a clean and renewable energy just like the one recently approached – hydrogen production with microalgae [5].

Most of the recent research in microalgal culturing has been carried out in photobioreactors with external light supplies, designed as either tubular reactors, flat panel reactors, or column reactors with large surface areas, short internal paths, and small dark zones [6].

The control of photosynthetic growth processes is generally difficult to realize because of the nonlinear and time-varying nature of the systems. The slow response and the lack of suitable on-line sensors able to read the most important state variables are also obstacles for an accurate control.

The aim of this paper is to synthesize and test in simulation two control algorithms based on the Fouchard *et al.*, model [7]. After a brief introduction, the second part describes the photobioreactor and the model used for this work. The third part is dedicated to results and discussion on the control algorithms followed by conclusions and references.

II. Photobioreactor Description and Modeling

The photobioreactor modeled in this work (Figure 1) has one enlightened surface of torus shape where the light falls perpendicularly. In this way the light per volume ratio is higher. The reactor is of only 4 cm wide giving thus a working volume of 1 L. The culture homogenization is provided through a marine impeller. The photobioreactor possess a complete loop of common sensors and automations for microalgae culture (pH, temperature, nutrients, dissolved O_2), and allows an accurate control of the injected and collected gas (O_2, H_2, N_2, CO_2). The batch and the continuous mode are both suitable for this type of reactor [8].

The microorganism modeled was the microalga *Chlamydomonas reinhardtii* (wt 137c strain from the Chlamydomonas Genetic Center, Duke University, Durham, USA) [7].

* "Dunărea de Jos" University of Galaţi, Domnească Street No. 47, 800008 – Galaţi ROMÂNIA, phone/fax: +40-2-36-47-09-05, e-mail: George.Ifrim@ugal.ro).

** CNRS, GEPEA, UMR-CNRS 6144, University Bd., CRTT-BP 406, 44602 Saint-Nazaire Cedex, FRANCE, phone: 33-2-40-17-26-11, fax: 33-2-40-17-26-18, e-mail: Mariana.Titica@univ-nantes.fr

The models used at microalgae culturing are as many as the objectives for which they were developed and as complex as their utility. As commonly applied in photobioreactor modeling, they can describe the kinetics of the photosynthetic growth coupled with the light transfer inside the culture, which needs to be modeled as soon as the light is absorbed by cells. Accurate formulation of such a coupling, to correctly consider its influence on the process, is a problem on its own [7].

In an optimal system where no factors limit the growth, the rate of photosynthesis and productivity is determined by the light availability [9].

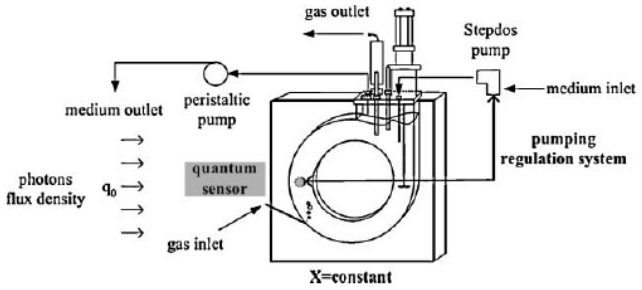


Figure 1 - Schematic representation of the photobioreactor operating system in turbidostat mode [7]

Various works were performed on photosynthetic growth modeling [10], [11]. The specific growth rate (μ) increases along with the increasing irradiance, reaching a maximum value, μ_{max}. Further increase in irradiance may inhibit growth – a phenomenon known as photoinhibition. Although this phenomenon is well documented, it has often been disregarded [12], [13].

Fouchard *et al.* [7] proposed a model which describes the three stages which precede the hydrogen production under sulfur limitation conditions.

The model retained

The model underlying the work presented in this paper is the model proposed by Fouchard *et al.* [7]. It introduces a continuous formulation to describe the progressive transition from oxygenic growth to anoxia in order to obtain biohydrogen. The model is expected to be independent of the case under study, with corresponding parameters estimated from individual sets of experiments. This condition is important for optimizing the culture conditions and to investigate new protocols for biohydrogen and biomass production [7].

According to Fouchard *et al.* the process is divided in three phases: photosynthetic growth, sulfur deprivation and hydrogen production. For the mathematical simulation of the process it was considered only the first phase in order to demonstrate that the model, with the parameters already identified [7], can be also used in biomass production processes (independent of hydrogen production conditions).

Because light is a limiting factor in the model formulation must be introduced the coupling between a radiative model [14] and a Haldane one to represent light-dependent photosynthetic growth kinetics [15]. The light transfer inside the culture – irradiance G – is dependent of the photobioreactor geometry. The torus-shaped photobioreactor [14], [15] under study (Figure 1) enables the one-dimensional hypothesis to be applied – light attenuation occurring in only one direction namely the depth of culture *z* which is perpendicular to the illuminated surface.

The two-flux model can then be used and the following formulation of irradiance (Eq.1) distribution can be employed as follows:

$$G(z) = 2q_0 \frac{(1+\alpha)e^{\delta(L-z)} - (1-\alpha)e^{-\delta(L-z)}}{(1+\alpha)^2 e^{\delta L} - (1-\alpha)^2 e^{-\delta L}} \quad (1)$$

where $\delta = X\sqrt{E_a(E_a + 2bE_s)}$ is the two-flux extinction coefficient, and $\alpha = \sqrt{(E_a)/(E_a + 2bE_s)}$ is the linear scattering modulus. E_a and E_s are the mass absorption and the mass scattering coefficients ($m^2 \cdot kg^{-1}$), and b is the backward scattering fraction (dimensionless). q_0 represents the hemispherical incident light flux. X represents the biomass concentration inside the photobioreactor ($kg \cdot m^{-3}$) and L represents the depth of the photobioreactor (m).

Light dependency is represented by a photosynthetic growth model with an inhibitory term [16] to characterize the small decrease of growth rate that can be observed for high irradiance (photoinhibition):

$$\mu_G = \mu_{\max} \frac{G}{K_I + G + \frac{G^2}{K_{II}}} \quad (2)$$

where K_I is the half-saturation constant and K_{II} the inhibition constant.

The specific growth rate expressed in this form (Eq.2) enable us to determine the local photosynthetic response $\mu_G(G(z))$. However in our simulation there was used the average photosynthetic response $\langle \mu_G \rangle$ calculated all over the reactor's volume, obtained by integrating local photosynthetic responses (Eq.3).

$$\langle \mu_G \rangle = \frac{1}{L} \cdot f(Q) \cdot \int_0^L \mu_G(G(z))dz \quad (3)$$

The mass balance equation (Eq.4) was used to illustrate the evolution of biomass concentration *X(t)*:

$$\frac{dX}{dt} = \langle r_X \rangle - DX \quad (4)$$

In this equation D (h^{-1}) is the dilution rate and $\langle r_X \rangle$ is the volumetric rate which is detailed in [7].

The connection between biomass and sulfur concentrations (sulfur limitation representing the protocol for biohydrogen production) is given by *f(Q)* function which represents the intracellular sulfur quota *Q* influence on the photosynthetic activity. It is inserted as a factor in volumetric rate $\langle r_X \rangle$, which can take values from 0 to 1. In the case of non-limiting intracellular sulfur quota, the specific growth rate is only related to light limitation – *f(Q)* = 1.

In the following section which regards the control of the photosynthetic growth process the sulfur substrate was considered unlimited and uninhibitory, therefore it does not influences the biomass concentration.

III. CONTROLLER DESIGN AND TEST IN SIMULATION

The biomass concentration is one of the most important variables, which needs to be controlled although it is the desired output or not. The biomass concentration values can be acquired through sampling or on-line measurements based on principles like cytometry or optical density. Currently, the most used on-line measurement of the biomass is the turbidity which represents a cheap and fast solution. However, the turbidity sensors need to be periodically calibrated and often cleaned in place. The turbidity is a non-discriminative measurement which includes all suspended solids, and aggregate formation can induce erroneous measurements.

Classical manners to control the biomass production in continuous photobioreactor are the turbidostat and the chemostat. In these cases the biomass *X* or the substrate *S* are kept constant through control variables such as the dilution rate (*D*) or/and the incident light flux (q_0) – to optimize the production system and to avoid the photoinhibition in early growth phases (Figure 2).

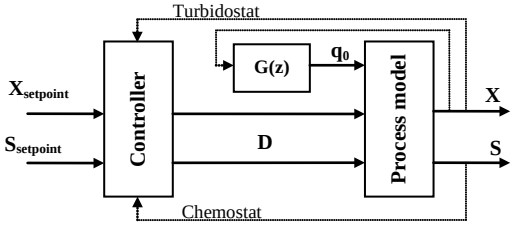


Figure 2 – General control structure of the micro-algae growth process

The control algorithms designed in this work considered only dilution rate D as control variable while the incident light flux was kept to a constant value.

Before choosing a proper solution for a controller, the biological system was linearized around a stationary steady state (nominal operating point). Having as target the control of the biomass *X(t)* through a command on the dilution *D*, it was determined the optimal dilution (D_o) which corresponds to the maximal performance. The performance of the system was determined by computing the performance index *I* (Eq.5) which represents the total amount of biomass obtained at the output of the photobioreactor after a given period of time *t*.

$$I = D \cdot V \cdot \int_0^t X dt = F \cdot \int_0^t X dt \tag{5}$$

where *V* is the photobioreactor volume (1L) and *F* is the flow rate (L/h).

Figure 3 shows the variation of the total amount of biomass produced after 5 days of cultivation in accordance with the dilution rate. The biomass amount (expressed in grams) was determined at four different values of the incident light flux (q_0): 110, 350, 650 and 1100 µmol photon·m^2·s.

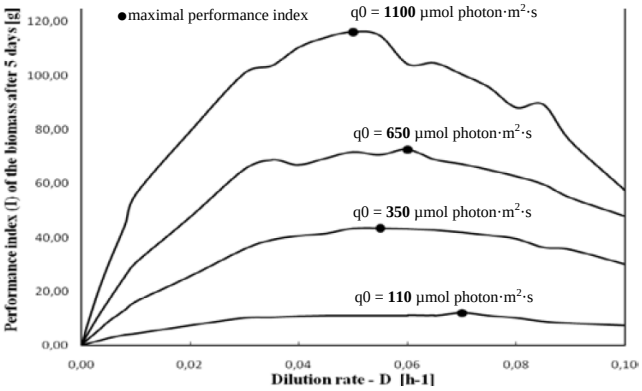


Figure 3 – The performance index variation at different light intensities

As it can be seen in figure 3 the optimal dilution takes values between 0.05 and 0.07 h^{-1}, on the given incident light flux interval. For further linearization there were considered the following nominal values: q_0=110 µmol photon·m^2·s involving D_o = 0.07 h-1. Considering the proposed values, the steady state value of the biomass was determined and the linearization was made around that point. The steady state value of the biomass was determined as being 0.18 g/L.

To linearize the system, the mass balance equation of the biomass was reconstructed in *Simulink*®. The linearization was made using the `linmod` function of *Matlab*® ver. 7.9 which returned the following transfer function (Eq. 6):

$$H(s) = \frac{-4.286}{s + 0.846} \tag{6}$$

Because the transfer function denominator is a first degree equation the most suitable controller for this case is a PI (proportional-integral) one.

A. The PI control of the biomass growth process

Having a stable system (negative root of the denominator) and a linear transfer function the first step was to synthesize

the control architecture of the system (Figure 4).

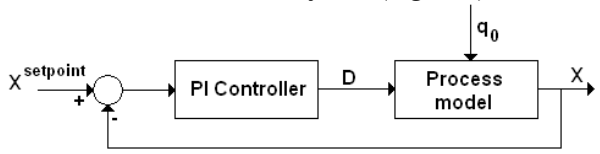


Figure 4 – The PI control structure for the biomass growth process

To determine the coefficients of the linear controller (Eq. 7), the linear system (Eq. 6) was loaded into the `rltool` of *Matlab*® software ver. 7.9.

$$H(s) = K_P\left(1 + \frac{1}{T_i s}\right) \tag{7}$$

where K_P is the proportional gain and T_i is the integral time setting of the controller.

The values of the coefficients were determined through graphical tuning.

The next step implied was to implement the linear controller into the nonlinear model and for this the following control (Eq. 8) was used:

$$D(t) = D_o + K_P \cdot e(t) + \frac{K_P}{T_i}\int_0^t e(t) \tag{8}$$

e(t) represents the error: biomass measured value subtracted from the setpoint.

Even if the controller is initialized with steady state values and the optimal dilution is inherent, its addition in the control variable formulation has an overshoot limiting effect of the integral term (Figure 5). Besides, an anti-windup structure implemented on the nonlinear model had a further intense reducing effect of the overshoot (Figure 5).

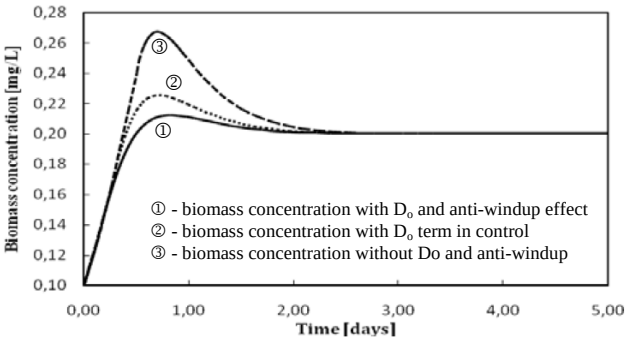


Figure 5 – The cumulated effect of the anti-windup structure and D_o (for $K_P = 1$ and $T_i = 0.5$)

It is well known that the shorter the integral time (T_i) is, the more often the proportional correction is repeated, and thus the integral contribution is more effective. If the system allows a higher integral time (T_i), the integral contribution will be more limited and the controller will "hit" faster and more accurately the setpoint. In figure 6 there are rendered the evolution of the biomass concentration and the control variable profile. The setpoint was chosen in the vicinity of the steady state (0.2 g/L).

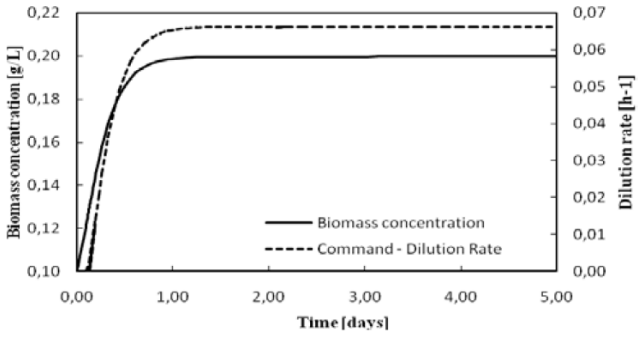


Figure 6 – Biomass concentration in a PI controlled system ($K_P = 1$, $T_i = 5$) and the control variable of the system

In figure 6 one can observe that the biomass reaches the setpoint value in less than 1 day with no overshoot, and chimes with the technological reality. The control variable is a smooth one which stabilizes at a steady value, namely the optimal dilution D_o.

The system must also have the ability to track different setpoints. Figure 7 illustrates the biomass evolution (at setpoints other that the steady state one) and the control variable (dilution rate) in constant light flux (110 µmol photon·m^2·s). The simulation was realized for a 20 days period conferring 5 days for every chosen setpoint: 0.2 g/L, 0.35 g/L, 0.5 g/L and back to 0.2 g/L.

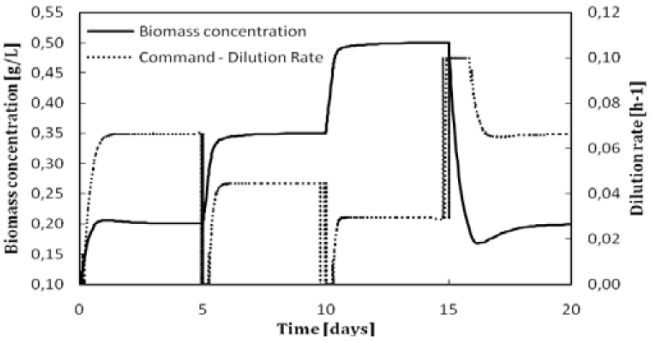


Figure 7 – Biomass concentration at different setpoints ($K_P = 1$, $T_i = 1$) and dilution rate needed to reach the setpoint

As it can be observed in figure 7 the biomass reaches the desired setpoint after more than 2 days which is considered to be a very slow response.

Another important variable of the system is the incident light flux whose variation substantially influences the biomass concentration. Hereinafter (figure 8) a variable incident light flux was imposed, the simulation being made for the same period of 20 days. On the first three days the photosynthetic growth deployed at 110 µmol photon·m^2·s, on the next 5 days the light flux was raised at 250 µmol photon·m^2·s, after another 5 days heightened at 350 µmol photon·m^2·s and on the last 8 days dropped back to 110 µmol photon·m^2·s.

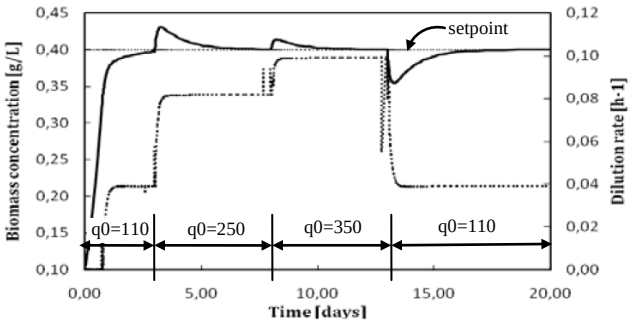


Figure 8 – Biomass concentration at different incident light fluxes (K_P = 1, T_i=1) – solid line, dilution rate for tracking the setpoint – dotted line and setpoint – thin dotted line

As it can be observed in figure 8 the controller reaches the setpoint (0.4 g/L) after more that 2 days, imposing an error of over 10%.

B. The Linearizing control of the biomass growth process

A more accurate method to track the setpoint of a biological system is the linearizing control which gives withal a faster response.

As it is well known, the linearizing control is a non-linear one designed to achieve a linear closed loop which is unconditionally stable no matter the operating point [17].

Let us consider the general model (Eq.9) of a biotechnological process:

$$\frac{d\xi}{dt} = K\varphi(\xi) - D\xi + F - Q \tag{9}$$

with ξ - state vector (dim(ξ)=N), φ - reaction rate vector (dim(φ)=M), F – flow rate and Q – gaseous product. The objective is to control a scalar output, y, which is a linear combination of state variables (Eq.10) that can be measured in the process.

$$y = \sum_{k=1}^{N} c_k \xi_k = C^T \xi \tag{10}$$

Any of the dilution rate D or the flow rate F can be considered as control inputs denoted by u. The control objective of the system is to track a reference signal $y^*(t)$. The principle of the linearizing control is to design a control law which is a multivariable nonlinear function of ξ, y^*, F and Q so that the tracking error is given by a prespecified stable linear model (reference model).

The linearizing control design procedure consists in three steps, as follows:

1. Establishing a model for variable y (a δ^{th} order differential equation – where δ is called relative degree).

$$\frac{d^{\delta} y}{dt^{\delta}} = f_0(t) + u(t) f_1(t) \tag{11}$$

2. Selecting a stable linear reference model of the tracking error (Eq.12):

$$\sum_{k=0}^{\delta} \lambda_{\delta-k} \frac{d^k}{dt^k}\left[y^*(t) - y(t)\right] = 0, \quad \lambda_0 = 1 \tag{12}$$

The coefficients $\lambda_{\delta-k}$ are chosen so that the differential equation (Eq.11) to be stable.

3. Calculus of the control law $u(t)$ so that the *I/O* model (Eq.11) exactly matches the reference model (Eq.12):

$$u(t) = \frac{1}{f_1(t)}\left[-f_0(t) + \sum_{k=0}^{\delta-1} \lambda_{\delta-k} \frac{d^k}{dt^k}\left[y^* - y\right] + \frac{d^{\delta} y^*}{dt^{\delta}}\right] \tag{12}$$

In the case of the microalgae photosynthetic growth process the model order can be considered equal to 1 (the nutrient concentration is considered unlimited), this means that the relative degree δ is also equal to 1. The structure of the linearizing control system is represented in figure 9.

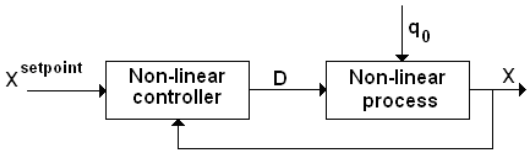


Figure 9 – Linearizing control structure for the biomass growth process

The following linearizing control was determined (Eq.13) according to the rules presented above (the dimensionless coefficient λ_1 was chose so that the system reaches the maximal specific growth rate):

$$D = \frac{r_x - \lambda_1 \cdot e}{24 \cdot X} \tag{13}$$

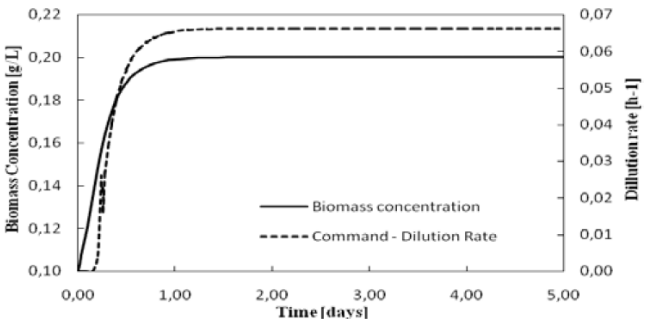


Figure 10 – Biomass concentration in a linearizing control system and the control variable of the system (λ_1=5)

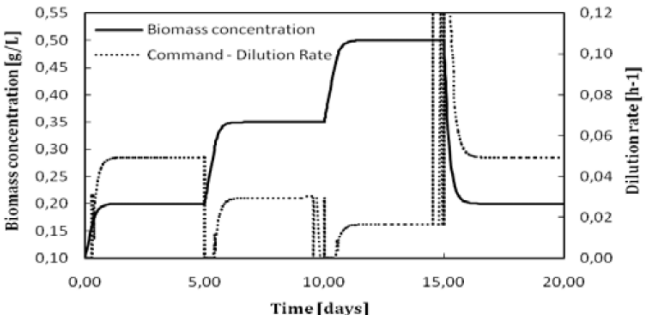


Figure 11 – Biomass concentration at different setpoints and dilution rate needed to reach the setpoint (λ_1=5)

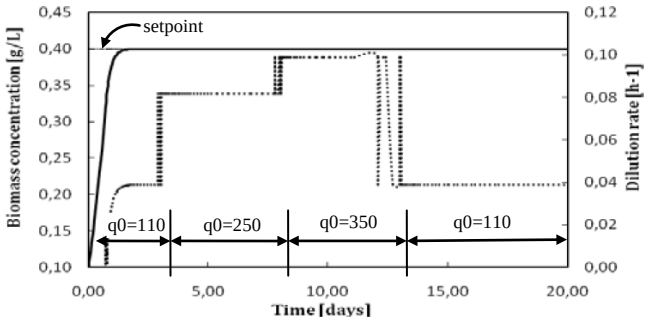


Figure 12 – Biomass concentration at different incident light fluxes – solid line, dilution rate for tracking the setpoint – dotted line and setpoint – thin dotted line (λ_1=5)

The simulation was made in the exact same conditions as the ones imposed for the linear PI controller.

In figure 10 it can be observed that the setpoint is accurately tracked by the nonlinear control, the system stabilizing itself at the same optimal dilution as the PI controller. However these details will be later validated in practice. The real difference between the two control laws can be observed in figures 11 and 12. The nonlinear control is not designed around a steady state value and, as a consequence, it can work better at various setpoints giving a faster and better response for a similar control variable. Moreover, the results for incident light flux variations are far better than the ones obtained with the PI controller in the same conditions.

IV. Conclusions

The linear PI controller gives satisfactory results in what regards the photosynthetic growth process, but the disadvantage is that it only works well around the steady state for which it was tuned. The results could be improved by designing a nonlinear PI controller able to adapt itself at a wider range of setpoints and light intensities.

Instead the linearizing control gives better results, being independent of a steady state value and easy to adjust, but it is more complex and requires a validated model and knowledge on the specific growth rate (μ). Its implementation compels the use of a process computer, whereas the PI structure is easier to implement on industrial scale.

The further objective is to validate both control laws on laboratory scale. Additional work will be done on the nonlinear linearizing controller implementation which will be coupled with a software sensor that gives the on-line estimation of the specific growth rate from gas measurement (O_2, CO_2).

Acknowledgement

The authors acknowledge the support of POSDRU – Grant No. 6/1.5/S/15 – 6583, acronym SIMBAD, and of SOLAR-H_2 EU project number 212508.

References

[1] Seckbach J., "Algae and Cyanobacteria in Extreme Environments", Springer Press (2007)

[2] Spolaore P, Joannis-Cassan C, Duran E, Isambert A, "*Commercial applications of microalgae*". J Biosci Bioeng (2006) 101:87–96

[3] Greenwell H. C., Laurens L. M. L., Shields R. J., Lovitt R. W. and Flynn K. J., "*Placing microalgae on the biofuels priority list: a review of the technological challenges*". J. R. Soc. Interface (2010) 7, 703–726

[4] Brennan L., Owende P., "*Biofuels from microalgae – A review of technologies for production, processing, and extractions of biofuels and co-products*", in press (2009) doi:10.1016/j.rser.2009.10.009

[5] Fouchard S., Pruvost J., Degrenne B., Legrand J., "*Investigation of H_2 production using the green microalga Chlamydomonas reinhardtii in a fully controlled photobioreactor fitted with on-line gas analysis*", International Journal Of Hydrogen Energy 33 (2008) 3302 – 3310

[6] Cornet J.F. and Dussap C.G., "*A Simple and Reliable Formula for Assessment of Maximum Volumetric Productivities in Photobioreactors*", Biotechnol. Prog. Vol. 25, No. 2 (2009) 424 – 435

[7] Fouchard S., Pruvost J., Degrenne B., Titica M., Legrand J., "*Kinetic Modeling of Light Limitation and Sulfur Deprivation Effects in the Induction of Hydrogen Production With Chlamydomonas reinhardtii: Part I. Model Development and Parameter Identification*", Biotechnology and Bioengineering, Vol. 102, No. 1 (2009) 232 – 245

[8] Fouchard S., Pruvost J., Legrand J., "*Investigation of H2 production by microalgae in a fully-controlled photobioreactor*", WHEC 16th/ 13-16 June 2006 – Lyon France

[9] Aiba S., "*Growth kinetics of photosynthetic microorganisms*". Adv. Biochem. Eng. 23 (1982) 85–156.

[10] Molina Grima E., Fernandez Sevilla J.M., Sanchez Perez J.A., Garcia Camacho F., "*A study on simultaneous photolimitation and photoinhibition in dense microalgal cultures taking into account incident and averaged irradiances*" J. Biotechnol. 45 (1996) 59 – 69

[11] Muller-Feuga A., "*Growth as a function of rationing: a model applicable to fish and microalgae*", Journal of Experimental Marine Biology and Ecology, 236 (1999) 1 – 13

[12] Cornet J.-F., Dussap C.G., Gros J.B., "*A simplified monodimensional approach for modelling coupling between radiant light transfer and growth kinetics in photobioreactors*", Chemical Engineering Science, 50 (1995) 1489 – 1500

[13] Carvalho A.P. and F.X. Malcata, "*Kinetic modelling of the autotrophic growth of Pavlova lutheri: study of the combined influence of light and temperature*", Biotechnology Progress, 19 (2003) 1128 – 1135

[14] Pottier L, Pruvost J, Deremetz J, Cornet J, Legrand J, Dussap C. 2005. "*A fully predictive model for one-dimensional light attenuation by Chlamydomonas reinhardtii in a torus photobioreactor*". Biotechnol Bioeng 91 (2005) 569 – 582

[15] Andrews J. "*A mathematical model for the continuous culture of microorganisms utilizing inhibitory substrates*". Biotechnol Bioeng 10 (1968) 707 – 723

[16] Pruvost J, Pottier L, Legrand J. 2006. "*Numerical investigation of hydrodynamic and mixing conditions in a torus photobioreactor*". Chem Eng Sc 61 (2006) 4476 – 4489

[17] Bastain G. and Dochain D., "*On-line Estimation and Adaptive Control of Bioreactors*", Elsevier Press (1990)

DSM Control of Inventory Systems with Deteriorating Stock – the Case of a Single Supply Source

P. Ignaciuk and A. Bartoszewicz

Abstract—**In this paper we propose a control-theoretic approach for the design of inventory management policy for systems with perishable goods. The challenging issue in this type of systems is to achieve a high service level with minimum costs for arbitrary demand when the replenishment orders are procured with positive lead-time. In contrast to the classical stochastic approaches, we employ a formal design methodology based on discrete sliding-mode (DSM) control. The proposed DSM controller with the sliding plane selected for a dead-beat scheme ensures a maximum service level with smaller holding costs and reduced order-to-demand variance ratio as compared to the classical order-up-to policy.**

I. Introduction

An appropriate inventory management policy is crucial for efficient operation of production and logistic systems [1]. Due to the similarity between the considered class of systems and engineering processes, it is a natural choice to apply control-theoretic methods in the design and analysis of strategies governing the flow of goods. However, it follows from the extensive review papers documenting the research work in the field [2]–[7] that certain areas of inventory control are not sufficiently addressed at the formal design level. The deficiency of application of systematic control approaches concerns in particular a large and very important class of problems related to the management of perishable commodities. Indeed, many products, such as food, drugs, gasoline, etc., lose market value over time, deteriorate due to changes in chemical structure or even become obsolete. The primary difficulty in developing control schemes for perishable inventories is the enlarged state space required for conducting the exact analysis of product lifetimes. The situation aggravates when the product demand varies rapidly in subsequent review periods and inventories are replenished with a nonzero delay, which frequently happens in modern supply chains. In such circumstances, in order to meet the service level constraints and at the same time keep stringent cost discipline, when placing an order it is necessary not only to account for the demand during procurement latency but also for the stock deterioration in that time. There are very few successful design examples based on formal control methods for perishable inventory systems. In paper [8], a linear-quadratic optimization is performed for a non-delayed process. Rodrigues and Boukas [9] design a piecewise affine control law for a production system with deteriorating on-hand inventory and zero lead-time. In [10], a robust controller for the continuous system with uncertain processing time and delay in control is designed by minimizing an H_∞-norm. However, the implementation of the strategy proposed in [10] requires numerical procedures for obtaining the control law parameters, which limits its analytical tractability.

In this paper, we apply a control-theoretic methodology to develop a new supply policy for periodic-review inventory systems with perishable goods. In the considered systems, the on-hand stock at a distribution center is used to fulfill an unknown, time-varying demand placed by retailers (or customers). The stock deteriorates exponentially at a constant rate and is replenished with delay from a supply source. We assume that the lead-time delay can span multiple review periods. The design objective is to obtain a high service level of the (unknown) customers' demand with minimum on-hand inventory. For this purpose, we propose discrete-time sliding-mode (DSM) control, which is well known to be efficient and robust regulation technique [11]. Since a proper choice of the switching plane is the key part of the design of sliding-mode controllers, especially in discrete-time domain [12], in this work, we determine the plane parameters for a dead-beat control. In this way we obtain fast response to the changes in demand and the minimum stock level. In contrast to other solutions reported previously in literature for perishable inventory systems, we adopt a formal design approach based in part on our previous results reported for the traditional inventory systems (with nondeteriorating stock) [13, 14], and obtain the controller in a closed-form. The closed-form solution allows us to define a number of advantageous properties of the proposed control scheme. In particular, we show that under the proposed policy the available stock is never entirely depleted despite unpredictable demand variations, which guarantees a maximum service level. We also specify a precise value of the storage space which should be reserved at the distribution center to always accommodate all the incoming shipments. This means that the potential

This work has been financed by the Polish State budget in the years 2010–2012 as a research project N N514 108638 "Application of regulation theory methods to the control of logistic processes". The first author gratefully acknowledges financial support provided by the Foundation for Polish Science (FNP). He is also a scholarship holder of the project entitled "Innovative Education without Limits – Integrated Progress of the Technical University of Łódź" supported by the European Social Fund.

P. Ignaciuk (e-mail: przemyslaw.ignaciuk@p.lodz.pl) and A. Bartoszewicz (e-mail: andrzej.bartoszewicz@p.lodz.pl) are with Institute of Automatic Control, Technical University of Łódź, 18/22 Stefanowskiego St., 90-924 Łódź, Poland.

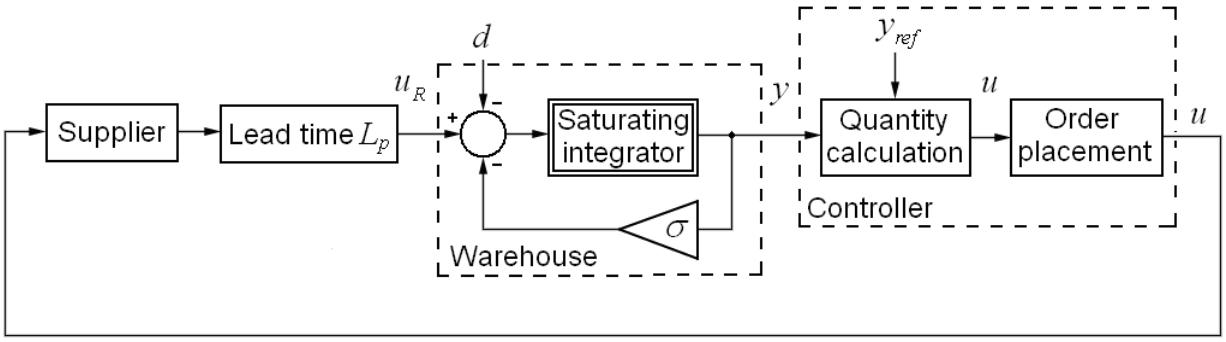


Fig. 1. System model.

necessity of expensive emergency storage outside the company premises is eliminated. Finally, we show that the order quantities generated by the presented controller are always nonnegative and bounded, which is required from the practical point of view.

II. Problem Formulation

The model of the analyzed periodic-review system is illustrated in Fig. 1. The stock replenishment orders u are issued at regular intervals kT, where T is the review period and $k = 0, 1, 2,...$. The order quantity is calculated on the basis of the current stock level $y(kT)$, the stock reference value y_{ref} and the orders history. Each non-zero order placed at the supplier is realized with lead-time L_p assumed to be a multiple of the review period, i.e. $L_p = n_pT$, where n_p is a positive integer. The saturating integrator in an internal loop represents the operation of accumulating the stock of perishable goods characterized by decay factor σ.

The imposed demand (the number of items requested from inventory in period k) is modeled as an *a priori* unknown, bounded function of time $d(kT)$,

$$0 \le d(kT) \le d_{\max}. . \tag{1}$$

Notice that this definition of demand is quite general and it accounts for any standard distribution typically analyzed in the considered problem. If there is a sufficient number of items in the warehouse to satisfy the imposed demand, then the actually met demand $h(kT)$ (the number of items sold to customers or sent to retailers in the distribution network) will be equal to the requested one. Otherwise, the imposed demand is satisfied only from the arriving shipments, and additional demand is lost (we assume that the sales are not backordered, and the excessive demand is equivalent to a missed business opportunity). Thus, we may write

$$0 \le h(kT) \le d(kT) \le d_{\max}. \tag{2}$$

For the considered system with perishable inventory the stock balance equation takes the following form

$$y\left[(k+1)T\right] = \rho y(kT) + u_R(kT) - h(kT), \tag{3}$$

where $u_R(kT)$ is the order received in period k and $\rho = 1 - \sigma$ represents the fraction of stock which remains in the warehouse when inventory deteriorates at rate σ. For instance, if $\sigma = 0.05$, then 5% of the stock perishes in each review period and $\rho = 0.95$ or 95% of the stock remains. We assume that the warehouse is initially empty, i.e. $y(kT) = 0$ for $k < 0$, and the first order is placed at $kT = 0$. Consequently,

$$y(kT) = \sum_{j=0}^{k-1} \rho^{k-1-j} u_R(jT) - \sum_{j=0}^{k-1} \rho^{k-1-j} h(jT). \tag{4}$$

Because of lead-time delay the first order arrives at the distribution center in period n_p, and $y(kT) = 0$ for $k \le n_p$. We assume that the goods arrive at the distribution center new and deteriorate if kept in the on-hand stock. Taking into account the initial conditions and the fact that $u_R(kT) = u[(k - n_p)T]$, the stock level for any $k \ge 0$ may be calculated from the following equation

$$\begin{aligned} y(kT) &= \sum_{j=0}^{k-1} \rho^{k-1-j} u\left[(j-n_p)T\right] - \sum_{j=0}^{k-1} \rho^{k-1-j} h(jT) \\ &= \sum_{j=-n_p}^{k-n_p-1} \rho^{k-n_p-1-j} u(jT) - \sum_{j=0}^{k-1} \rho^{k-1-j} h(jT) \\ &= \sum_{j=0}^{k-n_p-1} \rho^{k-n_p-1-j} u(jT) - \sum_{j=0}^{k-1} \rho^{k-1-j} h(jT). \end{aligned} \tag{5}$$

In order to save on notation in the remainder of the paper we will use k as the independent variable in place of kT.

The considered discrete-time system can also be described in the state space as

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{b}u(k) + \mathbf{v}h(k), \\ y(k) &= \mathbf{q}^T \mathbf{x}(k), \end{aligned} \tag{6}$$

where $\mathbf{x}(k) = [x_1(k)\ x_2(k)\ ...\ x_n(k)]^T$ is the state vector with $x_1(k) = y(k)$ representing the stock level in period k and

$x_j(k) = u(k-n+j-1)$ for any $j = 2, \ldots, n$ equal to the delayed input signal u; $\mathbf{A}$ is $n \times n$ state matrix, $\mathbf{b}$, $\mathbf{v}$, and $\mathbf{q}$ are $n \times 1$ vectors

$$\mathbf{A} = \begin{bmatrix} \rho & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & 1 \\ 0 & 0 & 0 & \ldots & 0 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}, \mathbf{v} = \begin{bmatrix} -1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}, \mathbf{q} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}, \tag{7}$$

and the system order $n = n_p + 1 = L_p/T + 1$ depends on review period T and lead-time L_p. The desired system state $\mathbf{x_d} = [x_{d1}\ x_{d2} \ldots x_{dn}]^T = [1\ 1-\rho\ 1-\rho \ldots 1-\rho]^T y_{ref}$, and y_{ref} denotes the reference stock level.

III. Proposed Inventory Policy

In this section, we design a new supply policy for the considered inventory system with perishable goods. We adopt a formal approach based on the theory of discrete sliding-mode control. First, the design procedure is conducted, and a closed-form solution of the control law is presented. Afterwards, the important properties of the obtained controller related to the flow of goods are defined as three theorems.

A. Sliding-mode controller design

Let us define the system error as

$$\mathbf{e}(k) = \mathbf{x_d} - \mathbf{x}(k). \tag{8}$$

We introduce a sliding hyperplane described by the following equation

$$s(k) = \mathbf{c}^T \mathbf{e}(k) = 0, \tag{9}$$

where $\mathbf{c}^T = [c_1\ c_2 \ldots\ c_n]$ is the vector describing the sliding plane such that $\mathbf{c}^T\mathbf{b} \neq 0$. Substituting (6) into equation $\mathbf{c}^T\mathbf{e}(k+1) = 0$, the following feedback control law can be derived

$$u(k) = \left(\mathbf{c}^T\mathbf{b}\right)^{-1} \mathbf{c}^T \left[\mathbf{x_d} - \mathbf{A}\mathbf{x}(k)\right]. \tag{10}$$

Using (7) we can rewrite (10) as

$$u(k) = y_{ref}\left[c_1 + (1-\rho)\sum_{j=2}^{n} c_j\right] - c_n^{-1}\left\{c_1\rho x_1(k) + \sum_{j=2}^{n} c_{j-1} x_j(k)\right\}. \tag{11}$$

It is clear from (11) that the controller properties will be determined by an appropriate choice of the sliding plane parameters $c_1, c_2, \ldots, c_n$. Since typically in inventory control it is favorable to provide fast reaction to the changes in market conditions, we intend to find such parameters of the plane which will allow for the error elimination in the smallest number of steps after a demand surge (or decline).

The closed-loop state matrix $\mathbf{A_c} = [\mathbf{I_n} - \mathbf{b}(\mathbf{c}^T\mathbf{b})^{-1}\mathbf{c}^T]\mathbf{A}$ with control (11) applied is determined as

$$\mathbf{A_c} = \begin{bmatrix} \rho & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & 1 \\ -\frac{c_1\rho}{c_n} & -\frac{c_1}{c_n} & -\frac{c_2}{c_n} & \ldots & -\frac{c_{n-1}}{c_n} \end{bmatrix}, \tag{12}$$

and its characteristic polynomial as

$$\det\left(z\mathbf{I_n} - \mathbf{A_c}\right) = z^n + \frac{c_{n-1} - \rho c_n}{c_n} z^{n-1} + \ldots + \frac{c_2 - \rho c_3}{c_n} z^2 + \frac{c_1 - \rho c_2}{c_n} z. \tag{13}$$

For a dead-beat control $\det(z\mathbf{I_n} - \mathbf{A_c})$ should be equal to z^n, which is satisfied when

$$c_{n-1} = \rho c_n,\ c_{n-2} = \rho c_{n-1}, \ldots,\ c_2 = \rho c_3,\ c_1 = \rho c_2. \tag{14}$$

Having solved recursively this set of equations we obtain the following vector describing the parameters of the sliding plane

$$\mathbf{c}^T = \begin{bmatrix} \rho^{n-1} & \rho^{n-2} & \rho^{n-3} & \ldots & \rho & 1 \end{bmatrix} c_n. \tag{15}$$

Substituting (15) into (11), we get the control law

$$u(k) = y_{ref} - \rho^n x_1(k) - \sum_{j=2}^{n} \rho^{n-j+1} x_j(k). \tag{16}$$

From (7) the state variables x_j ($j = 2, 3, \ldots, n$) may be expressed in terms of the control signal generated at the previous $n-1$ samples as $x_j(k) = u(k-n+j-1)$ Since $x_1(k) = y(k)$ and $n = n_p + 1$, we obtain

$$u(k) = y_{ref} - \rho^{n_p+1} y(k) - \sum_{j=k-n_p}^{k-1} \rho^{k-j} u(j). \tag{17}$$

B. Properties of the proposed inventory policy

Further in this section the properties of the proposed inventory policy will be defined in three theorems. The first theorem shows that the order quantities determined from the algorithm are always nonnegative and bounded, which is a

crucial requirement for the practical implementation of an inventory management scheme. The second proposition specifies the warehouse capacity which needs to be provided to always accommodate the on-hand stock and the incoming shipments. Finally, the third theorem indicates how to select the reference stock value in order to ensure a maximum service level.

Theorem 1: The order quantities generated by the proposed policy are always bounded, and for any $k \geq 0$ the ordering signal satisfies the following set of inequalities

$$\sigma y_{ref} \leq u(k) \leq \max\left[y_{ref}, \sigma y_{ref} + \rho^{n_p+1} d_{\max} \right]. \tag{18}$$

Theorem 2: If the proposed inventory policy is applied, then the stock level is always upper-bounded, i.e.

$$\bigvee_{k\geq 0} y(k) \leq y_{ref}. \tag{19}$$

Theorem 3: If the proposed inventory policy is applied, and the reference stock satisfies

$$y_{ref} > d_{\max} \sum_{j=0}^{n_p} \rho^j, \tag{20}$$

then for any $k \geq n_p + 1$ the stock level is strictly positive.

The properties will be demonstrated in numerical tests described in Section IV.

IV. Numerical Example

We verify the properties of the proposed policy (17) in a series of simulation tests. The system parameters are chosen in the following way: review period $T = 1$ day, lead-time $L_p = n_p T = 8$ days, inventory decay factor $\sigma = 0.05$, which implies $\rho = 1 - \sigma = 0.95$, and the maximum daily demand at the distribution center $d_{\max} = 80$ items. The simulations are run for two demand patterns illustrated in Fig. 2. The first pattern (I) reflects seasonal demand changes, whereas the second one (II) represents a stochastic setting of normally distributed demand with mean equal to 40 items and standard deviation equal to 30 items. For the purpose of comparison we repeat the test for a classical order-up-to (OUT) policy (order up to a target level y_{OUT} whenever the total stock – equal to the on-hand stock plus open orders – drops below y_{OUT}). Two different settings of y_{OUT} are considered: in the first simulation (curve b in the graphs) it is adjusted to achieve the same service level as our policy, whereas in the second one (curve c in the graphs) the order-up-to level is set such both controllers result in the identical storage space assignment.

In order to obtain the maximum service level, according to (20), the reference stock level should be bigger than 592 items. We select $y_{ref} = 595$ items. For the OUT policy we set two different order-up-to levels: in one simulation it is set as 895 items, whereas in the second one it is adjusted to 670 items. The orders generated by our controller (a) and the classical inventory policy (b) and (c) are shown in Fig. 3, and the on-hand stock in Fig. 4. It is clear from the graphs that the proposed controller quickly responds to the sudden changes in the demand trend without oscillations or overshoots in the case of demand (I) and reduces oscillations in case (II). Moreover, the stock does not increase beyond the warehouse capacity, and it never drops to zero after the initial phase which implies the 100% service level. The OUT policy exhibits oscillations and requires bigger storage space to accommodate the stock to achieve the same service level (curve b in Fig. 4), which implies an increased holding cost. On the other hand, if the safety stock level is reduced for the OUT policy to maintain the same storage space as the one imposed by our controller, the OUT service level decreases to 94% (demand I). In that case, large oscillations appear in the ordering signal generated by the OUT policy leading to the bullwhip effect, which is avoided by our scheme.

Figure 5 shows the evolution of the sliding variable. We can see from the graph that $s(k)$ immediately decreases from its original value $s(0) = 594$ items to a relatively narrow band $s \in [0, 54$ items) and then always remains in this band, which constitutes a clear evidence of a properly established sliding motion in discrete-time domain.

V. Conclusion

In this paper, a new supply policy for periodic-review inventory systems with deteriorating stock was designed using strict control-theoretic methodology. The proposed policy based on sliding-mode dead-beat control provides fast reaction to the changes in market conditions and stable system operation for arbitrary positive lead-time. It also guarantees that all of the demand is satisfied from the on-hand stock, thus eliminating the risk of missed service opportunities and necessity for backorders.

References

[1] P. H. Zipkin, *Foundations of Inventory Management*. New York: McGraw-Hill, 2000.

[2] M. Ortega and L. Lin, "Control theory applications to the production-inventory problem: a review," *Int. J. Prod. Res.*, 2004, vol. 42, no. 11, pp. 2303–2322.

[3] H. Sarimveis, P. Patrinos, C. D. Tarantilis, and C. T. Kiranoudis, "Dynamic modeling and control of supply chain systems: a review," *Comps. Oper. Res.*, 2008, vol. 35, no. 11, 3530–3561.

[4] S. Nahmias, "Perishable inventory theory: a review," *Oper. Res.*, 1982, vol. 30, no. 4, pp. 680–708.

[5] F. Rafaat, "Survey of literature on continuously deteriorating inventory models," *J. Oper. Res. Soc.*, 1991, vol. 42, no. 1, pp. 27–37.

[6] S. K. Goyal and B. C. Giri, "Recent trends in modeling of deteriorating inventory," *Eur. J. Oper. Res.*, 2001, vol. 134, no. 1, pp. 1–16.

[7] I. Karaesmen I, A. Scheller-Wolf, and B. Deniz, "Managing perishable and aging inventories: review and future research directions," In: K. Kempf, P. Keskinocak, and R. Uzsoy (eds.) *Handbook of production planning*. Kluwer, Dordrecht 2008.

[8] A. Andijani and M. Al-Dajani, "Analysis of deteriorating inventory/production systems using a linear quadratic regulator," *Eur. J. Oper. Res.*, 1998, vol. 106, no. 1, pp. 82–89.
[9] L. Rodrigues and E.-K. Boukas, "Piecewise-linear H_∞ controller synthesis with applications to inventory control of switched production systems," *Automatica*, 2006, vol. 42, no. 8, pp. 1245–1254.
[10] E.-K. Boukas, P. Shi, R. K Agarwal, "An application of robust control technique to manufacturing systems with uncertain processing time," *Opt. Control Appls. Methods*, 2000, vol. 21, no. 6, pp. 257–268.
[11] B. Bandyopadhyay and S. Janardhanan, *Discrete-Time Sliding Mode Control. A Multirate Output Feedback Approach*. Springer-Verlag, Berlin 2006.
[12] Č. Milosavljević, B. Peruničić-Draženović, B. Veselić, D. Mitić, "Sampled data quasi-sliding mode control strategies," *IEEE Int. Conf. Ind. Techn.* 2006, Mumbai, India, pp. 2640–2645.
[13] P. Ignaciuk and A. Bartoszewicz, "LQ optimal sliding mode supply policy for periodic review inventory systems," *IEEE Trans. Autom. Control*, 2010, vol. 55, no. 1, pp. 269–274.
[14] P. Ignaciuk, A. Bartoszewicz, "LQ optimal and reaching law based sliding modes for inventory management systems," *Int. J. Sys. Sci.*, 2010 (*in press*).

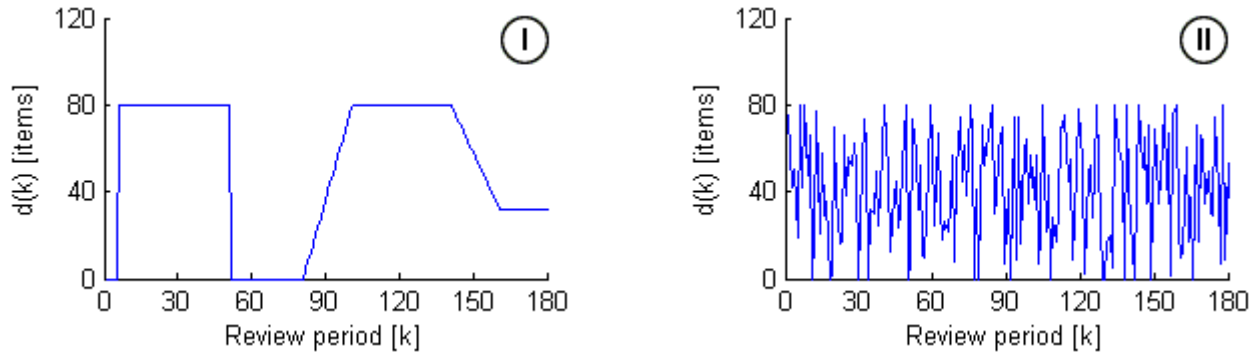


Fig. 2. Demand at the distribution center: I – demand trend, II – stochastic pattern with mean = 40 items and standard deviation = 30 items.

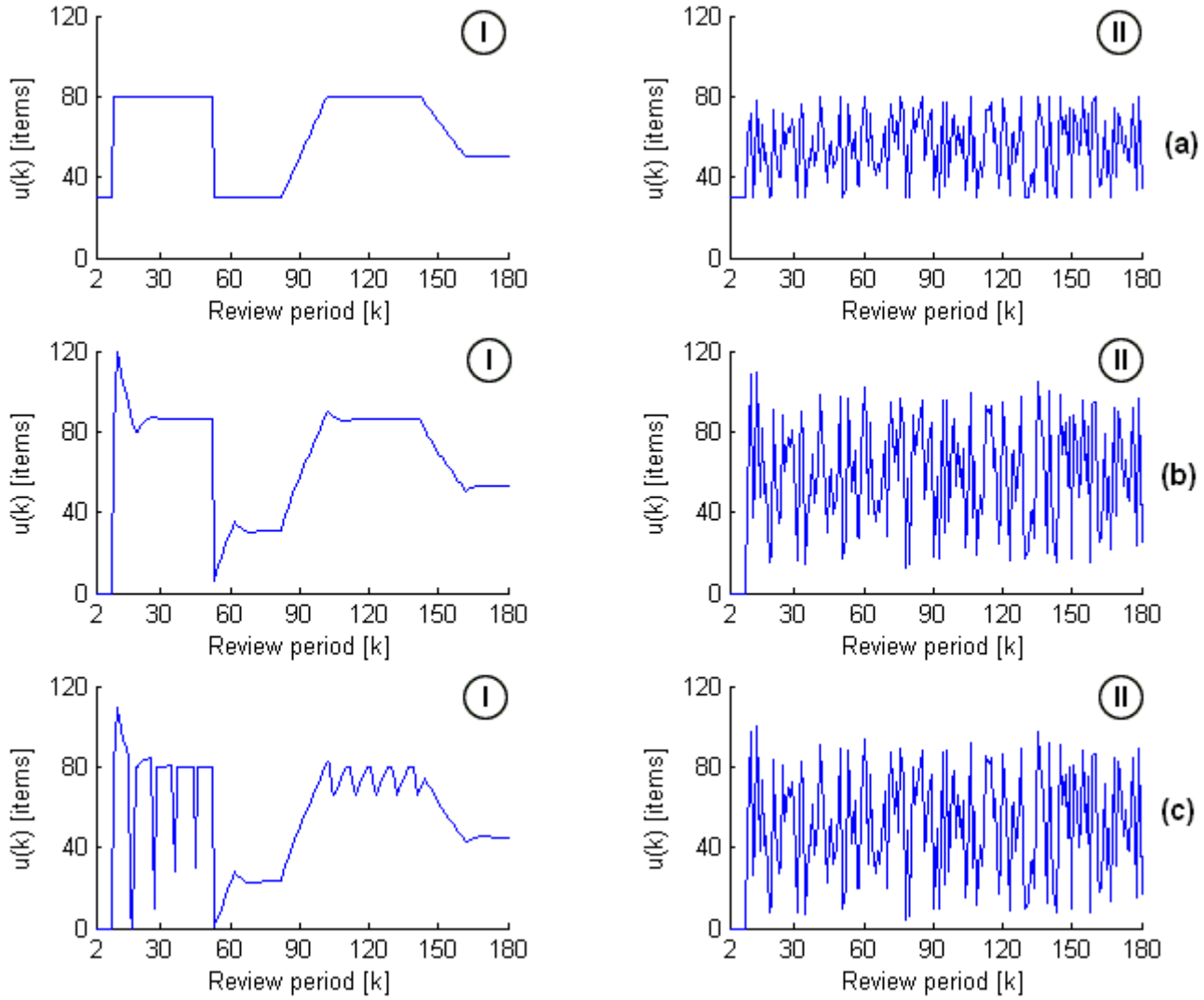


Fig. 3. Order quantities: a) policy (17), b) OUT policy (y_{OUT} = 895 items), c) OUT policy (y_{OUT} = 670 items).

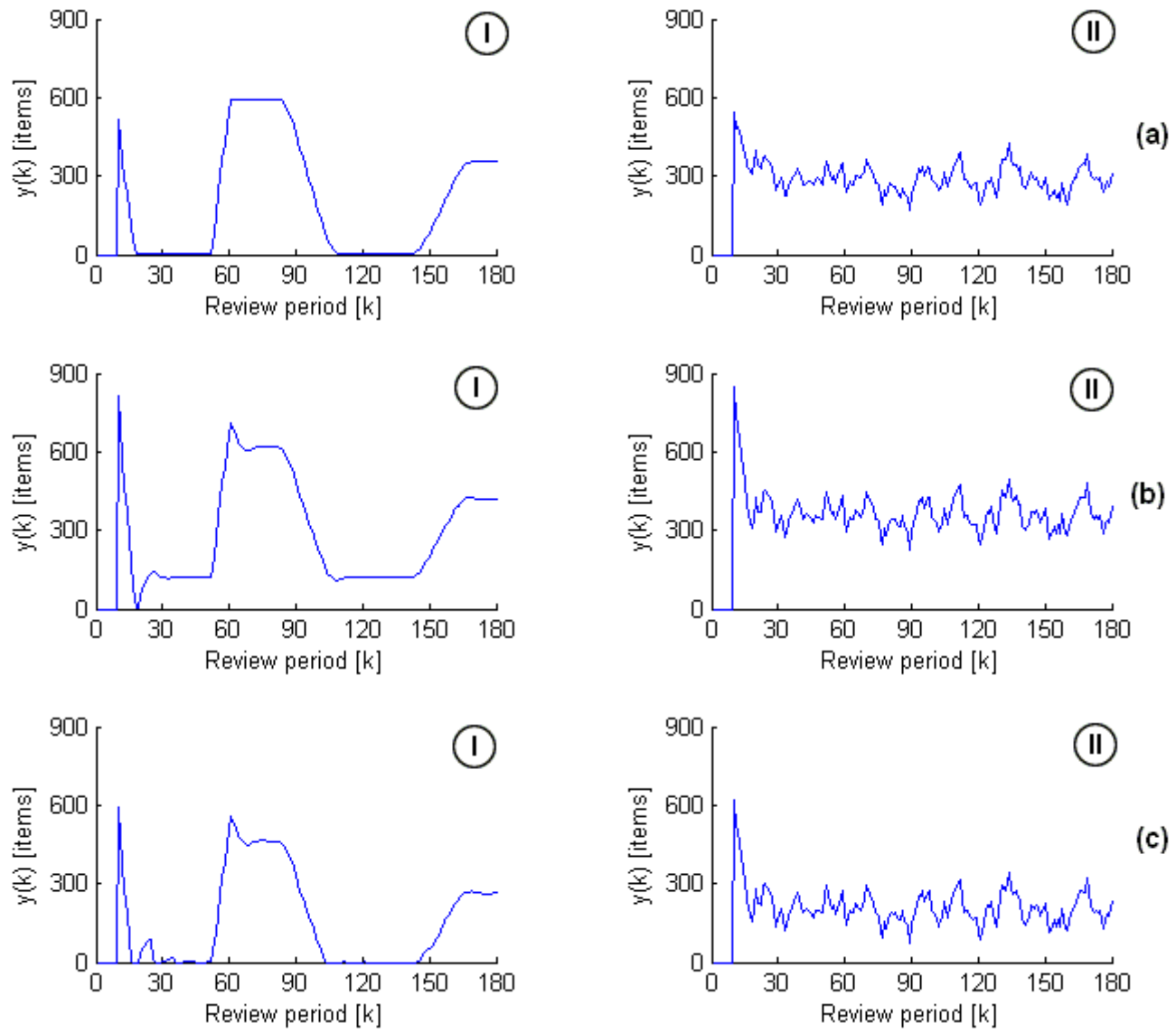


Fig. 4. Stock level: a) policy (17), b) OUT policy (y_{OUT} = 895 items), c) OUT policy (y_{OUT} = 670 items).

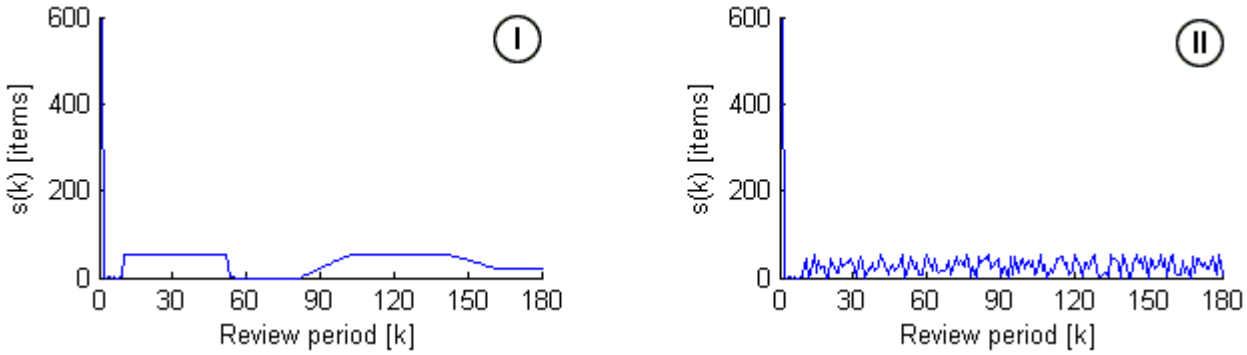


Fig. 5. Sliding variable.

An Electronic Support Measure (ESM) Angle of Arrival Measurement System

Jamshed Iqbal, Salman M. Khan and Faisal Masood

***Abstract*—The bearing (direction) of hostile radars is the most reliable de-interleaving parameter in an Electronic Support Measures (ESM) system, especially when the hostile radar exhibits frequency and Pulse Repetition Frequency (PRF) agility. This paper presents the design of a system that measures bearing using amplitude comparison technique. The system comprises of six monopulse directional antennas, receivers to obtain video signals, electronic circuits for amplitude comparison and a computer program to graphically display the bearing calculated from the results of amplitude comparison. The system is capable of finding Angle Of Arrival (AOA) of a pulse with the tolerance of ±6°.**

I. Introduction

In the past, beacon towers were used as a main source of sending signals. The signals were usually passed through series of towers before they can be directed toward their specific destination. Technological advancements in electronics and communication systems coupled with computer science revolution lead to replace this traditional approach. In many applications, including monitoring and tracking, the acquired data is meaningless without having information of positions of hostile. Angle of Arrival (AOA) or bearing is a method for determining the direction of propagation of a Radio-Frequency (RF) wave incident on an antenna array. Assuming a two element array which is spaced apart by ½λ of an incoming RF wave, if a wave is incident upon the array at bore sight, it will arrive at each antenna simultaneously. This will yield 0° phase-difference measured between the two antenna elements, equivalent to a 0° AOA. If a wave is incident upon the array at broadside, then a 180° phase difference will be measured between the elements, corresponding to a 90° AOA.

Measuring the bearing has its potential applications in wide range of areas. For example in finding the geodesic location or geo-location of cell phones. Multiple receivers on a Base Station (BS) can calculate the bearing of the cell phone's signal and this information can be combined to determine the exact phone's location on the earth. In case of wireless sensor network, awareness of the physical location for each node is required by many applications [1]. Similarly, AOA measurement can be used to find the location of any military radio transmitter. In optics, the bearing is considered from the perspective of interferometry while in submarine acoustics, bearing is the method to localize objects with active or passive ranging.

Various electronic warfare systems are known in the literature. Such systems are used in various applications including detecting the location of, for example, enemy radar devices. Such electronic warfare systems normally include a plurality of antennas configured in a linear one-dimensional array. These antennas are normally connected to one or more receivers, which, in turn, are connected to processing hardware for computing AOA of intercepted enemy radar pulses [2].

To find AOA or bearing of a pulse transmitted by a RF source, for instance radar, communication transmitter or navigational aid etc., is of significant importance in the Electronic Warfare (EW) context [3]. It is one of the major parameters measured by Electronic Support Measures (ESM) among others namely frequency, dB level, Pulse Width (PW) and Time Of Arrival (TOA). Since the hostile radar cannot vary its relative bearing at a fast rate, the AOA is the most reliable parameter used in an ESM digital processor to identify and sort radar signals [4].

There are several ways to measure AOA. Measurement by narrow-beam antenna & side-lobe cancellation is the simplest technique for this purpose but it gives low Probability Of Intercept (POI). Other techniques involve amplitude comparison, multiple element phase comparison, Doppler frequency shift measurement, differential TOA measurement, microwave lens and multiple beam systems [5]. In this paper we propose a system (Figure 1) that measures AOA by amplitude comparison.

The rest of the paper is organized as follows: In Section II, we will introduce the concept of proposed system. In Section III, the formulae for amplitude comparison technique are derived with reference to the responses of the available mono-pulse antennas and their DLVA circuits. The details of the amplitude comparison hardware (ACH) and the graphical display are given in Section IV. Section V presents the preliminary results of the experimental evaluation of the system. Section VI discusses the conclusion and future recommendations to improve the proposed system.

Manuscript received August 29, 2010.

J. Iqbal is with the ADVanced Robotics (ADVR) Department, Italian Institute of Technology (IIT) and University of Genova, Italy (phone: +39-010-71781481; fax:+39-010-720321; e-mail: jamshed.iqbal@iit.it).

S. M. Khan is with the School of Information Technology & Engineering (SITE), Ottawa Carleton Institute for Electrical and Computer Engineering, Canada (email: salman9201@yahoo.com)

F. Masood is a Senior Design Engineer in Powersoft19 Inc. (e-mail: fmasood@powersoft19.com).

II. System Concept

The system consists of six monopulse antennas labeled A to F with their bore-sights separated by 60° in azimuth. Each antenna feeds the received signal to a receiver. Each receiver consists of a Band-Pass Filter (BPF), an amplifier and a Detector Log Video Amplifier (DLVA). The DLVA output signal is the envelope of the input pulsed microwave signal and the output voltage is proportional to the input power level. The six video signals (output of DLVA's) are applied to ACH. Following the successful simulation in Altium Inc. PCAD's (Computer Aided Design) Mixed Signal Circuit Simulator, the prototype hardware was fabricated. The ACH outputs two maximum amplitude values and a binary pair-code corresponding to the antenna-receiver pair which gives the maximum amplitude values. Using these two maximum values and the pair-code, the AOA of the pulse is calculated. The AOA information is shown on tactical graphical display for the whole 2π azimuth range. An ESM system can use this bearing information to sort the radar pulse, and the maximum amplitude value for scan pattern analysis.

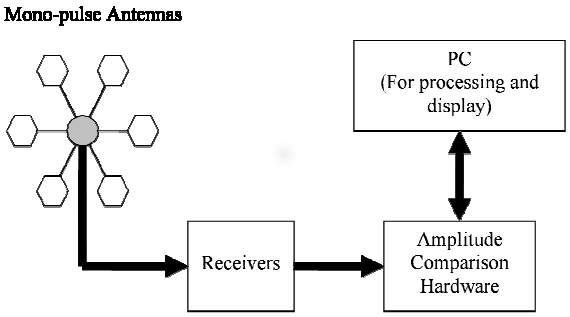


Fig. 1. Bearing measurement system based on amplitude comparison technique

III. Amplitude Comparison Technique

The basic principle of amplitude comparison technique for finding the bearing that the antennas directed towards the incoming pulse of a transmitter will generate the strongest output from the receivers connected to them. By comparing these amplitude values, the AOA of the pulse can be estimated. The system presented in this paper has six monopulse directional antennas with their bore-sights separated by 60° to cover the full 360° azimuth range (Figure 2). The set of antenna spacings is selected to minimize the probability that this algorithm produces an ambiguity error in the estimated AOA. The alternate antennas are grouped as ACE (group 1) and BDF (group 2). In each group the three corresponding amplitude values (outputs of DLVA's) are compared to find the maximum amplitude value. If the two maximum amplitude values are not from consecutive antennas, an error signal is generated. Otherwise a pair-code is generated to indicate the antenna pair giving the maximum amplitude signals. We define bearing (θ) as the sum of coarse bearing (θ_c) and fine bearing (θ_f):

$$\theta = \theta_c + \theta_f \quad (1)$$

where

$\theta_c \in \{60°, 120°, 180°, \ldots, 360°\}$

$\theta_f \in [-30, 30]$

The pair-code gives the coarse bearing (θ_c) with 60° resolution. For example, if antennas B and C give the maximum amplitude signals then the coarse bearing (θ_c) will be 120° from North. The two maximum amplitude values from a pair of antennas determine fine bearing (θ_f). Henceforth a relation will be derived to obtain θ_f from the two maximum amplitude values.

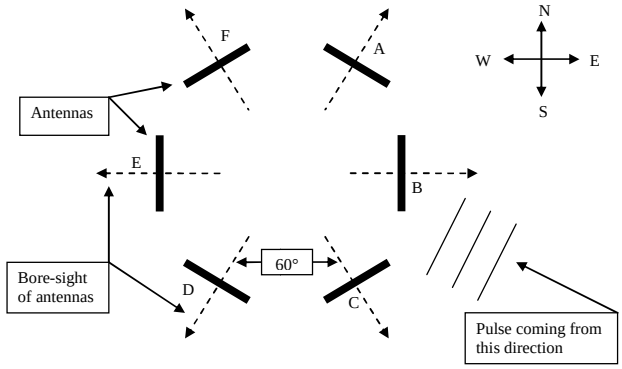


Fig. 2. Six Monopulse directional antennas

Consider a pair of antennas (Figure 3) which give maximum amplitudes for a pulse whose AOA falls within their bore-sight. The bore-sights of the antennas are shown with outward arrows and the AOA of the pulse is shown with inward arrows. The θ_f is indicated in the figure.

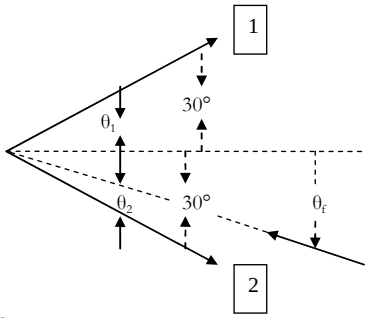


Fig. 3. Fine bearing

Using Gaussian approximation, we can write following equations for the power received (P_1 and P_2) by antennas 1 and 2 respectively:

$$P_1 = K \exp[-a*(\theta_1)^2] \quad (2)$$

$$P_2 = K \exp[-a*(\theta_2)^2] \quad (3)$$

Here θ_1 and θ_2 are off-bore-sight angles for antennas 1 and 2 respectively. Same values of constants 'K' and 'a' are used for both the antennas since same amount of power is incident on both of them and it is assumed that the antennas

are identical. Expressing P_1 and P_2 in dBm, (2) and (3) can be written as:

$$P_{1(dBm)} = K_1 - K_2(\theta_1)^2 \quad (4)$$

$$P_{2(dBm)} = K_1 - K_2(\theta_2)^2 \quad (5)$$

Here K_1 and K_2 are constants. Since the output voltage level of a DLVA is proportional to the input power level in dBm, therefore using equations (4) and (5), the output voltages from the DLVA's of antennas 1 and 2 can be written as:

$$V_1 = C_1 - C(\theta_1)^2 \quad (6)$$

$$V_2 = C_2 - C(\theta_2)^2 \quad (7)$$

C_1, C_2 and C are constants. C_1 and C_2 are not equal due to different responses of the two receiver paths. The difference in C_1 and C_2 can be catered for by calibrating all the six receivers. Experimentally it was verified that the receiver output voltage is related to the off-bore-sight angle of the antenna by equations like (6) and (7). In Figure 4, normalized output voltages are plotted against the off-bore sight angle for all the six antenna-receiver pairs. The equations were obtained using 2nd order polynomial curve fitting.

$$V_a = 4 - 3.2 \times 10^{-4} (\theta - 2)^2 \quad (8a)$$

$$V_b = 4 - 3.5 \times 10^{-4} (\theta - 1)^2 \quad (8b)$$

$$V_c = 4 - 3.6 \times 10^{-4} (\theta - 2)^2 \quad (8c)$$

$$V_d = 4 - 3.8 \times 10^{-4} (\theta - 1)^2 \quad (8d)$$

$$V_e = 4 - 3.7 \times 10^{-4} (\theta - 0)^2 \quad (8e)$$

$$V_f = 4 - 3.5 \times 10^{-4} (\theta - 2)^2 \quad (8f)$$

Where θ is the off-bore-sight angle in degrees. The average response was obtained by the following equation:

$$V_{avg}(\theta) = \frac{1}{6} \sum_{x=a}^{f} V_x(\theta) \quad (9)$$

Where x∈ {a,b,c,…,f}.

The normalized equation for V_{avg} comes out to be:

$$V_{avg} = 4 - 3.55 \times 10^{-4} (\theta - 1)^2 \quad (10)$$

The average value of C is 3.55 x 10^{-4} Volts/degree. Also there is an average drift error of 1° that can be ignored.

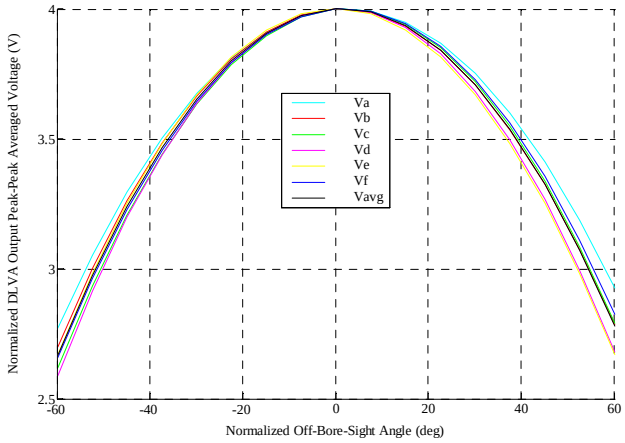


Fig. 4: Normalized response of the six mono-pulse antenna-receiver front end

Assuming that the calibration has been done and subtracting (6) from (7), the following equation is obtained:

$$V_2 - V_1 = -C(\theta_2^2 - \theta_1^2) \quad (11)$$

Putting $\theta_1 = 30° + \theta_f$ and $\theta_2 = 30° - \theta_f$ as shown in Figure 3, (11) can be written as:

$$V_2 - V_1 = (120C)\,\theta_f$$

$$\theta_f = C_0 (V_2 - V_1) \quad (12)$$

Where

$C_0 = 1/(120C) = 1/(120 \times 3.55 \times 10^{-4}) = 23.5$ deg/Volt

Equation (12) relates the fine bearing with the two maximum amplitude values.

IV. Amplitude Comparison Hardware

A prototype ACH has been developed for the comparison and digitization of the six video signals. The ACH primarily consists of comparators, analogue multiplexers (MUX), Analog to Digital Converters (ADC), and latches (Figure 5). Three comparators and MUX select the maximum amplitude signal in a group. The analogue MUX makes it possible to use one ADC in each group thereby making the design compact as well as efficient. The maximum amplitude signals from the two groups are then digitized and latched. The output bits of comparators are then latched as the pair-code. The triggering signal for the ADC's is generated by a threshold comparison sub-circuit (not shown in the figure).

Now the coarse bearing can be obtained by the pair-code and the fine bearing through (12). Then using (1) bearing can be calculated. This procedure of calculating bearing from the maximum amplitude values and the pair-code can be implemented by either using Programmable Read Only Memory (PROM) or employing software operations. For the system testing, PROM approach was used.

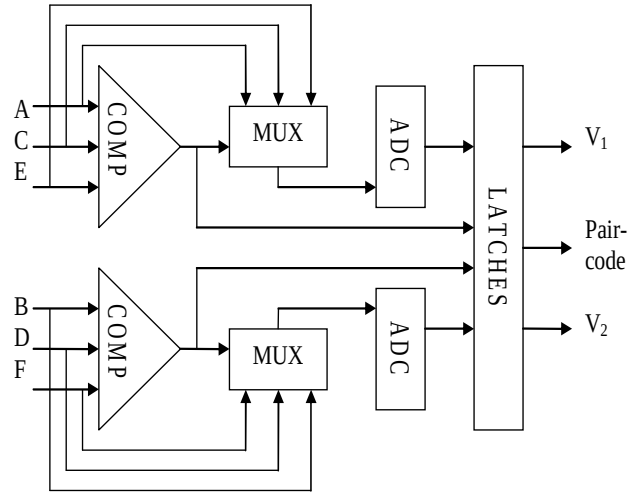


Fig. 5: Amplitude Comparison Hardware

A software program was written in C++ to read the bearing and the maximum amplitude data into the PC through its parallel port. The program graphically displays the bearing and dB level information with a refresh rate of 100ms. The graphical display is shown in Figure 6.

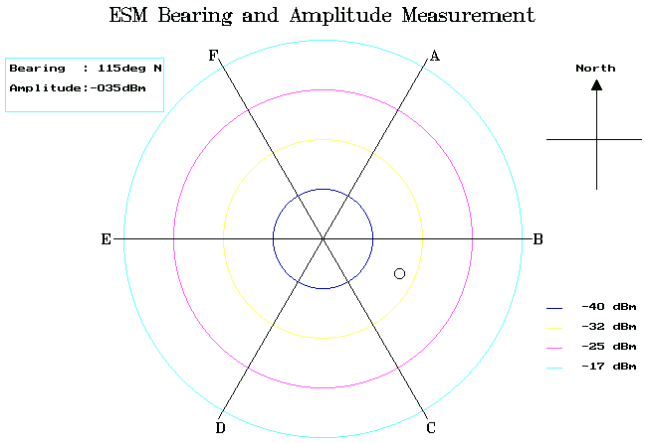


Fig. 6: Graphical display

V. Results

To evaluate the system performance, an experimental setup (Figure 7) was established. Using an RF generator and a transmitting antenna, a pulsed transmission was simulated. The transmitter was placed at different bearings from 90° to 150°. The corresponding RF signals received by the mono-pulse antennas were then processed. The corresponding bearing values displayed by the system were noted.

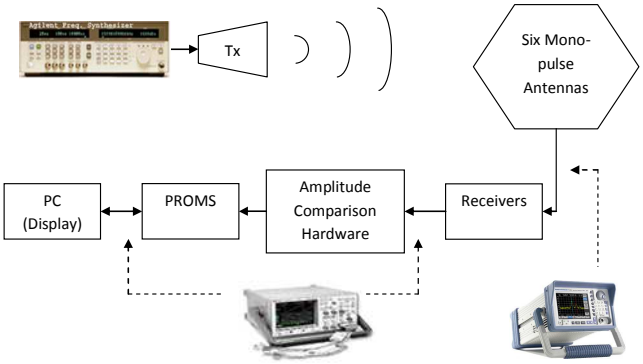


Fig. 7: Experimental setup for system evaluation

The results of actual and measured bearing are tabulated in Table I while Figure 8 shows the corresponding results in graphical form.

TABLE I
EXPERIMENTAL RESULTS
(ALL VALUES ARE IN DEGREES)

Actual Bearing (Ba)	Averaged Measured Bearing (Bm)	Bearing Error (Bm – Ba)
90	96	+6
98	98	0
105	108	+3
112	114	+2
120	115	-5
128	122	-6
135	126	-9
142	136	-6
150	144	-6
Standard Deviation =		5.7

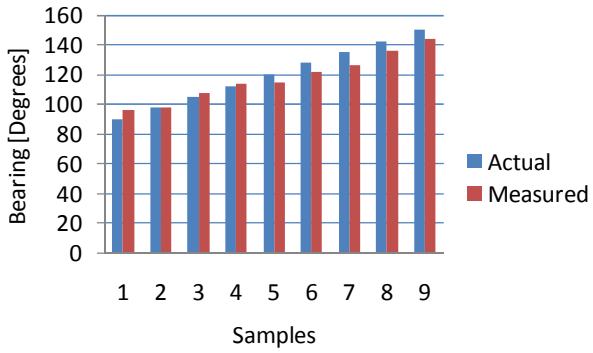


Fig. 8: Results in graphical form

VI. Conclusion

The results show that the error in bearing calculation usually doesn't exceed 6°. The main source of error is the difference in the response of the antenna-receiver front ends. This error can be removed by using antenna offset values and employing calibration scheme for the receivers. The value of C in (10) needs to be calculated with more sophisticated experimentation as the fine bearing critically depends on this constant. The fluctuations at the outputs of

the DLVA's are another source of error. This problem can be addressed by taking average of subsequent readings.

ACKNOWLEDGMENT

We express our gratitude to Mr. Osman Haider and Mr. Muhammad Shumail Waraich for assisting us in the hardware testing. Special thanks definitely go to Imaan Fatima for her support.

REFERENCES

[1] R. Peng and M. L. Sichitiu, "Angle of arrival localization for wireless sensor networks", in Proc. of the Third Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, (Reston, VA), Sep. 2006.

[2] "Pipelined processing algorithm for interferometer angle of arrival estimation", US Patent 6377214 issued on April 23, 2002.

[3] R. C. Johnson, H. Jasik, "Antenna Applications Reference Guide", McGraw-Hill Book Company, New York, 1987.

[4] J. B. Tsui, "Microwave Receivers with Electronic Warfare Applications", Krieger Publishing Company, Florida, 2005.

[5] F. Neri, "Introduction to Electronic Defense Systems", Second Edition, SciTech Publishing, 2006.

On the Popov Criterion for a Hyper-redundant Arm Control

Mircea Ivanescu, Nicu Bizdoaca, Mihaela Florescu

***Abstract*— A frequency stability criterion based on the Kahman – Yakubovich – Popov Lemma for the hyper-redundant arms with continuum element that performs the grasping function by coiling is discussed. The P control algorithms are proposed. The dynamic model of the continuum arm for the position control during non-contact and contact operations with the environment is studied. An extension of the Popov criterion is developed. The control algorithms based on SMA actuators are introduced. Numerical simulations and experimental results of the arm motion toward an imposed target are presented.**

I. INTRODUCTION

The control of the hyper-redundant robots is very complex, indeed, and a large number of researchers have tried to cater solutions. In [2], Gravagne analyzed the kinematic model of "hyper-redundant" robots, known as "continuum" robots. Remarkable results were achieved by Chirikjian and Burdick [6], who laid the foundations of the kinematic theory of hyper-redundant robots. Their findings are based on a "backbone curve" that captures the robot's macroscopic geometric features. Mochiyama investigated the problem of controlling the shape of an HDOF rigid-link robot with two-degree-of-freedom joints using spatial curves [7]. In [8], the "state of art" of continuum robots, their areas of application and some control issues are presented. Other papers [9, 10] deal with several technological solutions for actuators used in hyper-redundant structures and with conventional control systems.

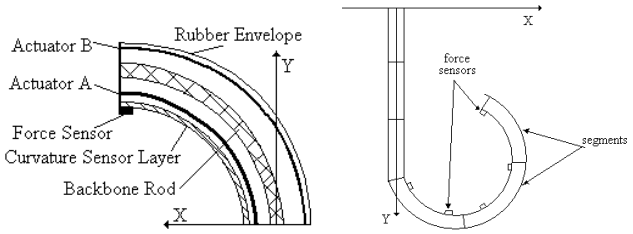
The current paper investigates the control problem of a class of hyper-redundant arms with continuum elements that performs the grasping function by coiling. The dynamics of the arm during non-contact or contact operations with the environment are analyzed. The frequency criteria for the stability and control algorithms are also discussed. The paper is organized as follows: Section II presents technological and theoretical preliminaries; Section III studies the dynamics of the arm and load in a grasping function; Section IV presents an extension of the Popov criterion for this class of systems; Section V verifies the control laws by computer simulation; Section VI presents some experimental results.

II. TECHNOLOGICAL SYSTEM

The hyper-redundant technological models are complex structures that operate in 3D space, but the grasping function of these arms is, generally, a planar function. Accordingly, the model discussed in this paper is a 2D model.

The technological model basis is presented in Fig.1.a. It consists of layered structures that ensure the flexibility, driving and position measuring. The high flexibility is obtained by an elastic backbone rod.

The driving layer is made up of two antagonistic SMA actuators, A and B, each of them having a number of SMA fibers that are connected to the ends of the beam and determine its bending by current control. These SMA fibers are well suited for grasping force control due to their high strength to weight ratio.



(a) (b)

Fig.1. (a) The segment layer structure; (b) The arm structure

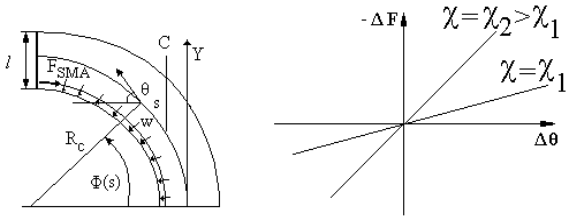
The measuring layer is represented by an electro-active polymer curvature sensor. This sensor is placed on the boundary of the beam and allows for its curvature measuring by the resistance measuring. The sensor system is completed by a number of force sensors placed at each terminal of the beam segment. A rubber envelope protects and isolates this layer structure from the operator environment.

The general form of the arm is shown in Fig.1.b. It consists of a number (N) of segments and the last m segments $(m < N)$ represent the grasping terminals.

As a theoretical model, we shall consider the beam in Fig.2.a with the length L and the thickness l. This beam has been deflected into a circular arc by a SMA fiber. The beam is composed of concentric arcs. The neutral arc defines the curvature of the beam,

$$c_V = \frac{d\phi}{ds},\ \phi = \frac{s}{R_c} \tag{2.1}$$

where ϕ represents the angle of the current position, s is the arc length from the origin, and R_c is the radix of the arc.



(a) (b)

Fig.2. (a) The arm parameters; (b) The force variation

We denote the equivalent force developed by the SMA actuators at the end of the beam $(s = L)$ by χ, the force density and the distributed force along the beam exercised by the SMA fibers on the beam surface by *w* and *F*, respectively, and τ is the equivalent moment of the beam.

From [11], we have the following relations

$$w = \frac{dF}{ds},\ w = \chi \cdot c_V,\ \tau = \chi \cdot \frac{l}{2} \tag{2.2}$$

$$dF = \chi \cdot d\phi,\ dF = -\chi \cdot d\theta \tag{2.3}$$

III. GRASPING DYNAMIC MODEL

The grasping function control is represented by the force control between the arm and load. Consider that the arm has achieved the desired position defined by the surface (object).

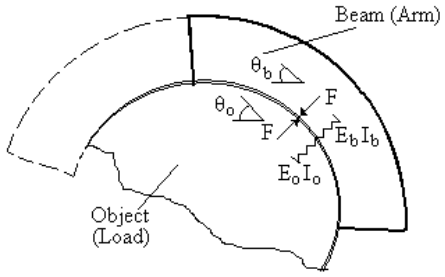


Fig.3. The grasping model

In Fig.3, an object with elastic and damping parameters E_oI_o, b_o and c_o, respectively, is grasped by coiling. Using the same procedure as developed in [16], the dynamic model of the two bodies in contact, arm and load, is represented by the following partial differential equations,

$$\dot{\tilde{e}} = \tilde{A}\frac{\partial^2 \tilde{e}}{\partial s^2} + \tilde{B}\tilde{e} + \tilde{c}f \tag{3.1}$$

$$\tilde{e}(0,s) = 0;\ \tilde{e}(t,0) = \frac{\partial \tilde{e}(t,0)}{\partial s} = 0 \tag{3.2}$$

$$E_bI_b \cdot \frac{\partial e_b(t,L)}{\partial s} = \tau^* \tag{3.3}$$

$$e_b(t,L) = e_\theta(t,L);\ \frac{\partial e_o(t,L)}{\partial s} = \frac{\partial e_b(t,L)}{\partial s} \tag{3.4}$$

$$\frac{\partial \dot{\tilde{e}}(t,L)}{\partial s} = -\alpha_1 \tilde{e}(t,L) - \alpha_2 \dot{\tilde{e}}(t,L) \tag{3.5}$$

where $\tilde{e}(t,s) = \left[e_b(t,s),\ \ e_o(t,s)\right]^T$, with $e_b(t,s) = \theta_b(s) - \theta_{bd}(s);\ e_o(t,s) = \theta_o(s) - \theta_{od}(s)$, *f* is the force error, τ^* is the control variation, and the indices *b* and *o* specify the parameters of the beam and object, respectively.

$$\tilde{A} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{1}{I_{\rho b}}\left(E_bI_b + E_oI_o\right) & 0 & -\frac{E_oI_o}{I_{\rho b}} & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{E_oI_o}{I_{\rho o}} & 0 & \frac{E_oI_o}{I_{\rho o}} & 0 \end{bmatrix} \tag{3.6}$$

$$\tilde{B} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{b_b}{I_{\rho b}} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -\frac{b_b}{I_{\rho o}} \end{bmatrix};\qquad \tilde{c} = \begin{bmatrix} 0 \\ \frac{c_b}{I_{\rho b}} \\ 0 \\ -\frac{c_o}{I_{\rho o}} \end{bmatrix} \tag{3.7}$$

IV. CONTROL ALGORITHM

The grasping force control is the second problem of the grasping control. A force sensor network is used to account for the contact between the arm and the load. We notice, from (2.2), that the force density *w* is constant along the arm segment and, in a steady state, *w* can be approximated to *f*. A force sensor with the position $s = s^* \in [0, L]$ is used to measure the contact force. The contact force – displacement relation of the sensor is assumed to lie in the positive sector (Fig.4).

$$\Delta F = -\psi(\Delta\theta),\ \psi(\Delta\theta) \cdot \Delta\theta \geq 0 \tag{4.1}$$

$$\psi(0) = 0 \ \text{ for } \ \Delta\theta = 0 \tag{4.2}$$

The nonlinearity $\psi(\Delta\theta)$ is single-valued, time invariant and constraint to a sector bounded by slope k_s which is assumed to meet

$$0 \le \frac{\psi(\Delta\theta)}{\Delta\theta} \le k_s < \infty \tag{4.3}$$

which is the case for most physically realistic elastic contacts.

In terms of the sensor characteristics, the convergence to zero of the error e_b is equivalent to the convergence to zero of the contact force error *f*

$$\lim_{t\to\infty} e_b(t, s^*) = 0 \;\Rightarrow\; \lim_{t\to\infty} f(t, s^*) = 0 \tag{4.4}$$

The sensor nonlinearity (4.1) – (4.3) and the dynamic model of the grasping contact described by (3.1) – (3.5) suggest the closed – loop system of Fig.4.

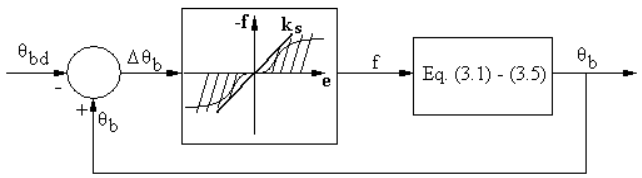


Fig.4. The grasping control closed loop system

Theorem. The closed – loop system (Fig.4) is absolutely stable if:

(1) $\left(-\tilde{A}+\tilde{B}\right)$ is a Hurwitzian matrix;

(2) the pair $\left(-\tilde{A}+\tilde{B},\quad \tilde{c}\right)$ is completely controllable;

(3) there is a positive definite and symmetrical matrix *P* such that $\left(\tilde{A}^T P + P\tilde{A}\right)$ is positive definite;

$$(4)\ \frac{1}{k_s} + \mathrm{Re}\left[\tilde{n}^T\left(j\omega I - \left(-\tilde{A}+\tilde{B}\right)\right)^{-1}\tilde{c}\right] \ge 0 \tag{4.5}$$

(5) the moment of the arm verifies the relation

$$\tau^* = k_p e_b(t, L), \qquad k_p > E_b I_b \tag{4.6}$$

Proof. See Appendix.

Equations (4.1), (4.3) and (3.1) – (3.5) describe the closed loop system (Fig.4), consisting of a partial derivative equation linear system and a nonlinear element represented by the function $\psi(\cdot)$ belonging to the sector $[0, k_s]$. In this case, the condition (4) represents the Popov criterion for this class of systems. According to it, the system will be absolutely stable if the plot of $\tilde{G}(j\omega)$

$$\tilde{G}(j\omega) = \tilde{n}^T\left[j\omega I - \left(-\tilde{A}+\tilde{B}\right)\right]^{-1}\tilde{c} \tag{4.7}$$

crosses the negative real axis at a point that lies to the right of the critical point defined by $-\frac{1}{k_s}$ (Fig.5).

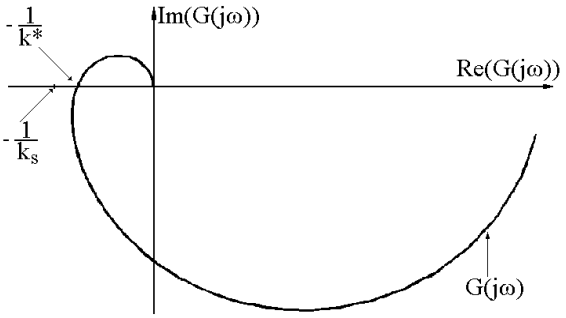


Fig.5. The plot of $G(j\omega)$ for the grasping control

For a pair "arm – load" specified, the plot of $\tilde{G}(j\omega)$ has a well-defined characteristic and the intersection with the real axis determines the limit value of *k*. Let k^* be the corresponding value of the crossing point. The absolute stability is guaranteed if the sensor parameters meet the condition

$$k_s \le k^* \tag{4.8}$$

V. Simulation

A hyper-redundant manipulator with 4 segments is considered. The parameters of the arm were selected run as follows: bending stiffness $E_b I_b = 1$, linear mass density $\rho_b = 0.5\,kg/m$, rotational inertial density $I_{\rho b} = 0.001 kg \cdot m^2$ and damping ratio 0.35. These constants are realistic for long thin backbone structures. The grasping function is exercised by the last three segments of the arm, the length of each arm segment is $L = 1$ (Geometrical parameters are scaled.). The load is a cylinder with the radix $R = 1$, bending stiffness $E_o I_o = 0.2$, rotational inertial density and damping ratio 0.22.

Fig.6 illustrates the grasping function of the arm. The initial position is a vertical one, and the arm motion by coiling of the arm can be seen.

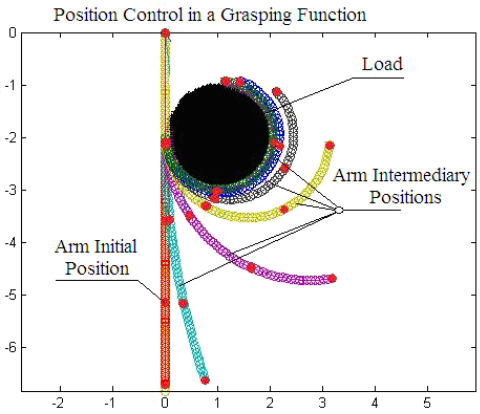


Fig.6. The simulation of the grasping operation

Fig.7 shows the frequency plots of $G(j\omega)$ and $\tilde{G}(j\omega)$ for the position and force control, respectively. The plot of $G(j\omega)$ crosses the negative real axis at -0.14, which imposes the limit of tension at $T^* = 7.15N$; the plot of $\tilde{G}(j\omega)$ crosses the negative real axis at -0.74, which corresponds to the critical value of the force sector at $k^* = 1.3$.

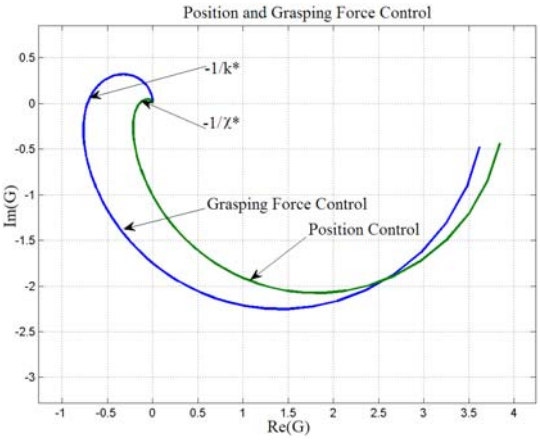


Fig.7. The plot of $G(j\omega)$ for position and force control

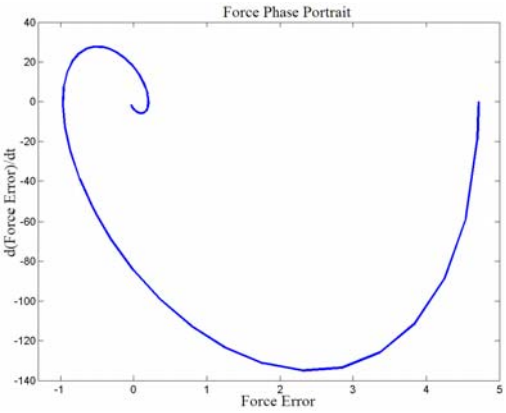


Fig.8. The force phase portrait

A P – control (4.5) with $k_p = 24$ is applied and the force phase portrait is illustrated in Fig.8. Please note the convergence to zero of the force error, but, also, the transient response of the system determined by the P – control law and a low damping factor of the system.

VI. Experimental Results

In order to verify the suitability of the control algorithm, a planar continuum terminal arm consisting by a layer structure has been employed for testing (Fig.9). The arm consists of two $(25\times 6\times 4mm)$ continuum segments with an elastic backbone rod. The two antagonistic SMA actuators ensure the actuation system. Each actuator consists of G fibers in parallel. A polymer thick film layer which exhibits a decrease in resistance with an increase of the film curvature is used. Also, a Force Sensing Resistor is used at the end of each segment.


Fig.9. Experimental platform

A Quancer based platform is used for control and signal acquisition. The load is represented by a sphere ball with $R_c = 0.02m$. A P-control with $k_p = 2.17$, $T_D = 5\text{s}$, $T_P = 7\text{s}$ is implemented. The contact force in the grasping operation is represented in Fig.10.

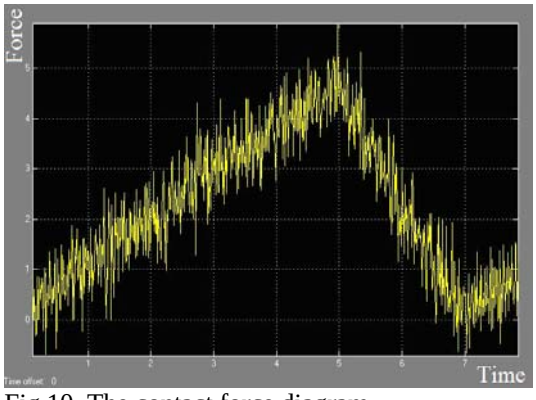


Fig.10. The contact force diagram

VII. Conclusions

The paper treats the control problem of a hyper-redundant robot with continuum elements that performs the coil function for grasping. First, the dynamic model of continuum arm for the position control during non-contact operations with environment is studied and a frequency stability criterion based on KYP Lemma is introduced. The P control algorithms are proposed. Then, the grasping control problem for the arm in contact with a load is analyzed. The dynamics of the system are discussed and an extension of the Popov criterion is proposed. The control algorithms based on SMA actuators are introduced. Numerical simulations and experimental results of the arm motion toward an imposed target prove the correctitude of the solutions.

VIII. Acknowledgement

This work was supported by CNCSIS –UEFISCSU, project number PNII – IDEI code 289/2008, "Ingineria inversa in modelarea cognitiva si controlul structurilor biomimetice", director contract prof. Nicu Bizdoaca, Ph.D., University of Craiova, Romania.

Appendix

Consider the following Lyapunov functional,

$$V = \int_0^L \tilde{e}^T P \tilde{e} ds \tag{A.1}$$

where P is a (2×2), symmetrical and positive definite matrix. The derivative of this functional will be

$$\dot{V} = \int_0^L \left[\begin{array}{l} \left(\dfrac{\partial^2 \tilde{e}^T}{\partial s^2} \tilde{A}^T P \tilde{e} + \tilde{e}^T P \tilde{A} \dfrac{\partial^2 \tilde{e}}{\partial s^2} \right) + \\ + \left(\tilde{e}^T B^T P \tilde{e} + \tilde{e}^T P B \tilde{e} \right) + 2 \tilde{e}^T P c f \end{array} \right] ds \tag{A.2}$$

By using the relation

$$\frac{\partial^2 \tilde{e}^T}{\partial s^2} \tilde{A}^T P \tilde{e} = \frac{\partial}{\partial s} \left(\frac{\partial \tilde{e}^T}{\partial s} \tilde{A}^T P \tilde{e} \right) - \frac{\partial \tilde{e}^T}{\partial s} \tilde{A}^T P \frac{\partial \tilde{e}}{\partial s}, \tag{A.3}$$

the derivative $\dot{V}$ will be

$$\begin{array}{l} \dot{V} = \displaystyle\int_0^L \left[\begin{array}{l} -\left(\dfrac{\partial \tilde{e}^T}{\partial s} \left(\tilde{A}^T P + P \tilde{A} \right) \dfrac{\partial \tilde{e}}{\partial s} \right) + \\ + \tilde{e}^T \left(B^T P + P B \right) \tilde{e} + 2 \tilde{e}^T P c f \end{array} \right] ds + \\ + \left(\tilde{e}^T \left(\tilde{A}^T P + P \tilde{A} \right) \dfrac{\partial \tilde{e}}{\partial s} \right) \Big|_0^L \end{array} \tag{A.4}$$

By using the inequality [14] and the condition (3) (Theorem 1)

$$\begin{array}{l} \displaystyle\int_0^L \frac{\partial \tilde{e}^T}{\partial s} \left(\tilde{A}^T P + P \tilde{A} \right) \frac{\partial \tilde{e}}{\partial s} ds \geq \int_0^L \tilde{e}^T \left(\tilde{A}^T P + P \tilde{A} \right) \tilde{e} ds - \\ - \left(\tilde{e}^T \left(\tilde{A}^T P + P \tilde{A} \right) \tilde{e} \right) \Big|_0^L \end{array}, \tag{A.5}$$

$$\begin{array}{l} \dot{V} \leq \displaystyle\int_0^L \left[\begin{array}{l} \tilde{e}^T \left(\left(-\tilde{A} + B \right)^T P + P \left(-\tilde{A} + B \right) \right) \tilde{e} + \\ + 2 \tilde{e}^T \left(P c - \dfrac{1}{2} n \right) \cdot f + \tilde{e}^T n f \end{array} \right] + \\ + \left(\tilde{e}^T \left(\tilde{A}^T P + P \tilde{A} \right) \left(\tilde{e} + \dfrac{\partial \tilde{e}}{\partial s} \right) \right) \Big|_0^L \end{array} \tag{A.6}$$

By using the conditions of the Yakubovich – Kalman – Popov Lemma

$$\left(-\tilde{A} + \tilde{B} \right)^T P + P \left(-\tilde{A} + \tilde{B} \right) = -Q Q^T \tag{A.7}$$

$$P \tilde{c} - \frac{1}{2} \tilde{n} = \sqrt{k_s^{-1}} Q \tag{A.8}$$

where $\tilde{n} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^T$, and the sector inequality (4.1), (4.3), we obtain

$$\begin{array}{l} \dot{V} \leq \displaystyle\int_0^L \left(-\tilde{e}^T Q Q^T e + 2 \tilde{e}^T \sqrt{k_s^{-1}} Q f - k_s^{-1} f^2 \right) ds + \\ + \left(\tilde{e}^T \left(\tilde{A}^T P + P \tilde{A} \right) \left(\tilde{e} + \dfrac{\partial \tilde{e}}{\partial s} \right) \right) \Big|_0^L \end{array} \tag{A.9}$$

$$\dot{V} \leq \int_0^L \left(-\tilde{e} Q - \sqrt{k_s^{-1}} f \right)^2 - \tilde{e}_L^T \left(\tilde{A}^T P + P \tilde{A} \right) \tilde{K} \tilde{e}_L^T \tag{A.10}$$

where $\tilde{K}$ is a positive definite matrix and $\tilde{e}_L = \tilde{e}(t, L)$.

Using the boundary conditions (3.1) – (3.5), and the conditions (5) of Theorem,

$$\dot{V} \leq 0 \qquad \text{(Q.E.D.)} \tag{A.11}$$

References

[1] Hemami, A., *Design of light weight flexible robot arm*, Robots 8 Conference Proceedings, Detroit, USA, June 1984, pp. 1623-1640.

[2] Gravagne, Ian A., Walker, Ian D., *On the kinematics of remotely - actuated continuum robots*, Proc. 2000 IEEE Int. Conf. on Robotics and Automation, San Francisco, April 2000, pp. 2544-2550.

[3] Gravagne, Ian A., Walker, Ian D., *Kinematic Transformations for Remotely-Actuated Planar Continuum Robots*, Proc. 2000 IEEE Int. Conf. on Rob. and Aut., San Francisco, April 2000, pp. 19-26.

[4] Gravagne, Ian A., Rahn, Christopher D., Walker, Ian D., *Good Vibrations: A Vibration Damping Setpoint Controller for Continuum*

Robots, Proc. 2001 IEEE Int. Conf. on Robotics and Automation, May 21-26, 2001, Seoul, Korea, pp. 3877-3884.

[5] Gravagne, Ian A., Walker, Ian D., *Uniform Regulation of a Multi-Section Continuum Manipulator*. Proc. IEEE Int. Conf. on Rob. and Aut, Washington, A1-15, May 2002, pp. 1519-1524.

[6] Chirikjian, G. S., Burdick, J. W., *An obstacle avoidance algorithm for hyper-redundant manipulators*, Proc. IEEE Int. Conf. on Robotics and Automation, Cincinnati, Ohio, May 1990, pp. 625 - 631.

[7] Mochiyama, H., Kobayashi, H., *The shape Jacobian of a manipulator with hyper degrees of freedom*, Proc. 1999 IEEE Int. Conf. on Robotics and Automation, Detroit, May 1999, pp. 2837-2842.

[8] Robinson, G., Davies, J.B.C., *Continuum robots – a state of the art*, Proc. 1999 IEEE Int. Conf. on Rob and Aut, Detroit, Michigan, May 1999, pp. 2849-2854.

[9] Ivanescu, M., Stoian, V., *A variable structure controller for a tentacle manipulator*, Proc. IEEE Int. Conf. on Robotics and Aut., Nagoya, 1995, pp. 3155-3160.

[10] Ivanescu, M., Florescu, M.C., Popescu, N., Popescu, D., *Position and Force Control of the Grasping Function for a Hyperredundant Arm*, Proc.of IEEE Int. Conf.on Rob. and Aut., Pasadena, California, 2008, pp. 2599-2604.

[11] Camarillo, D., Milne, C., *Mechanics Modeling of Tendon – Driven Continuum Manipulators*, IEEE Trans. On Robotics, vol. 24, no. 6, December 2008, pp. 1262 – 1273.

[12] Wongratanaphisan, T., Cole, M., *Robust Impedance Control of a Flexible Structure Mounted Manipulator Performing Contact Tasks*, IEEE Trans. On Robotics, vol. 25, no. 2, April 2009, pp. 445 – 451.

[13] Grant, D., Hayward, V., *Constrained Force Control of SMA Actuators*, Proc. ICRA 2000, San Francisco, pp. 1314 – 1320.

[14] Mihlin, S. G., *Variationnie Metodi b Matematiceskvi Fizike*, Nauka, Moscva, 1970 (Russian).

[15] Slotine, J.J., Weiping Li, *Applied Nonlinear Control*, Prentice-Hall International Editions, 1991.

[16] Ivanescu, M., Bizdoaca, N., Florescu, M.C., Popescu, N., Popescu, D., *Frequency Criteria for the Grasping Control of a Hyper-redundant Robot*, Proc.of IEEE Int. Conf.on Rob. and Aut., Anchorage, Alaska (ICRA 2010), May 3 – 8, 2010, pp. 1542-1549.

On Stabilization of Control Systems Containing Non-Coulomb Friction Nonlinearities

Alexander Kamachkin and Alexander Stepanov

***Abstract*— Sufficient conditions are deduced for existing of stabilizing control for systems containing relay nonlinearities describing non-Coulomb friction effect. The cases of stationary and perturbed systems are considered. For periodic solutions of the stationary systems sufficient conditions for asymptotic orbital stablity are obtained in development of results of S.V. Zubov.**

I. INTRODUCTION

Problems of stabilization and determining of stablility characteristics of steady-state regimes are among the central in a control theory. Especial difficulties can be met when dealing with the systems containing nonlinearities which are nonanalytic function of phase. Different models describing friction in real control systems (e.g. servomechanisms, such as various servo drives, autopilots, stabilizers etc.) are just concern this type. Friction appears in numerous control systems and is important due to its significant impact on it. An accurate account of a friction effect allows to receive adequate models of behaviour of such a systems.

Numerous works are devoted to the analysis of problem of stable oscillations presence in such systems (see, for example, [1], [2], [3], [4], [5]).

This article consists of two parts. In the first section a formal algorithm is given for construction of stabilizing control for the system containing non-Coulomb friction nonlinearity in development of results cited by [6], [8], [7]. The another contains some results concerning to a stability problem for periodic solutions of the controlled system (see [7], [4]).

The main results obtained below might generally be put in connection with classical results of V.I. Zubov's control theory school [6], [7]. Note that all examples presented in the article are purely illustrative; some examples concerning to similar systems can be found in [4], [5].

II. MODELS UNDER CONSIDERATION

Let us consider the following system:

$$\dot{x} = A\,x + c\,u, \quad u(t) = f\left(\sigma(t)\right), \quad \sigma(t) = \gamma' x(t), \quad (1)$$

or, in more general case,

$$\dot{x} = A\,x + \varphi(t) + c\,u, \quad (2)$$

here $x, c, \gamma \in \mathbb{E}^n$, $\|\gamma\| \neq 0$; A is real $n \times n$-matrix; $\varphi(t) \in \mathbb{E}^n$ is continuous bounded function of t, $t \geq t_0$, $|\varphi| \leq \hat{\varphi}$; nonlinearity f describes the effect of non-Coulomb friction ([4]):

$$f(\sigma) = \begin{cases} 0, & \begin{cases} |\sigma| \leq \sigma_0, \\ |\sigma| \in (\sigma_0;\ l\,], & f_- = 0; \\ \sigma \in (\sigma_0;\ l\,], & f_- > 0, \\ \sigma > l; \end{cases} \\ \kappa\,(\sigma - \sigma_0)\,, & \begin{cases} \sigma \in [-l;\ -\sigma_0)\,, & f_- < 0, \\ \sigma < -l; \end{cases} \\ \kappa\,(\sigma + \sigma_0)\,, \end{cases} \quad (3)$$

where $\kappa > 0$, $\sigma_0 \geq 0$, $l > \sigma_0$ (see Fig. 1.A), – a so-called Bulgakov's approximation of dry friction [5]; or

$$f(\sigma) = \begin{cases} 0, & \begin{cases} |\sigma| \leq l_0, \\ |\sigma| \in (l_0;\ l\,], & f_- = 0; \\ \sigma \in (l_0;\ l\,], & f_- > 0, \\ \sigma > l; \end{cases} \\ \kappa\,(\sigma - \sigma_0)\,, & \begin{cases} \sigma \in [-l;\ -l_0)\,, & f_- < 0, \\ \sigma < -l; \end{cases} \\ \kappa\,(\sigma + \sigma_0)\,, \end{cases} \quad (4)$$

here $\sigma_0 < l_0 < l$ (see Fig. 1.B), and at the moment t

$$f_- = f\left(\sigma(t-0)\right).$$

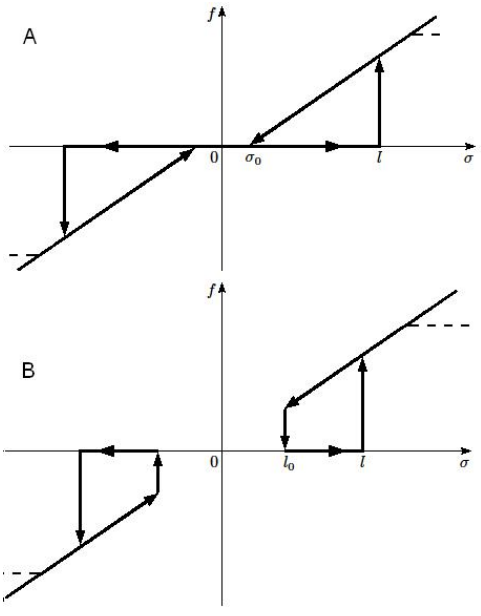


Fig. 1.

This work was not supported by any organization

A. Kamachkin is with Faculty of Applied Mathematics and Control Processes, Saint-Petersburg State University, Universitetskii prospekt 35, Petergof, Saint-Petersburg, Russia, akamachkin@mail.ru

A. Stepanov is with Synopsys GmbH, Saint-Petersburg office, Saint-Petersburg, Russia, stepanov17@yandex.ru

In more general case, saturation effect is accounted:

$$f(\sigma)=\begin{cases}\hat{f}(\sigma), & |\sigma|\le\beta\,l;\\ \kappa\,(\beta-\alpha)\;l, & \sigma>\beta\,l;\\ \kappa\,(\beta+\alpha)\;l, & \sigma<-\beta\,l;\end{cases}\qquad(5)$$

here $\alpha=\sigma_0/l$, $\beta>1$, and nonlinearity $\hat{f}$ has form (3) or (4).

III. STABILIZATION OF MOTIONS

Hereafter we will use concepts of stabilizing linear and nonlinear ("relay") control ([6], [7], [8]).

Let us say that linear control $u(t)=\kappa\sigma(t)=\kappa\gamma' x(t)$ is stabilizing for the system (or its motions)

$$\dot{x}=Ax+cu,$$

if $\hat{A}=A+\kappa\,c\,\gamma'$ is Hurwitz matrix (the sufficient conditions for the existing of such a control are not mentioned here because they are well-known).

Then, for given positive constants $\varepsilon>0$, $\delta\ge\varepsilon$ nonlinear control $u=f(\sigma(t))$ is said to be stabilizing for the same system (or its motions) if any solution of the system under control, starting at the initial moment of time $t=t_0$ in a δ-neighbourhood of the origin S_δ, converges into an ε-neighbourhood of the origin S_ε when $t\to+\infty$.

Theorem 1: Let there exists linear control

$$\hat{u}=\kappa\,\hat{\gamma}'\,x,$$

stabilizing for the system (1) (i.e. matrix $\hat{A}=A+\kappa c\hat{\gamma}'$ is Hurwitz), let V is positively definite matrix satisfying Lyapunov's matrix equation

$$\hat{A}'\,V+V\hat{A}=-E,\qquad(6)$$

where E is $n\times n$ identity matrix and $\lambda_{1,\,2}$ are minimum and maximum eigenvalues of the matrix V. If for given $\varepsilon>0$

$$l<\varepsilon\,\sqrt{\lambda_1/\lambda_2}\;\left(2\,\kappa\;\|c'\,V\|\right)^{-1},\qquad(7)$$

then control $f\,(\hat{\gamma}'\,x(t))$, where nonlinear function f is given by the formula (3) or (4), stabilizes motions of the system (1) for any $\delta\ge\varepsilon$.

If, moreover, for given $\beta>0$

$$\beta\,l>\delta\,\sqrt{\lambda_2/\lambda_1}\;\|\hat{\gamma}'\|\,,\qquad(8)$$

then control $f\,(\hat{\gamma}'\,x(t))$, where nonlinear function f is given by the formula (5), stabilizes motions of the system (1) for given δ and ε.

Proof: Consider the system (1):

$$\dot{x}=Ax+cu=Ax+c\hat{u}+c\;(u-\hat{u})=\hat{A}x+c\;(u-\hat{u})\,,$$

here control statement u is given by (3) or (4). Let $v(x)=x'Vx$, where matrix V is solution of (6). Let $\gamma=\hat{\gamma}$. In that case $|u-\hat{u}|\le\kappa\,l$, and

$$\begin{gathered}\dot{v}\big|_{(1)}=-\|x\|^2+2\;(u-\hat{u})\;c'\,V\,x\le\\ \le-\|x\|^2+2\,\kappa\,l\,\|c'\,V\|\;\|x\|<0,\end{gathered}\qquad(9)$$

if $\|x\|>2\,\kappa\,l\;\|c'\,V\|$. So if $\varepsilon>2\,\sqrt{\lambda_2/\lambda_1}\,\kappa\,l\;\|c'\,V\|$ then S_ε is the sought-for ε-neighbourhood of the origin containing all solutions of (1) when $t\to+\infty$ (here constant $\sqrt{\lambda_2/\lambda_1}$ is a ratio of radii of spheres circumscribed around and inscribed into a level surface $v(x)=C$ with centres in the origin). The latter inequality holds true if condition (7) is satisfied.

If nonlinearity f is given by the formula (5) then inequality (9) holds true when $|\sigma|<\beta\,l$. On the other hand, $|\sigma|<\beta\,l$ if $\|x\|<\beta\,l\;\|\hat{\gamma}\|^{-1}$. So, S_δ is the sought-for δ-neighbourhood of the origin if

$$\delta<\sqrt{\lambda_1/\lambda_2}\,\beta\,l\;\|\hat{\gamma}\|^{-1}\,.$$

The latter inequality holds true if condition (8) is satisfied. ■

Now let us pass onto the system (2):

Theorem 2: Let $\kappa,\hat{\gamma},\hat{u},V,\lambda_{1,\,2}$ are the same as in the previous theorem, let

$$\gamma=\hat{\gamma}-\mu\,c'\,V$$

where μ is positive constant. Then, if for given $\varepsilon>0$

$$\mu>\sqrt{\lambda_2/\lambda_1}\;(\kappa\,\nu\,\varepsilon)^{-1}\;(\hat{\varphi}+\kappa l)\;\|c'\,V\|-(2\,\kappa\,\nu)^{-1}\,,\qquad(10)$$

where ν is minimum eigenvalue of matrix $Vcc'V$, then control $f\,(\gamma'\,x(t))$, where nonlinear function f has form (3) or (4), stabilizes motions of the system (2) for any $\delta\ge\varepsilon$.

If, moreover, for given $\beta>0$

$$\mu<\left(\sqrt{\lambda_1/\lambda_2}\,\delta^{-1}\,\beta\,l-\|\hat{\gamma}\|\right)\;\|c'\,V\|^{-1}\qquad(11)$$

then control $f\,(\gamma'\,x(t))$, where nonlinear function f is given by (5), stabilizes motions of the system (2) for given δ and ε.

Proof: Let μ is choosen as on (10), let us define vector γ. Let

$$\tilde{u}=\kappa\,\gamma'\,x=\hat{u}-\mu\,\kappa\,c'\,V\,x.$$

The system (2) can be rewritten as

$$\dot{x}=Ax+c\;(\varphi(t)+\tilde{u})+c\;(u-\tilde{u})\,,$$

where $u=f\,(\gamma'\,x)$ has form (3) or (4). Then for $v(x)=x'\,V\,x$ one can obtain

$$\begin{gathered}\dot{v}\big|_{(2)}=-\|x\|^2+2\;(\varphi-\mu\,\kappa\,x'\,Vc)\;c'\,V\,x+\\ +2\;(u-\hat{u})\;c'\,V\,x=-x'\,(E+2\mu\kappa\,Vcc'\,V)\,x+\\ +2\varphi c'\,V\,x+2\;(u-\hat{u})\;c'\,V\,x\le\\ \le-(1+2\mu\kappa\nu)\;\|x\|^2+2\,(\hat{\varphi}+|u-\tilde{u}|)\;\|c'\,V\|\;\|x\|\le\\ \le-(1+2\mu\kappa\nu)\;\|x\|^2+2\,(\hat{\varphi}+\kappa l)\;\|c'\,V\|\;\|x\|<0,\end{gathered}$$

if

$$\|x\|\;>(1+2\mu\kappa\nu)^{-1}\;2\,(\hat{\varphi}+\kappa l)\;\|c'\,V\|\,.$$

So, S_ε is the sought-for ε-neighbourhood of the origin if

$$\varepsilon>\sqrt{\lambda_2/\lambda_1}\;(1+2\mu\kappa\nu)^{-1}\;2\,(\hat{\varphi}+\kappa l)\;\|c'\,V\|\,.$$

The lattest inequality holds true due to (10).

If the nonlinearity under consideration has form (5) then the estimation above holds true if $|\sigma|<\beta l$. Condition (11) is sufficient for that. ■

Similar result can be obtained in case when the right part of the system contain several nonlinearities:

$$\dot{x} = Ax + Cu = Ax + \sum_{i=1}^{r} c_i u_i, \tag{12}$$

here C is $n \times r$-matrix, $r \leq n$; c_i is i-th column of C; $u' = (u_1, \ldots, u_r)$, $u_i = f_i(\sigma_i)$, $\sigma_i = \gamma_i' x$, $\|\gamma_i\| \neq 0$.

Theorem 3: Let there exists a set of linear stabilizing controls for the system (12) (i.e. matrix $\hat{A} = A + \sum_{i=1}^{r} \kappa_i c_i \hat{\gamma}_i'$ is Hurwitz), and V is positively definite matrix satisfying Lyapunov's matrix equation

$$\hat{A}'V + V\hat{A} = -E.$$

Let $\lambda_{1,2}$ are minimum and maximum eigenvalues of the matrix V. Then, if for given positive constants ε and δ

$$\max_{i=\overline{1,r}} l_i < \varepsilon \sqrt{\lambda_1/\lambda_2} \left(2 \sum_{i=1}^{r} \kappa_i \|c_i' V\| \right)^{-1}, \tag{13}$$

$$\max_{i=\overline{1,r}} (\beta_i l_i) > \delta \sqrt{\lambda_2/\lambda_1} \max_{i=\overline{1,r}} \|\hat{\gamma}_i\|, \tag{14}$$

then controls $f_i(\hat{\gamma}_i' x)$ of sort (3) or (4) are stabilizing for the system (12) and given ε and δ.

Proof: Let us rewrite system (12) in the following way:

$$\dot{x} = \hat{A}x + \sum_{i=1}^{r} c_i (u_i - \hat{u}_i).$$

Let $\gamma_i = \hat{\gamma}_i$, $i = \overline{1, r}$, then for

$$|\sigma_i| < \beta_i l_i, \quad 1 \leq i \leq r \tag{15}$$

one have:

$$|u_i - \hat{u}_i| \leq \kappa_i l_i \leq \kappa_i \max_{i=\overline{1,r}} l_i,$$

and

$$\dot{v}\big|_{(12)} \leq -\|x\|^2 + 2\|x\| \sum_{i=1}^{r} \kappa_i l_i \|c_i' V\| \leq$$

$$-\|x\|^2 + 2 \max_{i=\overline{1,r}} l_i \|x\| \sum_{i=1}^{r} \kappa_i \|c_i' V\| < 0,$$

if

$$\|x\| > 2 \max_{i=\overline{1,r}} l_i \sum_{i=1}^{r} \kappa_i \|c_i' V\| < 0.$$

So the estimation for ε is:

$$\varepsilon > 2\sqrt{\lambda_2/\lambda_1} \max_{i=\overline{1,r}} l_i \sum_{i=1}^{r} \kappa_i \|c_i' V\| < 0.$$

implying condition (13).

Inequality (15) holds true if $\|\sigma_i\| < \min_{i=\overline{1,r}} (\beta_i l_i)$, and the latter inequality is true if

$$\|x\| < \left(\max_{i=\overline{1,r}} \|\hat{\gamma}_i\| \right)^{-1} \min_{i=\overline{1,r}} (\beta_i l_i).$$

So the estimate for β is

$$\delta < \sqrt{\lambda_1/\lambda_2} \left(\max_{i=\overline{1,r}} \|\hat{\gamma}_i\| \right)^{-1} \min_{i=\overline{1,r}} (\beta_i l_i),$$

implying inequality (14). ■

IV. STABILITY OF PERIODIC MODES

Note that the system (1), where control statement u is given by the formula (3) or (5), has a trivial solution $x = 0$. Of course, this solution may be stable or unstable depending of Hurwitz property of the matrix A. Moreover, the system under consideration may have stable point $x = 0$ and stable periodic modes simultaneously.

Let us mention an obvious fact concerning presence of fixed points of the system.

Denote

$$\hat{A} = A + \kappa c \gamma'.$$

Lemma 1: If matrices A and $\hat{A}$ are Hurwitz, and

$$\kappa\, \gamma'\, \hat{A}^{-1} c < 1, \tag{16}$$

then system (1), (3) has the unique fixed point $x = 0$ which is asymptotically stable with domain of attraction coincident with $\mathbb{E}^n$.

If the previous conditions hold true and

$$\kappa \gamma'\, A^{-1} c < -\beta/(\beta - \alpha).$$

then the system (1), (5) (where $\hat{f}$ has form (3)) has an unique fixed point $x = 0$ which is asymptotically stable with domain of attraction coincident with $\mathbb{E}^n$.

Proof: If $u = \kappa(\sigma \mp \sigma_0)$, then

$$\dot{x} = A\, x + c\, \kappa\, (\sigma \mp \sigma_0) = \hat{A}\, x \mp \kappa\, c\, \sigma_0.$$

Denote p fixed point of the last system, i.e.

$$0 = \hat{A}\, p \mp \kappa\, c\, \sigma_0,$$

or (matrix $\hat{A}$ is nonsingular),

$$p = \pm\, \kappa\, \hat{A}^{-1}\, c\, \sigma_0.$$

In case if $\gamma'\, p < \pm\sigma_0$, all solutions of the system (1), (3) tend in time into domain

$$-\sigma_0 < \gamma'\, x < \sigma_0.$$

This fact, taking into account Hurwitz property of A, completes the proof.

In the case of the system (1), (5) further deduction can be implemented in the similar manner. ■

Let us colligate one of results of [7]. Consider the system (1) and suppose that nonlinearity f has for example form (4). Denote $x(t - t_0, x_0, u)$ solution of the system (1) for unchanging control law u and initial conditions (t_0, x_0).

Theorem 4: Let the system (1), (4) has periodic solution with two control switching points $s_{1,2}$ such as $\gamma' s_1 = l_0$, $\gamma' s_2 = l$, i.e. exists a pair of positive constants $T_{1,2}$ such as

$$s_1 = x(T_2, s_2, u)\big|_{u=\kappa(\sigma-\sigma_0)}, \quad s_2 = x(T_1, s_1, 0).$$

Let

$$\gamma' A s_2 \neq 0, \quad \gamma' \left(\hat{A} s_1 - \kappa \sigma_0 c \right) \neq 0; \tag{17}$$

and

$$\|M_1 M_2\| < 1, \tag{18}$$

where

$$M_1 = \left(E - \frac{As_2\gamma'}{\gamma' As_2}\right) e^{A\tau_1},$$

$$M_2 = \left(E - \frac{\left(\hat{A}s_1 - \kappa\sigma_0 c\right)\gamma'}{\gamma'\left(\hat{A}s_1 - \kappa\sigma_0 c\right)}\right) e^{\hat{A}\tau_2},$$

then the periodic solution under consideration is asymptotically orbitally stable.

Proof: As $s_2 = e^{A\tau_1}s_1$, then

$$\frac{\partial s_2}{\partial \tau_1} = Ae^{A\tau_1}s_1 = As_2, \quad \frac{\partial s_2}{\partial s_1} = e^{A\tau_1}.$$

Similarly,

$$s_1 = e^{\hat{A}\tau_2}s_2 - \kappa\sigma_0 \int_0^{\tau_2} e^{\hat{A}(\tau_2-\tau)}ds\, c,$$

and

$$\frac{\partial s_1}{\partial \tau_2} = \hat{A}e^{\hat{A}\tau_2}s_2 - \kappa\sigma_0\left(\hat{A}\int_0^{\tau_2} e^{\hat{A}(\tau_2-s)}ds\, c + c\right) =$$

$$= \hat{A}s_1 - \kappa\sigma_0 c, \quad \frac{\partial s_1}{\partial s_2} = e^{\hat{A}\tau_2}.$$

So,

$$d\left(\gamma' s_1\right) = 0 = \gamma' As_2 d\tau_1 + \gamma' e^{A\tau_1}ds_1, \;\; d\tau_1 = -\frac{\gamma' e^{A\tau_1}ds_1}{\gamma' s_2},$$

$$ds_2 = e^{A\tau_1}ds_1 - \frac{As_2\gamma' e^{A\tau_1}ds_1}{\gamma' As_2} = M_1 ds_1;$$

$$d\left(\gamma' s_1\right) = 0 = \gamma'\left(\hat{A}s_1 - \kappa\sigma_0 c\right)d\tau_2 + \gamma' e^{\hat{A}\tau_2}ds_2,$$

$$d\tau_2 = -\frac{\gamma' e^{\hat{A}\tau_2}ds_2}{\gamma'\left(\hat{A}s_1 - \kappa\sigma_0 c\right)},$$

$$ds_1 = e^{\hat{A}\tau_2}ds_2 - \frac{\left(\hat{A}s_1 - \kappa\sigma_0 c\right)\gamma' e^{\hat{A}\tau_2}ds_2}{\gamma'\left(\hat{A}s_1 - \kappa\sigma_0 c\right)} = M_2 ds_2.$$

$$ds_2^{k+1} = M_1 M_2 ds_2^k, \quad k = 0, 1, 2, ...,$$

here ds_2^k are successive deviations of control switching points of diverged solution x from s_2. So the system under consideration causes continuous contracting mapping of some neighbourhood of s_2 lying on hyperplane $\gamma' x = l$ to itself (see (17), (18)).

Use of fixed point principle completes the proof. ■

Example 1: Let

$$A = \begin{pmatrix} -0.1 & 0 \\ 0 & 0.1 \end{pmatrix}, \quad c = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad \gamma = \begin{pmatrix} 0.2 \\ -1 \end{pmatrix},$$

$$\kappa = 0.5, \quad \sigma_0 = 0, \quad l_0 = 0.5, \quad l = 1.$$

In that case system (1), (4) has periodic solution with two points of control switching $s_{1,2}$;

$$s_1 = \begin{pmatrix} 0.9166... \\ -0.3166... \end{pmatrix}, \quad s_2 = \begin{pmatrix} 0.3092... \\ -0.9382... \end{pmatrix},$$

$$\tau_1 = 2.040..., \quad \tau_2 = 10.867...,$$

$$M_1 \approx \begin{pmatrix} 0.873 & -0.433 \\ 0.175 & -0.087 \end{pmatrix}, \quad M_2 \approx \begin{pmatrix} 0.212 & -0.463 \\ 0.042 & -0.093 \end{pmatrix}.$$

Here the condition (18) is satisfied, so the periodic solution is asymptotically orbitally stable.

The system considered has also an asymptotically orbitally stable periodic solution with control switching points $-s_{1,2}$.

Note that statement of the theorem may be extrapolated onto the case of nonlinearity (3) without any changes (with the exception of condition $\gamma' s_1 = \sigma_0$).

Example 2: Consider the system (1), (3), let

$$A = \begin{pmatrix} -0.1 & -1 \\ 0 & 0.1 \end{pmatrix}, \quad c = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad \gamma = \begin{pmatrix} 0.3 \\ -1 \end{pmatrix},$$

$$\kappa = 0.5, \quad \sigma_0 = 0.75, \quad l = 1;$$

it has a pair of periodic solutions with control switching points $s_{1,2}$ and $-s_{1,2}$, where

$$s_1 = \begin{pmatrix} 2.1729... \\ -0.0981... \end{pmatrix}, \quad s_2 = \begin{pmatrix} 2.1930... \\ -0.3421... \end{pmatrix},$$

$$\tau_1 = 5.370..., \quad \tau_2 = 12.488....$$

In such a case

$$M_1 \approx \begin{pmatrix} 0.2814 & 0.2452 \\ 0.0844 & 0.0736 \end{pmatrix}, \quad M_2 \approx \begin{pmatrix} -0.1630 & 0.3588 \\ -0.0489 & 0.1076 \end{pmatrix},$$

condition (18) is satisfied, so the periodic solutions considered are asymptotically orbitally stable.

Note that finding of the vectors $s_{1,2}$ and values $\tau_{1,2}$ is a laborious computing procedure. Let us cite a particular case in which exact expressions for s_i and τ_i can be obtained.

Lemma 2: Let us consider the system (1) where control statement is given by (4). Let $A'\gamma = \lambda\gamma$, for some real $\lambda > 0$ (i.e. γ is an eigenvector of the matrix A'). If

$$\gamma' c < -\frac{\lambda l_0}{\kappa\left(l_0 - \sigma_0\right)} \tag{19}$$

then the system under consideration has two periodic modes with period

$$\tau = \tau_1 + \tau_2, \quad \tau_1 = \lambda^{-1}\ln\frac{l}{l_0}, \quad \tau_2 = \hat{\lambda}^{-1}\ln\frac{\hat{\lambda}\, l_0 + u_0}{\hat{\lambda}\, l + u_0},$$

where

$$\hat{\lambda} = \lambda + \kappa\,\gamma' c, \quad u_0 = -\kappa\,\sigma_0\,\gamma' c.$$

Proof: If $u = 0$, then

$$\dot{\sigma} = \lambda\,\sigma, \quad \sigma(0) = l_0, \quad l = \sigma\left(\tau_1\right) = l_0\, e^{\lambda\tau_1},$$

$$\tau_1 = \lambda^{-1}\ln\left(l/l_0\right).$$

If $u = \kappa\left(\sigma - \sigma_0\right)$, then

$$\dot{x} = \hat{A}\, x - \alpha\,\kappa\, c\, l, \quad \dot{\sigma} = \hat{\lambda}\sigma + u_0.$$

Due to (19) $\hat{\lambda} < 0$, and $\dot{\sigma} < \hat{\lambda}l_0 + u_0$, when $\sigma \in [l_0;\, l]$. So $\dot{\sigma} < 0$ if $\hat{\lambda}l_0 + u_0 < 0$, or

$$\lambda l_0 + \kappa\left(l_0 - \sigma_0\right)\gamma' c < 0.$$

The latter inequality holds true if condition (19) is satisfied.

Then,

$$l_0 = \sigma(\tau) = e^{\hat{\lambda}\tau_2}\sigma(\tau_1) + \hat{\lambda}^{-1}\left(e^{\hat{\lambda}\tau_2} - 1\right)u_0 =$$
$$= e^{\hat{\lambda}\tau_2}\left(l + \hat{\lambda}^{-1}u_0\right) - \hat{\lambda}^{-1}u_0,$$
$$l_0 + \hat{\lambda}^{-1}u_0 = e^{\hat{\lambda}\tau_2}\left(l + \hat{\lambda}^{-1}u_0\right),$$

the last equation yields expression for τ_2.

Denote control switching points $s_{1,2}$,

$$\gamma' s_1 = l_0, \quad \gamma' s_2 = l.$$

Let us derive explicit expressions for $s_{1,2}$. Suppose for simplicity that $\det \hat{A} \neq 0$. In that case

$$s_1 = e^{\hat{A}\tau_2}s_2 - \kappa\sigma_0\hat{A}^{-1}\left(e^{\hat{A}\tau_2} - E\right)c$$

(if matrix $\hat{A}$ is singular then one can easy derive expressions for s_1 using Cauchy's formula). After substitution

$$s_2 = e^{A\tau_1}s_1$$

in the later expression one have:

$$s_1 = e^{\hat{A}\tau}s_1 - \kappa\sigma_0\hat{A}^{-1}\left(e^{\hat{A}\tau_2} - E\right)c,$$

and

$$s_1 = \kappa\sigma_0\left(e^{\hat{A}\tau} - E\right)^{-1}\hat{A}^{-1}\left(e^{\hat{A}\tau_2} - E\right)c,$$

if $\det\left(e^{\hat{A}\tau} - E\right) \neq 0$. ■

The statement of the previous theorem suppose various extensions. For example we can consider periodic solutions of the system with the larger number of control switching points.

Theorem 5: Let the system (1), (4) has periodic mode with four points of control swithching s_i, $i = \overline{1,4}$ such as

$$\sigma_1 = l_0, \quad \sigma_2 = -l, \quad \sigma_3 = -l_0, \quad \sigma_3 = l, \quad \sigma_i = \gamma' s_i,$$

i.e. for some positive T_i, $i = \overline{1,4}$,

$$s_1 = x(T_4, s_4, u)\big|_{u=\kappa(\sigma-\sigma_0)}, \quad s_2 = x(T_1, s_1, 0),$$
$$s_3 = x(T_2, s_2, u)\big|_{u=\kappa(\sigma+\sigma_0)}, \quad s_4 = x(T_3, s_3, 0).$$

Denote

$$A_1 = A_3 = A, \quad A_2 = A_4 = \hat{A},$$
$$c_1 = c_3 = 0, \quad c_2 = \sigma_0\,\kappa\,c, \quad c_4 = -c_2.$$

Let

$$\gamma'\left(A_i s_{i+1} + c_i\right) \neq 0$$

(hereafter suppose $i + 1 = 1$ if $i = 4$), and

$$\|M\| = \|M_4\,M_3\,M_2\,M_1\| < 1, \tag{20}$$

where

$$M_i = \left(E - \frac{\left(A_i s_{i+1} + c_i\right)\gamma'}{\gamma'\left(A_i s_{i+1} + c_i\right)}\right)e^{A_i\tau_i}.$$

In such a case the periodic solution under consideration is asymptotically orbitally stable.

Proof: Using the above definitions, one can obtain:

$$\dot{x} = A_i x + c_i, \quad x(0) = s_i, \quad s(\tau_i) = s_{i+1},$$

so

$$s_{i+1} = e^{A_i\tau_i}s_i + \int_0^{\tau_i} e^{A_i(\tau_i - s)}c_i\,ds,$$

and

$$\frac{\partial s_{i+1}}{\partial s_i} = e^{A_i\,\tau_i}, \quad \frac{\partial s_{i+1}}{\partial \tau_i} = A_i s_{i+1} + c_i.$$

After differentiation:

$$\gamma' ds_{i+1} = 0 = \gamma' e^{A_i\,\tau_i}ds_i + \gamma'\left(A_i s_{i+1} + c_i\right)d\tau_i.$$

So the expression for $d\tau_i$ may be written as

$$d\tau_i = -\frac{\gamma' e^{A_i\,\tau_i}ds_i}{\gamma'\left(A_i s_{i+1} + c_i\right)},$$

and

$$ds_{i+1} = \left(E - \frac{\left(A_i s_{i+1} + c_i\right)\gamma'}{\gamma'\left(A_i s_{i+1} + c_i\right)}\right)e^{A_i\tau_i}\,ds_i = M_i\,ds_i.$$

Finally,

$$ds_1^{k+1} = M ds_1^k,$$

where ds_1^k are successive deviations from the control switching point s_1 lying on control switching hyperplane $\sigma = l_0$. Use of fixed point principle completes the proof. ■

Example 3: Let

$$A = \begin{pmatrix} -0.1 & -0.5 \\ 0 & 0.1 \end{pmatrix}, \quad c = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad \gamma = \begin{pmatrix} 0.3 \\ -1 \end{pmatrix},$$
$$\kappa = 0.5, \quad \sigma_0 = 0, \quad l_0 = 0.5, \quad l = 1.$$

Then the system (1), (4) has periodic solution with four points of control switching s_i, $i = \overline{1,4}$,

$$s_1 = \begin{pmatrix} 1.9195... \\ 0.0759... \end{pmatrix}, \quad s_2 = \begin{pmatrix} -1.2800... \\ 0.6160... \end{pmatrix},$$
$$s_3 = -s_1, \quad s_4 = -s_2,$$
$$\tau_1 = \tau_3 = 1.9635..., \quad \tau_2 = \tau_4 = 20.9341...,$$
$$M = M_4\,M_3\,M_2\,M_1 \approx \begin{pmatrix} 0.00018 & 0.00056 \\ 0.00006 & 0.00017 \end{pmatrix},$$

condition (20) is satisfied. So the solution under consideration is asymptotically orbitally stable.

The above statement may be extrapolated onto the case of nonlinearity (3) without any changes (with the exception of conditions $\sigma_1 = \sigma_0$, $\sigma_3 = -\sigma_0$). Consider an example:

Example 4: Let A, c, γ, κ and l are the same as in the previous example, $\sigma_0 = 0.2$. In that case system (1), (3) has periodic solution with four control switching points;

$$s_1 = \begin{pmatrix} 1.4165... \\ 0.2250... \end{pmatrix}, \quad s_2 = \begin{pmatrix} -1.0562... \\ 0.6832... \end{pmatrix},$$
$$s_3 = -s_1, \quad s_4 = -s_2,$$
$$\tau_1 = \tau_3 = 0.5088..., \quad \tau_2 = \tau_4 = 11.109...;$$
$$M = M_4\,M_3\,M_2\,M_1 \approx \begin{pmatrix} 0.0123 & 0.0437 \\ 0.0037 & 0.0131 \end{pmatrix},$$

condition (20) is satisfied, i.e. the solution under consideration is asymptotically orbitally stable.

Note that all variety of movements of the system (1) of course is not limited with fixed points or periodic modes. For example we can observe an ω-limit set of the system coinciding with hyperplane $\sigma = 0$ if nonlinearity f has form (3), $\sigma_0 = 0$, $A'\gamma = \lambda\gamma$ and

$$\lambda > 0, \quad \hat{\lambda} = \lambda + \kappa\gamma' c < 0$$

(let us omit the proof).

V. CONCLUSION

The above results suppose further development. Investigation of stable modes of the forced system (2) is an individual complex and interesting task (see [4]). Results similar to obtained in the previous part can be outlined for periodic solutions of the system (1) with more complicated periodic modes having a large amount of control switching point, or for the system (12). As to the first part, more complicated stabilization problems can be considered, e.g. that of discrete stabilization, stabilization with incomplete information etc. (see [8]).

REFERENCES

[1] V.N. Shamberov. Effect of non-Coulomb dry friction on the stability of automatic systems. *Doklady Physics*, Vol.50, No.3, 2005. PP. 151-153.

[2] C. Canudas de Wit, H. Olsson, K. Åström and P. Lischinsky. A new model for control of systems with friction. In *IEEE Trans. on Automatic Control*, Vol. 40, No. 3, pp. 419-425, 1995.

[3] H. Ollson, K.J. Åström. Friction Generated Limit Cycles. In *IEEE Trans. on Control Systems Technology*, Vol.9, Issue 4, pp. 629-636, 2001.

[4] A.M. Kamachkin, V.I. Shamberov. Automatic systems with essentially nonlinear characteristics. *St.-Petersburg state marine technical univ. press*, 1995. In Russian.

[5] V.V. Petrov, A.A. Gordeev. Nonlinear servomechanisms. *Moscow, Mashinostroenie Publishers*, 1979. In Russian.

[6] V.I. Zubov. Theory of oscillations. *Singapore etc., World Scientific*, 1999.

[7] S.V. Zubov, N.V. Zubov. Mathematical methods for stabilization of dynamic systems. *St.-Petersburg univ. press*, 1996. In Russian.

[8] E.Ya. Smirnov. Stabilization of program motions. *St.-Petersburg univ. press*, 1997. In Russian.

Design and Layout Implementation of Microstrip Balanced Diode Filter

Salman M. Khan, Jamshed Iqbal and Faisal Masood

***Abstract*—The paper describes the design and simulation of a balanced mixer using a 180º hybrid ring to combine Radio Frequency (RF) input and Local Oscillator (LO) and a pair of non-linear elements. The centre RF and Intermediate Frequency (IF) have been set at 10 GHz and 400 MHz respectively and there is no band inversion. The 180º hybrid is implemented as a 90º hybrid with an extra quarter wavelength transmission line. The Schottky diodes act as non-linear elements. The design has all elements needed to have the diodes correctly biased in DC. Computer Aided Design (CAD) based softwares have been used to design and simulate the mixer. Prior to design, the performance has been evaluated. The conversion loss and the optimum LO power to minimize the loss has been determined. The bandwidth of the mixer at -3dB has been computed and simulation results provide plots of conversion losses as a function of frequencies. Extension of this plot shows the image frequency rejection. Computation of LO-RF isolation as well as the LO leakage to the IF port have also been carried out. Other necessary performance parameters e.g. noise figure, noise contribution by each element and 3rd order intercept point etc have been determined. Following the performance evaluation of the proposed mixer, finally a Printed Circuit Board (PCB) layout has been designed. Thus the present work completely encompasses related issues of microstrip balanced diode filter design.**

I. Introduction

Mixers are frequency translation devices. They allow the conversion of signals between a high frequency (the RF frequency) and a lower Intermediate Frequency (IF) or baseband. In communications systems the RF is the transmission frequency, which is converted to an IF to allow improved selectivity (filtering) and an easier implementation of low noise and high gain amplification. This paper details the design of mixer circuits, concentrating on low cost Printed Circuit Board (PCB) based designs using discrete Surface Mount Technology (SMT) components. Mixers are used to shift the frequency band of a given signal to a different center frequency. It is accomplished by multiplying the input signal with a tone.

$$F[v(t)\cos 2\pi f_{LO}t] = \frac{1}{2}[V(f-f_{LO}) + V(f+f_{LO})] \qquad V(f) = F[v(t)] \quad (1)$$

Manuscript received August 29, 2010.

S. M. Khan is with the School of Information Technology & Engineering (SITE), Ottawa Carleton Institute for Electrical and Computer Engineering, Canada (email: salman9201@yahoo.com)

J. Iqbal is with the ADVanced Robotics (ADVR) Department, Italian Institute of Technology (IIT) and University of Genova, Italy (phone: +39-010-71781481; fax:+39-010-720321; e-mail: jamshed.iqbal@iit.it).

F. Masood is a Senior Design Engineer in Powersoft19 Inc. (e-mail: fmasood@powersoft19.com).

Two bands are theoretically obtained: sum and difference. For down converters, the wanted signal is the difference. Intermediate frequency (IF) is the difference between signal frequency f_{RF} and LO frequency $f_{LO.}$ Image frequency produces the same IF as the input signal. A double-balanced diode mixer normally make use of four diodes in a ring or star configuration with both the LO and RF being balanced. All ports of the mixer are inherently isolated from each other. The advantages of a double-balanced design over a single balanced design are increased linearity, improved suppression of spurious products (all even order products of the LO and/or the RF are suppressed) ant the inherent isolation between all ports. The disadvantage is that they require a higher level LO drive.

A mixer uses the non-lineal characteristics of the diode to generate an output consisting on the sum and the rest between two frequencies (RF and OL). After the diodes, there is a low pass filter so as to discard RF + OL frequency. The general scheme of the design is shown in Figure 1.

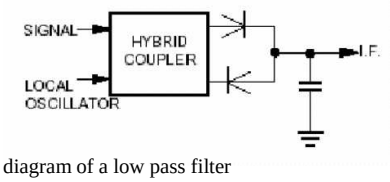


Fig. 1. Block diagram of a low pass filter

The hybrid coupler is usually implement with a 180-degree hybrid but in this case is used a 90-degree hybrid because in the layout the output ports are nearer than in the 180-degree one. The used diodes will be a pair that offers the ADS library. They are two identical diodes, which are polarized through a network of transmission lines. In this case, the mixer is analyzed with a harmonic balance simulation that will allow us calculate the inter-modulation products. This simulation needs the HB controller (Simulation-HB library). The basic needed values are the frequency of the inputs, the harmonics of each frequency and the maximum order of the inter-modulation products.

Table I shows the mixer specifications.

TABLE I
MIXER SPECIFICATIONS

Parameter	Value
Input Frequency	10 GHz
Output Frequency	400 MHz
Hybrid	Microstrip 90º hybrid plus 90º TL
Diodes	Schottky
Band inversion	No

With these specifications, the first step in the design phase was to select the fundamental components in a mixer. Block diagram of the mixer is shown in Figure 1. The approach for a diode passive mixer employs a 180º hybrid mixer to combine the LO with RF to feed the two signals using the non linear element which is a pair of Schottky diodes HSMS8202 suitable for frequency range of 10GHz. Using a 180º hybrid (based on 90º hybrid plus a 90º phase shift TL), we kept the LO with 180º phase shift in the two outputs of the hybrid and fed together into the diodes added in counter phase. This technique will automatically isolate the LO and the IF output.

Furthermore, a coupled line RF input Band Pass Filter (BPF) for image rejection purposes and a step impedance Low Pass Filter (LPF) for the IF output has been incorporated into the mixer chain. This approach keeps microstrip Printed Circuit Board (PCB) substrate implementation simple. The transmission lines have been designed taking into consideration the implementations limitations in standard microstrip boards.

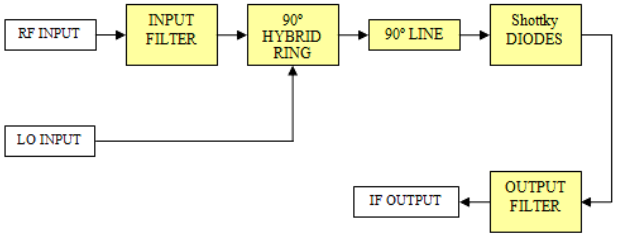


Fig. 2. Block diagram of the mixer circuit

The rest of the paper is organized as follows: In Section II, we will introduce the concept from theoretical point of view. Section III contains the design details and implementation results. The performance evaluation and simulation results have been presented in Section IV. Section V presents the layout implementation of the proposed design. Finally, Section VI discusses the conclusion.

II. THEORETICAL ANALYSIS

Mixers are frequency translation devices thereby permitting the conversion of signals between a high frequency (the RF frequency) and an intermediate frequency, usually lower than the RF. Most of the mixer circuits are based on the non linear response of a component, which can be a diode or a transistor. Diode is used as the non-linear element. Schottky diode offers better solution because of having a low forward voltage drop and a very fast switching action. The Schottky diode can be modeled using (1) while its IV characteristics have been shown in Figure 2.

$$I = a_1 \cdot V + a_2 \cdot V^2 + a_3 \cdot V^3 + \ldots (2)$$

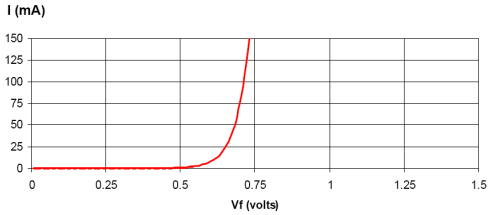


Fig. 3. I-V characteristics of Shottky diode

If the diode is excited with a sum of two signals,

$$I = a_1 \cdot (\cos(w_1 \cdot t) + \cos(w_2 \cdot t)) + a_2 \cdot (\cos(w_1 \cdot t) + \cos(w_2 \cdot t))^2 + \ldots (3)$$

Expanding the formula and applying the trigonometric relation,

$$2 \cdot \cos(w_1 t) \cdot \cos(w_2 t) = \cos((w_1 - w_2)t) + \cos((w_1 + w_2)t) \ldots\ldots (4)$$

The interesting product of a mixer is the first order product; which gives addition and subtraction of inputs. The higher frequencies components at $2 \cdot RF$ and $2 \cdot LO$ have been obtained and needed to be filtered out. The mixer presented works with the subtraction of LO and RF. Because of the requirement of having no band inversion, the LO has to be at frequency of 9.6 GHz to give a 400 MHz output.

$$IF = RF - L.O$$
$$IF = 10Ghz - 9.6Ghz = 400Mhz$$
$$RF\_\mathrm{Im} = RF - 2 \cdot IF = 10Ghz - 800Mhz = 9.2Ghz \ldots\ldots (5)$$

Where RF_im is the image frequency at the input that will be converted to the same IF frequency at the output. RF_Im must be filtered. The diode mixers have usual loses of 5 to 8 dB. The basic schematic of a single diode mixer is shown in Figure 4.

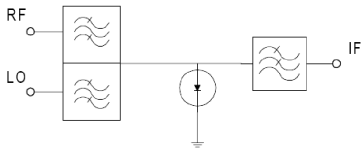


Fig. 4. Schematic diagram of a diode mixer

The LO in a diode mixer has to reach signals of about 8 dBm of power, so it is a good idea to cancel the output from the IF port. A technique to do this, used in proposed microstrip as well is to design a phase shift network (Fig. 5). The 180º hybrid ring makes a 180º phase shift only in the LO section, thus, when added together in the diode, the two waveforms will be cancelled with each other. An attenuation of 40 dB has been achieved in our simulation using this technique.

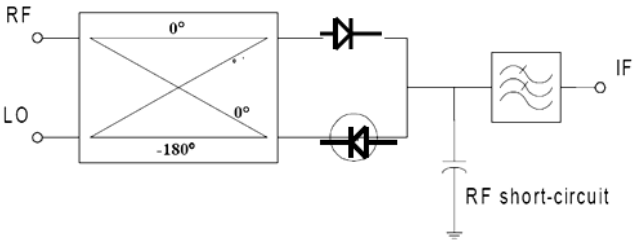


Fig. 5. Block diagram for Phase shift network

III. Circuit Design

The circuit implementation is done in the HP Advanced Design System Computer Aided Design (CAD) program, using the Microstrip Tl library and the component library for the diode model. Substrate parameters are based on a commercial microwave substrate, obtained directly from the datasheet and placed in the MSUB ADS section. Following the block diagram presented in Section I, the complete mixer implementation has been divided in several subsystems.

A. *Coupled lines Input Filter*

Implementation of the input filter is based on a microstrip coupled lines sub circuit and theoretical calculations from the filter theory. The filter order is N=3 and the input and output impedances are 50 ohms. The schematics of coupled lines input filter is shown in Figure 6 while Figure 7 illustrates the corresponding simulated response.

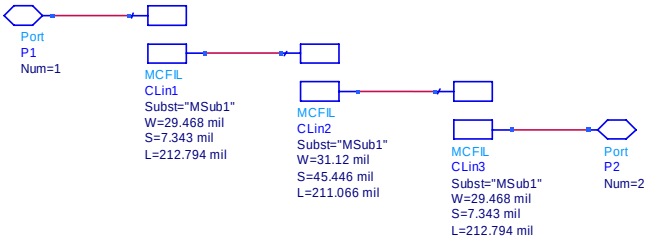


Fig. 6. Schematic diagram of coupled lines input filter

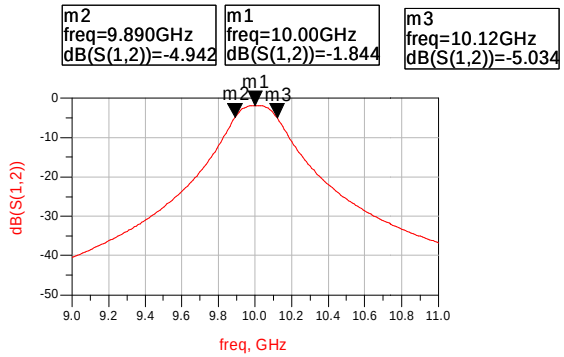


Fig. 7. Coupled lines input filter simulated response

The pass band attenuation is -1.844 dB and the bandwidth for 3 dB attenuation with reference to band pass attenuation is 230 MHz.

B. *180º Hybrid Design*

The next subsystem is the hybrid ring that permits feeding the sum of filtered RF signal and LO signals to the diode pair thereby keeping the isolation and port match at correct levels. The ring has been implemented and simulated in a separate circuit. Figures 9 to 12 show the corresponding results.

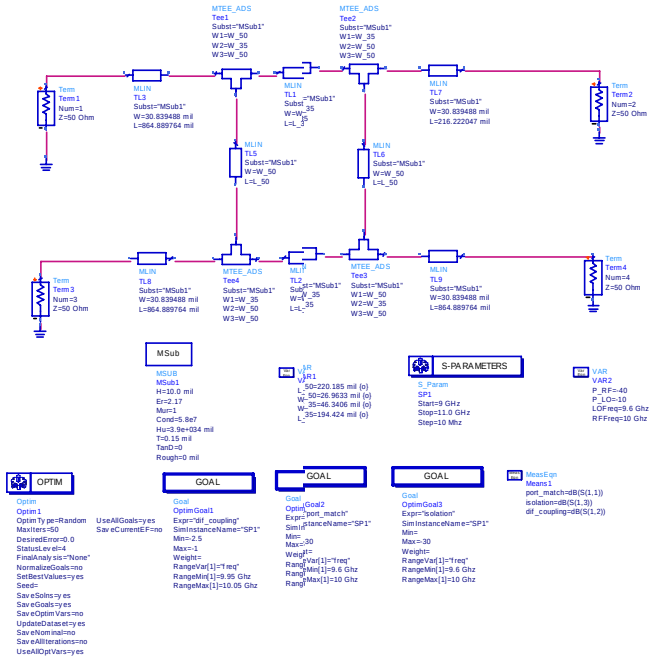


Fig. 8. Schematic diagram of the hybrid ring design implemented in order to achieve port match at correct levels

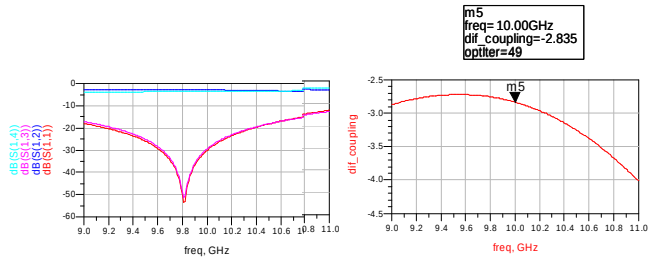


Fig. 9. Hybrid ring simulated response to keep isolation and match the ports at correct levels

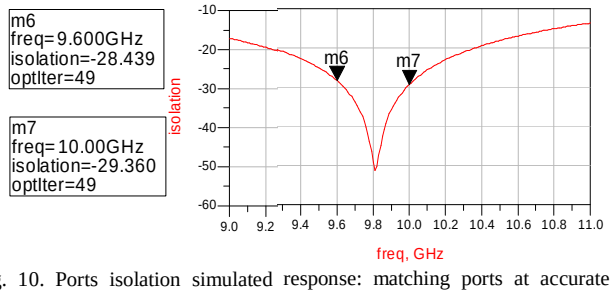


Fig. 10. Ports isolation simulated response: matching ports at accurate levels

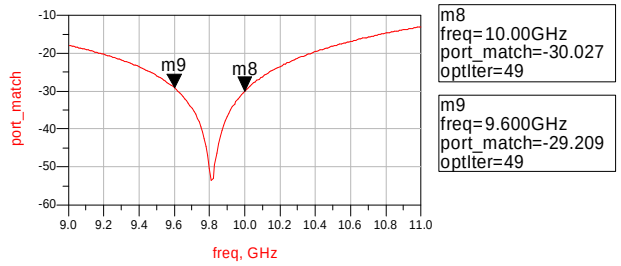


Fig. 11. Port matching simulated response: matching ports at accurate levels

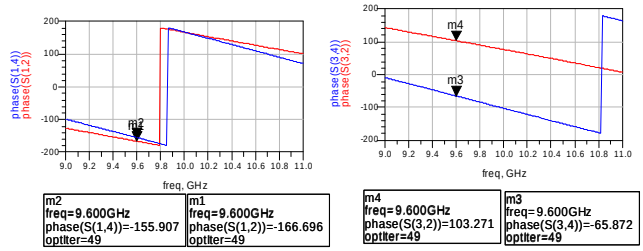


Fig. 12. Phase shift simulated response: matching ports at accurate levels

The final implementation of hybrid ring is shown in Figure 13.

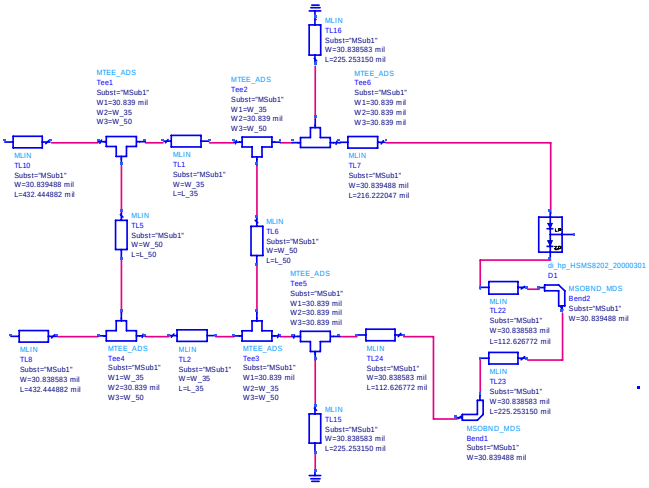


Fig. 13. Schematic diagram of hybrid ring circuit with filtered RF signals and LO signals as inputs to the diode pair

For the hybrid ring, the distance and impedances have been computed using 'linecalc' function. The final implementation has a number of bend lines to keep the layout easy to build. Another important point to mention in this schematic is the use of two λ/4 short circuit terminated stubs at the middle frequency between the LO and the RF. This is to provide a path to ground for the DC signal generated by the diodes in the mixing process.

C. *Step Impedance LPF*

The LPF in a mixer has to avoid the output of the LO and the RF frequencies and also other inter-modulation products from the IF band. For the mixer in our case, the IF was 400 MHz and the bandwidth was about 200 MHz.

Theoretically a LPF at 400 MHz plus half of the bandwidth will attenuate the undesired signals. However in practice, 400 MHz is a very low frequency compared with the 9.6 MHz frequency of LO. At 400 MHz a microstrip filter must have a big size transmission lines, because the wavelength is more less 70 cm. The solution is to design a LPF but with a comfortable cutoff frequency near 9.6 GHz that can be implemented with step impedances. The corresponding captured schematic is shown in Figure 14.

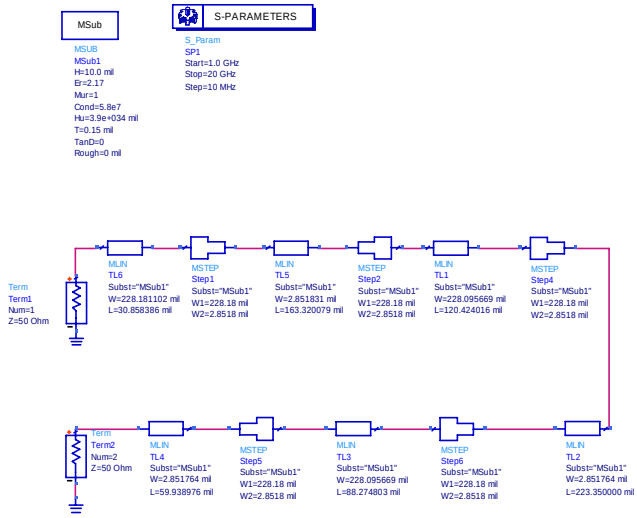


Fig. 14. Schematic diagram of step impedance Low pass filter

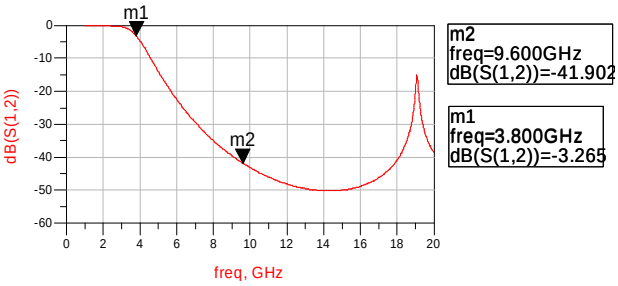


Fig. 15. Step impedance LPF simulated response: avoiding output of LO/RF frequencies and inter-modulation products from the IF band

D. *λ/4 Stub*

Using an output filter will bounce back the LO and RF energy to the ring because at 9.6 GHz and 10 GHz the filter shows high impedance. The solution is to put a stub in the diode middle output with the behavior of a band stop filter, with enough bandwidth to cover the entire LO and RF frequencies. This element is the microstrip radial stub (Figure 16). Using the Tune tool in ADS, the center band reject frequency can be adjusted as depicted in Figure 17.

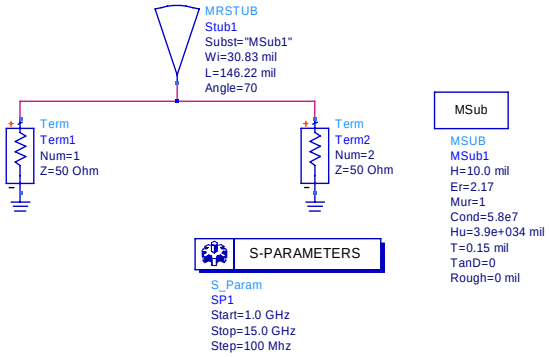


Fig. 16. Schematic diagram of microstrip radial stub

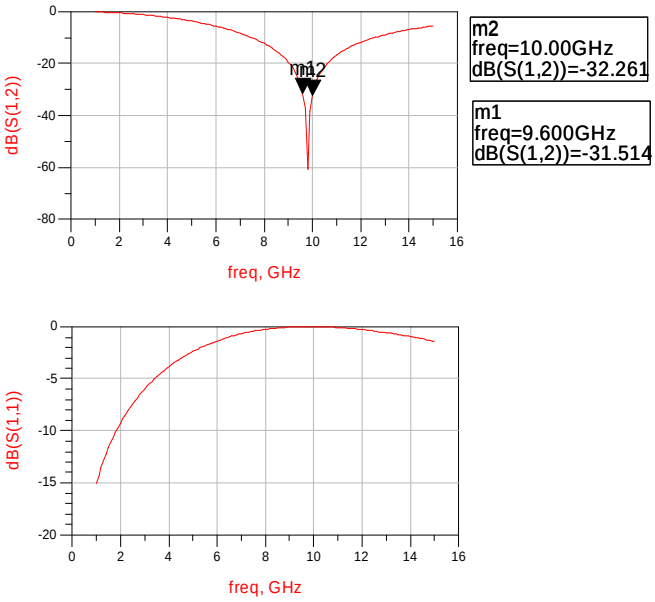


Fig. 17. Center band reject frequency adjustment

IV. Performance Evaluation And Simulation Results

The complete mixer circuit is evolved as a result of combining several subsystems discussed in Section III. The resultant overall circuit having three ports: RF input, LO input and IF output is shown in Figure 16 (at the end of paper). The next step is essentially to investigate the performance of the designed mixer. Various conversion losses have been analyzed and important parameters like noise figure and 3rd order interception point have been computed.

A. Conversion Loss versus LO Power and Isolation

In this simulation scenario carried out by the HP ADS 2008 assistant for mixers, the conversion loss versus LO power and isolation has been evaluated. The mixer sub circuit has to be placed in the test point, and the frequency parameters must be adjusted to the mixer specifications (Figure 18). The simulation results include plot of the conversion loss with LO power with the RF input kept at -40 dBm (Figure 19 to 20).

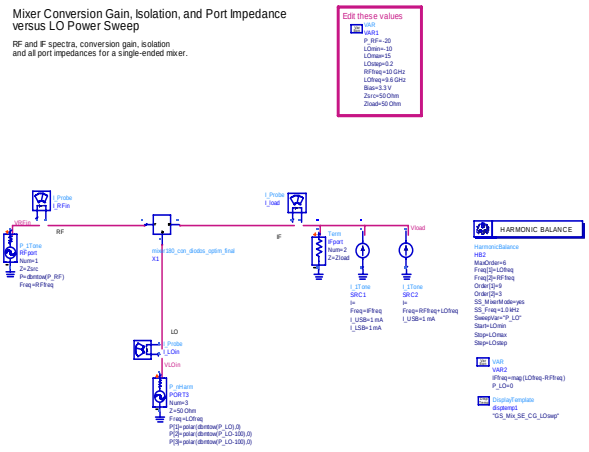


Fig. 18. Schematic for evaluation of conversion loss

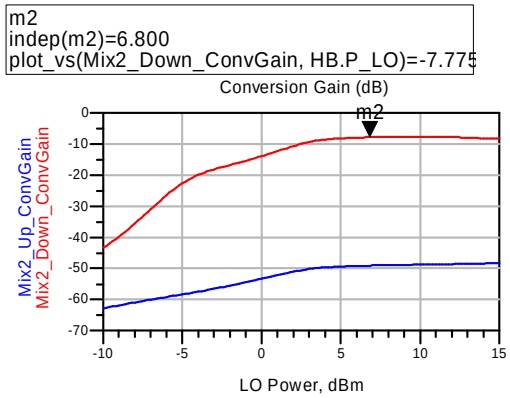


Fig. 19. Conversion loss as a function of LO power

For an optimum point we have chosen LO Power of 6.8 dBm, the conversion loss was 7.77 dB for the whole mixer, including all the filters. The port to port isolation simulation results are shown in Figure 20. The isolation between the LO port and the IF port was -79 dB and between the LO to RF ports, it was -40.7 dB.

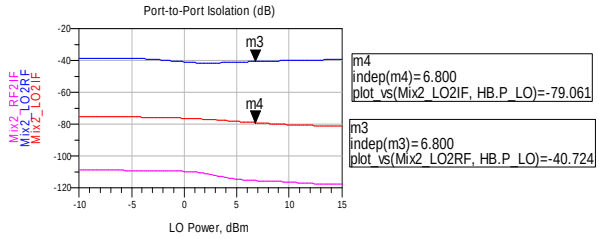


Fig. 20. Port to port isolation simulated response

B. Conversion Loss versus Input Frequency

Using the mixer simulation assistant, the mixer has been simulated with the same template. In this case, the parameter to sweep was input frequency. Figure 20 depicts values of parameters used in this simulation scenario.

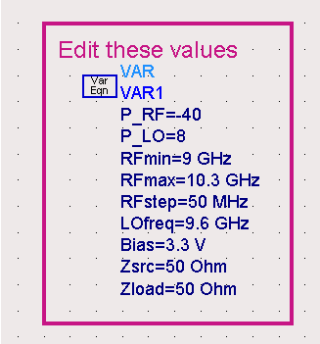


Fig. 21. Parameters value for evaluation of conversion loss as a function of input frequency

To get a frequency response including the image rejection response, the sweep will be from 9 GHz to 10.3 GHz with a step size of 50 MHz step size. The corresponding results are shown in Figure 21. The 3dB bandwidth was 200MHz centered at 10 GHz and the image rejection at 9.2GHz was 42.39dB. In Figure 21, the false gain peak at 9.6 GHz is because the LO output is in the IF.

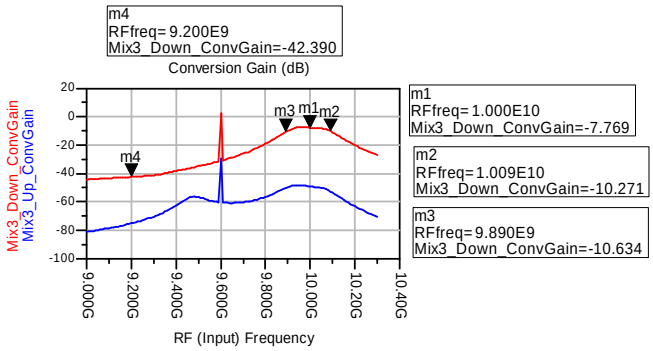


Fig. 22. Conversion loss as a function of input frequency simulated response

C. *Noise Figure*

The Noise Figure (NF) and contribution of each element of the proposed mixer has been analyzed using the ADS 2008 has an assistant. The results have been shown in Figure 23.The simulation results show a NF of 7.281 dB with LO power of 6.8 dBm.

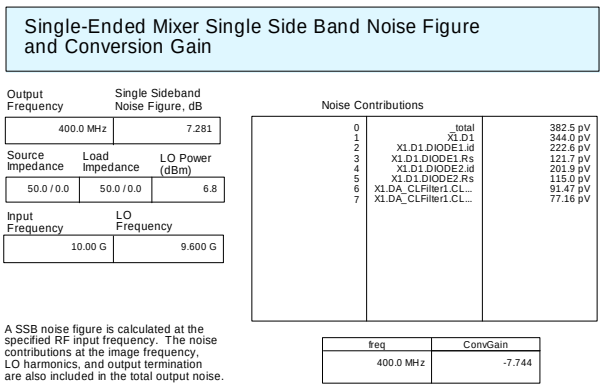


Fig. 23. Results from the Noise Figure analysis

D. *Third Order Interception Point*

The third order interception point has been simulated with two tones and a power sweep to see the input power needed to obtain a third order harmonic power equal to the main conversion signal output. We have fed the mixer with two tones separated by a small frequency difference and have simulated the mixer for a sweep of input power (Figure23). The two input tones were separated by 125 MHz, centered on 10 GHz, so the outputs will be 10.0625 GHz – 9.6 GHz and 9.9475 GHz – 9.6 GHz. The third order inter-modulation was placed at 375 MHz (n*125MHz where n=3). The results are presented in Table II.

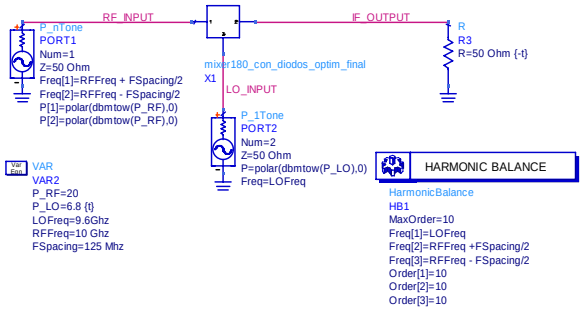


Fig. 24. Schematic of analysis of 3rd order interception point

TABLE II
SIMULATION RESULTS

RF Power (dBm)	IF O/p Power (dBm)	3rd Order O/p Power (dBm)
-40	-48	-225
-10	-18	-91
0	-10	-69
10	-8	-35
20	-13	-30
30	-22	-19

The simulation results are in coincidence with the theoretical concept that explains that 3rd order interception point is about 10 dB or 15 dB above the 1dB compression point.

V. LAYOUT IMPLEMENTATION

After design and evaluation of the mixer, Agilent ADS has been used to implement the microstrip layout of the schematic. Steps for implementing the proposed layout have been mentioned below. The final layout of the Printed Circuit Board (PCB) has been shown in Figure 26.

1. Include extra T.L to be able to solder the connectors (SMA in-circuit female type).
2. Import the components and Microstrip lines to layout.
3. Move them to the right positions to keep the board without extra tracks.
4. Identify the needs of transmission line bends to fit the point.
5. Place the bends in the schematic.
6. Start again the process from point 1 until no more bends required.

VI. Conclusion

The simulated performance of the optimized mixer has been tabulated in Table III.

TABLE III
MIXER PARAMETERS

Parameter	Value
Optimum LO power	6.8 dBm
Conversion loss	7.7 dBm
Bandwidth 3 dB	200 MHz
Image rejection	42.39 dB
Isolation LO vs. IF	-79 dB
Isolation LO vs. RF	-40.7 dB
Noise Figure	NF = 7.2dB.
3rd Order interception point	+ 30 dBm (RF I/p power)

From these parameters, it is clear that the proposed work presents a realistic diode mixer. Further optimization can be done in the hybrid ring, biasing to process to obtain more isolation between the LO and IF or RF. A conversion loss of 7.7 dB is acceptable taking into account that the mixer incorporates the input and the output filters. We have designed the components to be implemented using only microstrip techniques, except the transistor and a low pass coil to eliminate the DC level at the output, because this coil has to be able to present a high impedance at 400 MHz which is too low frequency for a microstrip transmission line. A prototype of PCB has also been designed and has been sent for fabrication.

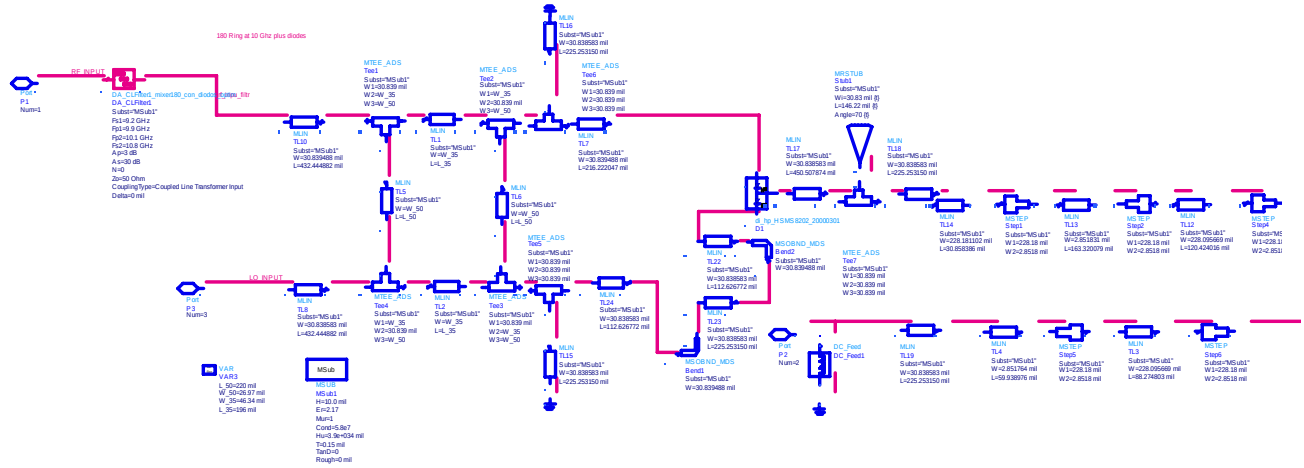


Fig. 25. Schematic of overall mixer circuit

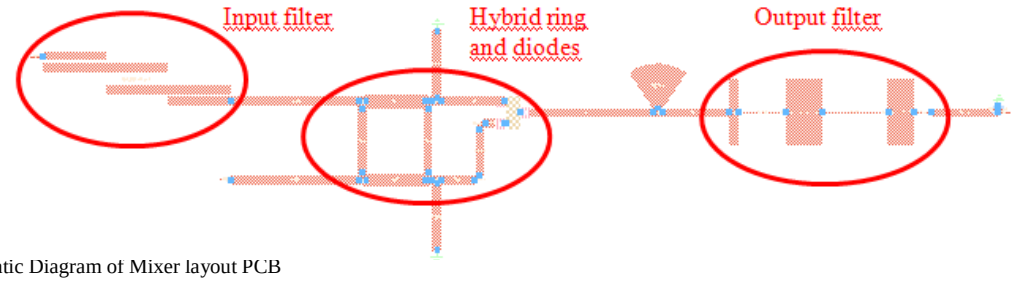


Fig. 26. Schematic Diagram of Mixer layout PCB

References

[1] David M. Pozar, “Microwave Engineering”, 2nd ed.

[2] Sung Tae Choi, Design of Microstrip Balanced Diode Mixer for 38 GHz Band

[3] L. Pradell, *Design and Analysis of RF and MW Systems Course Slides*, UPC, Barcelona, Spain 2008

[4] Max W. Medley, “RF and MW Circuits: Analysis, Synthesis and Design”, Artech House Inc, December 1992

[5] Hunter, M.T.J. “The Basics of System Design”, Proceedings of the IEE Tutorial Colloquium on “How to Design RF Circuits”, Wednesday 5th April 1999, Savoy Place, London

[6] Maas, S.A. “Microwave Mixers”, Artech House, ISBN 0-89006-605-1

[7] Walker, J.L.B. “Filters”, Proceedings of the IEE Tutorial Colloquium on “How to Design RF Circuits”, Wednesday 5th April 1999, Savoy Place, London

[8] Walker, J.L.B. ”Improvements to the Design of the 180° Rat Race Coupler and its Application to the design of Balanced Mixers with High LO to RF Isolation”, IEEE MMT-S Digest, 1997, Vol. II, pp 747-750

Temperature Control Application for a Ventilation System using PIC18F4620

Bogdan Levarda and Cristina Budaciu
Department of Automatic Control and Applied Informatics,
Faculty of Automatic Control and Computer Engineering, Iaşi, Romania
e-mail:lvrd_bogdan@yahoo.com, chalauca@ac.tuiasi.ro

***Abstract* — Applications that require temperature control are often meet in industry. In this paper a low cost application for temperature control in a ventilation system using the PIC18F4620 was designed and developed. Ventilating is the process of changing or replacing air in any space to control temperature or remove moisture, smoke, dust, unpleasant smells or bacteria. Ventilation in a test room refers both to the exchange of air to the outside as well as circulation of air within a room. This study includes real time temperature control using a PID controller implemented on a microcontroller.**

KEYWORDS: temperature control, educational low cost application, PID controller, PIC18F4620

I. INTRODUCTION

Today, the demand for accurate temperature control and air ventilation control has conquered many of industrial domains such as process heat, alimentary industry, automotive, industrial spaces or office buildings where the air is cooled in order to maintain a comfortable environment for its occupants. One of the most important concerns involved in heat area consist in the desired temperature fruition and consumption optimization. To fulfill such a challenge one should promote suitable control strategies. In the last decade extensively research has been made with respect to temperature control for different types of processes. In the paper [1] the authors propose a fuzzy PID thermal control system for a casting process. Real time PID control for water heating system using PIC16F887 microcontroller was designed and implemented as is shown in [2]. The authors propose design architecture for water temperature control. The study implies both acquisition and modeling techniques and control strategies based on PID controller.

A mandatory demand to implement the agreeably solution is the model acquirement which describes the complex behavior of the system. The paper focuses on the model identification and control of temperature in a test room designed for a ventilation system.

The main reason to derive a low cost application is that to be suitable for different studies with respect to control strategies for temperature control or to be of interest for plant design. The goal of this study is to analyze and develop an educational plant that can be used at different application laboratory where students study about microcontrollers, data acquisition, system identification and especially control system design. Plant implementation can easily be reused by users who are not expert in plant design. The laboratory architecture proposed in this paper permits to run experiments while interacting with its components. The control architecture allows for the users to implement their knowledge of control engineering in an easy way due to process access by a friendly design. The designed application with PIC18F4620 is also useful in predictive control research for embedded controller [3]. Moreover, in recent years, the requirements for the quality of control design in process increased due to the computing power high complexity [4].

The paper is organized as follows: the second part of the paper presents the cooling system design and implementation with PIC18F4620. The next section deals with the data acquisition and model identification. In the third part of the paper a control system design based on PID controller was developed. Also, some real time experiments have been performed in order to illustrate the performances of the implemented controller. Finally some conclusions are given.

II. LOW COST APPLICATION FOR TEMPERATURE CONTROL USING PIC18F4620

Large scales of application are dedicated to control the temperature in a realistic environment suitable to various needs. In order to obtain the optimum performance it would be advantageous to provide a temperature control structure by providing a safer cooling or heating system with better performances in terms of energy efficiency, flexibility and portability. Extensively research has been made on heating control because of the necessity in practical applications [5]. In this paper the idea focuses on a simple process that could be suitable to educational application in order to illustrate different control design aspects. Moreover, the designed plant is a low cost application with components easy of access. The plant consists in a cooler, a resistance, a sensor and a microcontroller that are encapsulated in a

This work was supported by CNCSIS-UEFISCSU, project number PN II-RU PD cod 331/2010.

small testing room, as is shown in the Fig. 1. The analog temperature sensor LM335 will offer information of the current state of degrees at every sampling time.

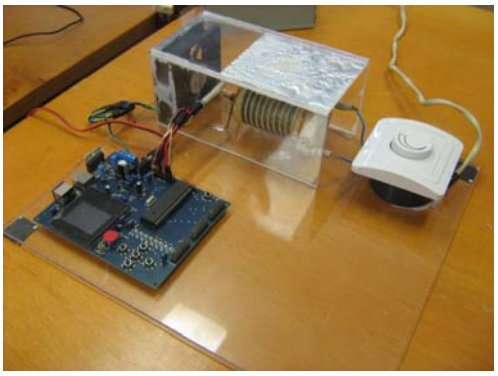


Fig.1 Low cost implementation of the cooling system

To relieve the performance of the proposed cooling system it is necessary to increase the test room temperature with an electrical heat resistor made of nickel. The heat discharge in the small test room will be fixed by limiting the value of the current with a variable voltage actuator for alternative current. The function of the cooler will be to decrease the air temperature from the test room.

The microcontroller will send a signal to L298 a H-bridge, who will command the DC motor of the cooler as is depicted in Fig. 2. In order to adjust the air temperature to the desired set point, a controller is needed. It is to be mentioned that the experiment starts from a fixed temperature in the testing room. The heat resistor is made by nickel and emits a heat quantity in the environment. To avoid a high temperature degree, the current range of the resistor is limited by a voltage potentiometer to obtain a 77 Celsius degree in the conceived test room. This part of the system used a 220V external source.

One of the active elements of the system is the PC cooler. Using the L298 commands, the motor will spin respecting DC motor speed control strategy. The H Bridge L298 will assure the alimentation for the cooler DC motor. The motor speed is varying from 950 to 3600 rotation per minute offering a maximum 44.3 CMF airflow.

The LM335 analog sensor is a useful transducer for the system. The sensor operates form -40 C degrees to 100 C degrees and in this application is used the plastic TO-92 packages sensor, considering basic temperature. With one degree precision and easy to calibrate using a 10 kOhm resistor it offers the actual K (Kelvin) degrees. The K degree will be converted in Celsius degree, in order to send the information to the microcontroller.

The proposed development board used in this paper named MDB01 (motherboard) produced by SMTD (Smart Tech Design) was specially designed for application with low cost microcontrollers like Microchip 8 bits 16xxx and 18xxx, with 40 pins, family. This board contains the PIC 18F4620 which can significantly reduce power consumption during operation. The 40Mhz operating frequency combined with 13 input channels for the 10-bit Analog-to-Digital Module, 65536 Bytes of Program Memory and 1024 Bytes for Data EEPROM are a few of the features of 18F4620. The microcontroller uses a 75 instruction set and 83 extended instruction set enabled. The device permits enhanced USART Serial Communication and also admits 5 I/O ports and 4 Timers.

The communication to the computer is made with a USB cable which adapts the signal to serial communication with a hardware device FT232RL, as illustrated in [2]. This facility offers the possibility to establish connections with all modern PC or notebook.

MDB01 is equipped with L298 Dual Full Bridge, providing the command actions. It is a high voltage, high current dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepper motors. Two enable inputs are provided to enable or disable the device independently of the input signals. In this application it is used a single input due to one single DC motor. The emitters of the lower transistors of each bridge are connected together and the corresponding external terminal can be used for the connection of an external resistor. An additional supply input is provided so that the logic works at a lower voltage. A user friendly part of the process is the color LCD S1D15G14 with 98x67 pixels interfaced with 8 bits with 18F4620 which is able to display in real-time the temperature degree from the sensors.

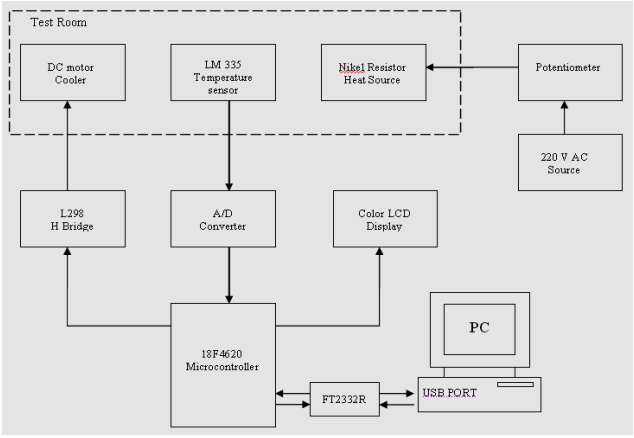


Fig. 2 The process scheme

The developed low cost application is dedicated in order to illustrate different control strategies for temperature control subject to cooling phenomenon. In the next section, the attention is focused on real time data acquisition, followed by the open loop model identification.

III. MODEL IDENTIFICATION AND CONTROL DESIGN FOR THE COOLING SYSTEM USING PIC 18F4620

The study is dedicated to the educational activities in the field of control system design. The general purpose of the process is to ensure a desired temperature in a closed loop safety operational functionality and for understanding the dynamics of a cooling system. The basic information about the process description is given in section II. The aim of this section is to illustrate how to acquire real time data from the process, the model identification and control system methods suitable for air temperature control. The system is linear, discrete-time and single input-single output. The input signal considered is the duty cycle of the PWM signal and the output is represented by the air temperature value measured from the LM335 analog sensor.

The system transfer functions are assumed to be functions in continuous time domain.

Acquisition phase represents an important step in model identification of the dynamics of the plant that requires accuracy in sending input commands to the system and precise measurement of the output values. This involves a perfect timing between the command action and output signal. Data values, command signal and output offer a great possibility to create a complex or simplified model for the plant. Communication channels must be synchronized and safeguards must action to the potential external disturbances. The better model obtained the greater performance control action is performed. However, the distribution of the temperature depends on a lot of factors which influences the type and the parameters of the model. The control action is given by the duty cycle values of the PWM signal which are applied to the cooling DC motor.

The control command from the microcontroller is given using a MicroC software application, the code being written in C++. In order to obtain the 12 V at the DC motor it is mandatory to use a suitable frequency to L298 bridge. The bridge was used to compensate the PIC18F4620 output voltage which was limited to 5 V, because the PIC voltage reference is given by the PC USB port. The L298 bridge can offer the 12V for maximizing the DC motor performance. It is to be mentioned that the L298 H-Bridge replaces the standard D/A converter. After setting the DC motor operation frequency value to 2.44 kHz, different duty cycle values were tested. In the system identification phase, a step input from 20% to 90% duty cycle was applied.

In order to acquire a representative data set it was assumed that in the testing room a 77 C degree was settled. At this constant ambient temperature value the experiments started. First it was applied a command action to the motor at the 20% of the duty cycle for 4 minutes. In this time the output values of the LM335 sensor were collected each second and saved in the EEPROM memory. This quick memory can record very accurate the A/D sent values. A number of samples in C degrees were recorded in this time.

Stationary regime is settled in about 4 minutes, afterwards a new command is sent by the microcontroller to the L298 bridge. This forces the motor to work at a 90% duty cycle and the air temperature in the test room decreases. The new temperature values were recorded after the last output value given by the 20% duty cycle command. It can be mentioned that the 90% duty cycle command was applied using 1 second sampling time. The values from the EEPROM memory were sent to a PC using a USB communication cable and all output values were saved in a text file. FT232RL interface helps to convert the serial transmitter of the 18F4620 into a large scale USB interface who acts like a serial communication channel.

The frequency for the DC cooling motor was tested during the acquisition stage to set up the pulse width period in the time of the experiments. After testing how the system works at different PWM frequency a 2.44 kHz was established. Some tests were made in the test room varying the DC motor cooling power. The PWM command was observed on the oscilloscope as shown in Fig.3.

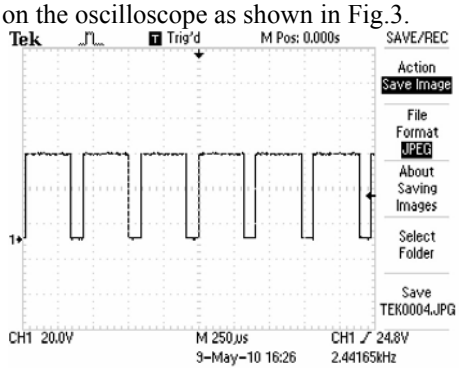


Fig.3 Oscilloscope 2.44 kHz PWM

In order to identify the parametrical model of the cooling process the Identification Toolbox from Matlab was used. The cooling process can be described as a continuous process but we can control it like a discrete time system. For a good description of the process we need to create a model of the process.

The communication of the system with a computer is necessary to transfer the information from the LM335 sensor to the computer. Using the USB cable, the voltage values from the sensor were read by the analog to digital converter at every second and after converting to C degree the information were send to the PC. To found the model of the plant some relevant information need to be stored. An open loop response can offer necessary information to build a real model that can be processed in the Identification Toolbox from Matlab environment. A step will be applied to the system and the output response will be saved in Matlab workspace. The input for the system can be considered the duty cycle value of the PWM and will be considered in percent values. Some different values

of the step were applied to the system and the responses are plotted in the Fig. 4 and Fig.5 .

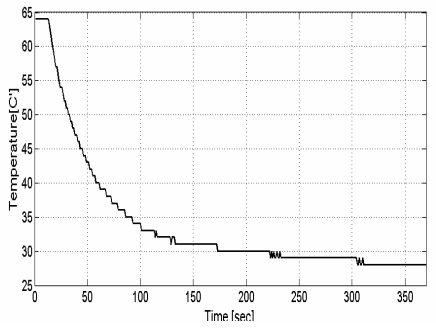


Fig 4. Step response at 50% duty cycle

From the existing responses the last set of data was used to identify the continuous time model in Matlab sofware. In the Identification Toolbox, the command and the output response was used to create a model. Sampling period was also introduced for the model realization.

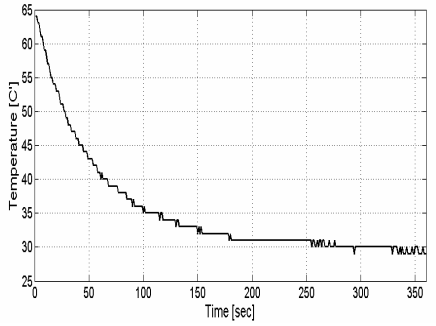


Fig 5. Step response at 90% duty cycle

The toolbox offers the possibility to create a continuous time model. From Process Models menu one would start to build some simple models which reflect the dynamic plant. The model obtained is based on Levenberg Marquardt search method. The Levenberg-Marquardt (LM) algorithm is an iterative technique that locates the minimum of a function that is expressed as the sum of squares of nonlinear functions. Levenberg-Marquardt (LM) algorithm is the most widely used optimization algorithm. It outperforms simple gradient descent and other conjugate gradient methods in a wide variety of problems [6]. A simple model was found after some simulation tests and the continuous time mathematical model of first order is:

$$G_f(s)=\frac{k_f}{1+sT_f}e^{-sL}, \qquad (1)$$

with the process constant time $T_f = 45.947$, amplification factor $k_f=0.325$ and the dead time $L=20.947$ measured in samples periods of 1 sec.

In the next paragraph, a PID controller is design with the parameters tuned based on the experimental methods. The continuous time model (1) is used in order to create a PID controller.

IV. THE CONTROL SYSTEM DESIGN FOR TEMPERATURE CONTROL USING PIC18F4620

In this study a parallel form of a proportional–integral–derivative controller was implemented. The form of the PID controller in the continuous time domain is expressed as:

$$U(s)=K_p\cdot[1+\frac{1}{s\cdot T_i}+s\cdot T_d]\cdot E(s), \qquad (2)$$

where $E(s)$ is the signal error and $U(s)$ is the control input to the process. K_p is the proportional gain, T_i is the integral time constant, and T_d is the derivative time constant.

The discrete time domain form of the controller became:

$$U(z)=E(z)\cdot K_p\cdot\left[1+\frac{T_s}{T_i\cdot(1-z^{-1})}+T_d\cdot\frac{(1-z^{-1})}{T_s}\right], \qquad (3)$$

For the discrete PID, the parameters were calculated with the following relations:

$$\begin{aligned} Kp_{discrete}&=K_p \\ Ti_{discrete}&=\frac{K_p\cdot T_s}{T_i}, \\ Td_{discrete}&=\frac{K_p\cdot T_d}{T_s} \end{aligned} \qquad (4)$$

The realization of the PID controller was based on several experimental methods: Ziegler-Nichols method based on step response, Chien-Hrones-Reswick method and the Cohen-Coon method [7]. Using Matlab Simulink Toolbox the parameters obtained from each method where tested and the best parameters were used for PID implementation on the microcontroller. The tunning parameters based on Ziegler-Nichols method are: proportional gain $K_p=6.246$, integral time $T_i=61.113$, derivative time $T_d=0$. The discrete time values for sampling period $T_s=1\,\text{sec}$ are: $Kp_{discrete}=6.246$, $Ti_{discrete}=0.102$. These values were tested in a real time application using the microcontroller for cooling the temperature in the test room from initial condition of 57 C degree to a new reference of 30 C degree.

Using Chien-Hrones-Reswick method results the following parameters: proportional gain $K_p=2.429$, integral time $T_i=55.136$, derivative time $T_d=0$. The discrete time values using sampling period $T_s=1\text{sec}$ are: $Kp_{discrete}=2.429$, $Ti_{discrete}=0.044$. The continuous time parameters from Chien-Hrones-Reswick method were implemented considering as performances the zero overshoot and minimum time response.

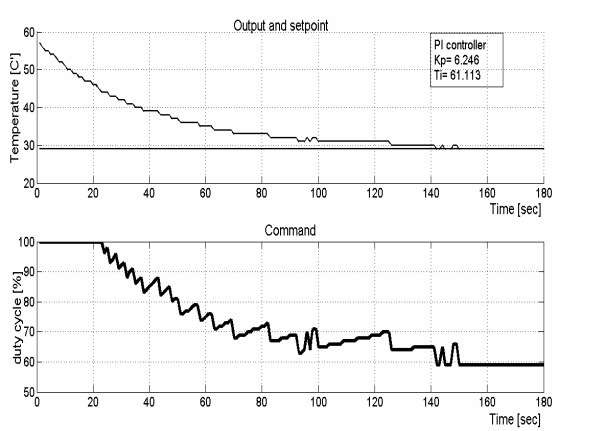


Fig 6. Ziegler-Nichols PI real time response

In order to get better performances the tuning parameters were slightly modified such as $K_p=5$, and integral time $T_i=40$, resulting the discrete time values $Kp_{discrete}=5$, $Ti_{discrete}=0.125\cdot$

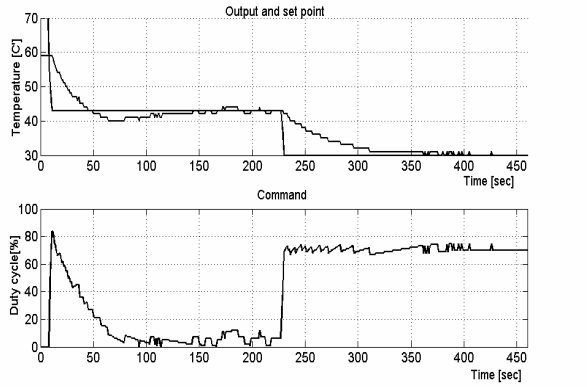


Fig.7 Chien-Hrones-Reswick PI reference changing

With these modified values a new real time temperature control has been performed, subject to the set point changes.

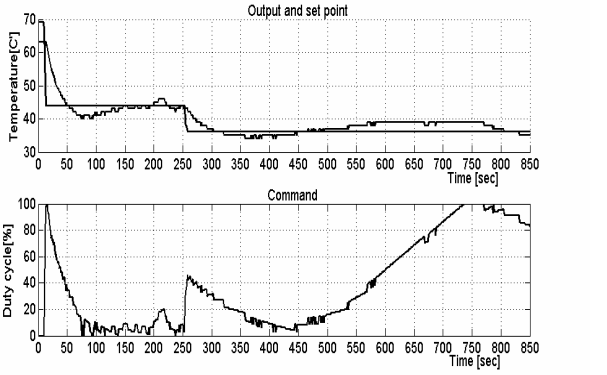


Fig 8. Chien-Hrones-Reswick PI disturbance rejection

The next experiment starts with a initially temperature of 59 C degree in the test room the reference was settled to 43 C degree. A new step value in set point was set to 30 C degree. After the transitory regime the PID controller reach the set point as is shown in the Fig. 7.

The controller is able to reject the load disturbance effect if in the test room the temperature suddenly increased.

The real time experiment demonstrates the fact that the proposed control system design is able to reject the disturbance, a 120V voltage on nickel resistor, the action control compensating the heat quantity as is illustrated in the Fig. 8.

V. CONCLUSIONS

In this paper a low cost application for temperature control in a ventilation system using the PIC18F4620 was designed and developed. The laboratory architecture proposed in this paper permits to run experiments while interacting with its components. The control architecture allows for the users to implement their knowledge of control engineering in an easy way due to process access by a friendly design. Some PID tuning methods have been tested and adjustments have been made to the parameters in order to obtain better performances. The implemented PID controller on PIC18F4620 can offer zero steady state error even if a load disturbance is introduced in the test room. In the cases where the plant response is slow, it may be possible to decrease the processor speed and save power. In Future work includes research of some advanced control strategies and the implementation on this low cost plant, which may reflect better control actions in terms of small sampling rate.

REFERENCES

[1] Tiebao Yang, Xiang Chen, Henry Hu A Fuzzy PID Thermal Control for Die Casting Processes 22nd IEEE International Symposium on Intelligent Control.

[2] Dogan Imbrahim Microcontroller Based Temperature Monitoring and Control ,September 2008.

[3] Kadirkamanathan V., Halaucă C and Anderson S. *Predictive control of fast-sampled systems using the delta-operator*, International Journal of Systems Science, 40:7, 745-756, 2009.

[4] Bouhenchir, H., Cabassud, M., Le Lann, M. V., Casamatta, G. A general simulation model and a heating–cooling strategy to improve controllability of batch reactors. *Trans IchemE*, Part A, 79, 641–654, 2001.

[5] Weijun Yu; Xianyi Qian. The Constant Temperature Automatic Control System Design of 3G Base Station without Man's Guard. Proceedings of 2009 International Conference on Information Engineering and Computer Science, ICIECS 2009.

[6] Manolis I. A. Lourakis, A Brief Description of the Levenberg-Marquardt Algorithm Implemened by Levmar 2005.

[7] Corneliu Lazar, Draguna Vrabie, Sorin Carari, Sisteme automate cu regulatoare PID, Ed. MatrixRom, Bucuresti 2004.

Some Remarks on the Fragility of PD Controllers for SISO Systems with I/O Delays

Bogdan Liacu* , Cesar Mendez-Barrios, Silviu-Iulian Niculescu
L2S (UMR CNRS 8506), CNRS SUPELEC Univ Paris Sud,
3 rue Joliot Curie, 91192, Gif-sur-Yvette, France.
{Bogdan.Liacu, Cesar.Mendez,
Silviu.Niculescu}@lss.supelec.fr

Sorin Olaru
E3S, Departement Automatique, SUPELEC,
3 rue Joliot Curie, 91192, Gif-sur-Yvette,
France.
Sorin.Olaru@supelec.fr

***Abstract*— This paper focuses on the fragility analysis of Proportional-Derivative(PD) controllers for single-input-single-output (SISO) systems affected by input (or output) delays. Using a geometric approach, we present an method to analyze the fragility of PD controllers and to provide practical guidelines for the design of non-fragile PD controllers.**

Keywords: PD-controller, Stability, Fragility, Delay.

I. INTRODUCTION

According to the literature [3], [23], [24], [25], proportional-derivative(PD) controllers are largely used in teleoperation systems. For such systems, a very fast response is required and, in most of the cases, they are affected by communication delays.

This paper focuses on the fragility of PD controllers for single-input-single-output (SISO) systems in the presence of input/output (I/O) delays. In order to develop controllers for real environments, a careful analysis must be taken into account for the variation of the parameters. Generally, real systems present parameter variations which often lead the system to an unstable behavior. In our opinion, the notions of controller *fragility* is the best appropriate for such a study, see for instance, [1], [14], [15] for such a notion and some preliminary results.

It is worth mentioning that the problem received a lot of attention in the case free of delay systems, see for example [14] (robustness techniques design leading to fragile controllers), [13] (non-fragile PID control design procedure), [1] (appropriate index to measure the fragility of PID controllers), but the delay case was not sufficiently addressed. In the sequel, we develop a *simple method* to analyze the fragility of a given PD-controller, for any SISO strictly proper system affected by input/output (constant) time-delays. The results are presented by using a *geometrical approach.*

The method proposed in the paper consists in two steps. More precisely, the first step uses the D-decomposition method suggested by Neimark [21] (see [3], [17] for further comments) in order to derive and compute the boundaries of the stability regions. In this context, the *stability crossing curves* represent the collection of all points for which the corresponding characteristic equation of the closed-loop system has at least one root on the imaginary axis. These curves define a partition of the space of parameters in several regions, each region having a constant number of unstable roots for all the parameters inside the region. By taking into account the crossing boundaries characterization in the controller parameter-space, the second step consists in deriving an algorithm to determine explicitly the *optimal non-fragile* controller. The presented algorithm allows to explicitly compute the (closed-loop) stability radius in the controller parameter space. Finally, as a by-product of the analysis, we can easily derive the maximum controller gain interval guaranteeing the closed-loop stability for a prescribed derivative coefficient.

The remaining paper is organized as follows: in section 2 some preliminary results are briefly presented. Next, the fragility algorithm for PD controllers is described in Section 3. Illustrative examples as "gantry crane" are considered in Section 4. Finally, some concluding remarks end the paper.

Throughout the paper the following notations are used: $\mathbb{C}$ is the set of complex numbers, $j = \sqrt{-1}$. For $z \in \mathbb{C}$, $\angle z \in [0, 2\pi)$, $\Re(z)$ ($\Im(z)$): argument, real (imaginary) part of z. Let $x, y \in \mathbb{C}^n$, the inner product will be denoted by $\langle x, y\rangle = x^* y$, where x^* is the complex conjugate transpose of y. By using $\emptyset$ we denote the empty set and by $\mathbb{R}_+$ we define the set of positive real numbers.

II. PRELIMINARY RESULTS

Consider the class of strictly proper SISO open-loop system with I/O delays given by the transfer function:

$$H_{yu}(s) = \frac{P(s)}{Q(s)} e^{-s\tau} = c^T (sI_n - A)^{-1} b e^{-s\tau}, \qquad (1)$$

where (A, b, c^T) is a state-space representation of the open-loop system ($A \in \mathbb{R}^{n\times n}$, $b, c \in \mathbb{R}^n$). The control law is defined by a classical PD controller K(s) of the form:

$$K(s) = k(1 + T_d s) = k_p + k_d s. \qquad (2)$$

*Corresponding author, Tel: +33 (0)1 69 85 13 79, Fax: + 33 (0)1 69 85 13 89; the author is also with E3S, Departement Automatique, SUPELEC, 3 rue Joliot Curie, 91192, Gif-sur-Yvette, France, and CEA, LIST, Interactive Robotics Laboratory BP 6, 18 route du Panorama, F-92265 Fontenay-aux-Roses, France.

Remark 1: For analytical reasons, we will consider the case $k_p = 0$, although such a case does not have a practical counterpart, since standard derivative controllers are not applicable.

Therefore, the stability of the closed-loop systems is given by the locations of the zeros of the following meromorphic function $H : \mathbb{C} \times \mathbb{R}^2 \times \mathbb{R}_+ \mapsto \mathbb{C}$ given by:

$$H(s; k_p, k_d, \tau) = 1 + \frac{P(s)}{Q(s)}(k_p + k_d s)e^{-s\tau}, \qquad (3)$$

which has an infinite (but, countable) number of roots (see e.g., [9], [11]).

As mentioned in the introduction, the aim of this paper is to derive an appropriate PD controller (k_p^*, k_d^*) and a positive value d such that the control law(2) stabilizes the system (1) for any k_p and k_d as long as:

$$\sqrt{(k_p - k_p^*)^2 + (k_d - k_d^*)^2} < d. \qquad (4)$$

For the brevity of the paper and without any loss of generality, we make the following:

Assumption 1: The polynomials $P(s)$ and $sQ(s)$ in (3) do not have common zeros.

In the sequel, we recall some geometric results that enable us to generate the stability crossing curves in the space defined by the controller's parameters (k_p, k_d) (similar results for different types of dynamics can be found in [10] - delay parameters space and [18], [19] - some particular class of distributed delays). These curves represent the collection of all pairs (k_p, k_d) for which the characteristic equation (3) has at least one root on the imaginary axis of the complex plain.

According to the continuity of zeros with respect to the system's parameters (see, for instance, [4] for the continuity with respect to delays), the number of roots in the right half plane (RHP) can change only when some zeros appear and cross the imaginary axis. Therefore, a useful concept is the frequency crossing set Ω defined as the set of all real positive ω for which there exist at least a pair (k_p, k_d) such that:

$$H(j\omega; k_p, k_d, \tau) = 1 + \frac{P(j\omega)}{Q(j\omega)}(k_p + k_d j\omega)e^{-j\omega\tau} = 0. \quad (5)$$

We only need to consider positive frequencies ω, that is $\Omega \subset (0, \infty)$ since obviously,

$$H(j\omega; k_p, k_d, \tau) = 0 \iff \overline{H(-j\omega; k_p, k_d, \tau)} = 0. \quad (6)$$

Proposition 1: For a given $\tau \in \mathbb{R}_+$ and $\omega \in \Omega$ the corresponding crossing point (k_p, k_d) is given by:

$$k_p = -\Re\Big(\frac{Q(j\omega)}{P(j\omega)}e^{j\omega\tau}\Big), \qquad (7)$$

$$k_d = -\frac{1}{\omega}\Im\Big(\frac{Q(j\omega)}{P(j\omega)}e^{j\omega\tau}\Big). \qquad (8)$$

Remark 2: It is easy to see that $\forall\ \omega \in \Omega$ we have $P(j\omega) \neq 0$. Otherwise, $Q(j\omega)= 0$, that contradicts Assumption 1.

Proposition 2: Let the *relative degree* of the system (1) be $\rho = 1$. Then, the closed-loop system (1) becomes of Neutral-Type and,

$$\begin{cases} k_p = \left|\frac{q_n}{p_{n-1}}\right| \\ k_p = -\left|\frac{q_n}{p_{n-1}}\right| \end{cases},$$

belong to the stability crossing boundary. Here p_{n-1} and q_n are the main coefficients of the polynomials $P(s)$ and $Q(s)$, respectively:

$$P(s) = \sum_{i=1}^{n-1} p_i s^i, \quad Q(s) = \sum_{i=1}^{n} q_i s^i.$$

Remark 3: Observe that in the case when $m = n - 1$, the corresponding closed-loop system is a quasipolynomial of *neutral* type (see, for instance,[22] for further discussions on the topics).

Proposition 3: [20]: Let k_d^* and $k_d^* > 0$ be given. Let $\Omega_{k_p^*, k_d^*}$ denotes the set of all frequencies $\omega > 0$ satisfying equation (5) for at least one pair of (k_p, k_d) in the rectangle $|\ k_p\ | \leq k_p^*, |\ k_d\ | \leq k_d^*$. Then $\Omega_{k_p^*, k_d^*}$ consists of a finite number of intervals of finite length. Precisely, $\omega \in \Omega_{k_p^*, k_d^*}$ if and only if:

$$\left|\frac{Q(j\omega)}{P(j\omega)}\right|^2 \leq (k_p^*)^2 + (k_d^*)^2\omega^2. \qquad (9)$$

Then, when ω varies within some interval Ω_l satisfying the inequality (9), (7)-(8) define a continuous curve. Denote this curve by $\mathcal{T}_l$ and consider the following decompositions:

$$R_0 + jI_0 = j\frac{\partial H(s; k_p, k_d, \tau)}{\partial s}\Big|_{s=j\omega}, \qquad (10)$$

$$R_1 + jI_1 = -\frac{\partial H(s; k_p, k_d, \tau)}{\partial k_d}\Big|_{s=j\omega}, \qquad (11)$$

$$R_2 + jI_2 = -\frac{\partial H(s; k_p, k_d, \tau)}{\partial k_p}\Big|_{s=j\omega}. \qquad (12)$$

The implicit function theorem indicates that the tangent of $\mathcal{T}_l$ can be expressed as follows:

$$\begin{aligned} \begin{pmatrix} \frac{dk_p}{d\omega} \\ \frac{dk_d}{d\omega} \end{pmatrix} &= \begin{pmatrix} R_2 & R_1 \\ I_2 & I_1 \end{pmatrix}^{-1} \begin{pmatrix} R_0 \\ I_0 \end{pmatrix} \\ &= \frac{1}{R_2 I_1 - R_1 I_2} \begin{pmatrix} R_1 I_0 - R_0 I_1 \\ R_0 I_2 - R_2 I_0 \end{pmatrix}, \end{aligned} \qquad (13)$$

provided that

$$R_1 I_2 - R_2 I_1 \neq 0. \qquad (14)$$

In order to derive the stability region of the system given by (3), [20] characterized the smoothness of the crossing curves and the corresponding direction of crossing.

Proposition 4: The curve $\mathcal{T}_l$ is smooth everywhere except possibly at the point corresponding to $s = j\omega$ is a multiple solution of (3).

A. Direction of Crossing

The next paragraph focuses on the characterization of the crossing direction corresponding to the curves defined by (7)-(8). We will call the direction of the curve that corresponds to increasing ω the *positive direction*. We will also call the region on the left hand side as we head in the positive direction of the curve *the region on the left*.

Proposition 5: Assume $\omega \in \Omega_l$, k_p, k_d satisfy (7) and (8) respectively, and ω is a simple solution of (6) and:

$$H(j\omega'; k_p, k_d, \tau) \neq 0, \forall \omega' \neq \omega, \tag{15}$$

(i.e. (k_p, k_d) is not an intersection point of two curves or different section of a single curve). Then, as (k_p, k_d) moves from the region on the right to the region on the left of the corresponding crossing curve, a pair of solution of (3) crosses the imaginary axis to the right (through $s = j\omega$) if

$$R_1 I_2 - R_2 I_1 > 0. \tag{16}$$

The crossing is to the left if the inequality is reversed. Any given direction, (d_1, d_2), is to the left-hand side of the curve if its inner product with the left-hand side normal $\left(-\frac{\partial k_d}{\partial \omega}, \frac{\partial k_p}{\partial \omega}\right)$ is positive, i.e.,

$$-d_1 \frac{\partial k_d}{\partial \omega} + d_2 \frac{\partial k_p}{\partial \omega} > 0, \tag{17}$$

from which we have the following result.

Corollary 1: Let ω, k_p and k_d satisfy the same condition as Proposition 5. Then as (k_p, k_d) crosses the curve along the direction (d_1, d_2), a pair of solutions of (3) crosses the imaginary axis to the right if

$$d_1(R_2 I_0 - R_0 I_2) + d_2(R_1 I_0 - R_0 I_1) > 0. \tag{18}$$

The crossing is in the opposite direction if the inequality is reversed.

III. MAIN RESULT: FRAGILITY OF PD CONTROLLERS

Consider now the PD fragility problem, which is the problem of computing the maximum controller parameters deviation without losing the closed-loop stability - given the pair of parameters (k_p^*, k_d^*) such that the roots of the equation:

$$Q(s) + P(s)(k_p + k_d s)e^{-s\tau} = 0, \tag{19}$$

are located in $\mathbb{C}_-$ (that is the closed-loop system is asymptotically stable), find the maximum parameter deviation *d* $\in \mathbb{R}_+$ such that the roots of (3) stay located in $\mathbb{C}_-$ for all controllers (k_p^*, k_d^*) satisfying:

$$\sqrt{(k_p - k_p^*)^2 + (k_d - k_d^*)^2} < d. \tag{20}$$

This problem can be more generally reformulated as: *find the maximum parameter deviation d such that the number of unstable roots of (3) remains unchanged.*

First, let us introduce some notation:

$$\mathcal{T} = \bigcup_{l=1}^{N} \mathcal{T}_l, \qquad \mathcal{T} = \{(k_p, k_d) | \omega \in \Omega_l\}, \tag{21}$$

$$\overrightarrow{k(\omega)} = (k_p(\omega), k_d(\omega))^T, \qquad \overrightarrow{k^*} = (k_p^*, k_d^*)^T. \tag{22}$$

Let us also denote $d_{\mathcal{T}} = \min\limits_{l \in \{1,\dots,N\}} d_l$, where:

$$d_l = min\Big\{\sqrt{(k_p - k_p^*)^2 + (k_d - k_d^*)^2} | (k_p, k_d) \in \mathcal{T}_l\Big\}. \tag{23}$$

With the notation and the results above, we have:

Proposition 6: The maximum parameter deviation from (k_p, k_d), without changing the number of unstable roots of the closed-loop equation (3) can be expressed as:

$$d = min\left\{k_{d\infty}, |k_p^* - k_p(0)|, \min_{\omega \in \Omega_f} \left\{\left\|\overrightarrow{k(\omega)} - \overrightarrow{k^*}\right\|\right\}\right\}, \tag{24}$$

where

$$k_{d\infty} := \begin{cases} \min\left\{\left|k_d^* - \left|\frac{q_n}{p_m}\right|\right|, \left|k_d^* + \left|\frac{q_n}{p_m}\right|\right|\right\} & \text{if} \quad m = n-1 \\ \emptyset & \text{if} \quad m < n-1 \end{cases}$$

and Ω_f is the set of roots of the function $f : \mathbb{R}_+ \mapsto \mathbb{R}$,

$$f(\omega) \triangleq \left\langle \left(\overrightarrow{k(\omega)} - \overrightarrow{k^*}\right), \frac{d\overrightarrow{k(\omega)}}{d\omega} \right\rangle. \tag{25}$$

Proof: We consider that the pair (k_p^*, k_d^*) belongs to a region generated by the crossing curves. Since the number of unstable roots changes only when (k_p, k_d) get out of this region, our objective is to compute the distance between (k_p^*, k_d^*) and the boundary of the region. Furthermore, the boundary of such a region consists of "pieces" of crossing curves and possibly the segments of the shifted axis:

$$k_p := k_p + \left|\frac{q_n}{p_m}\right|, \quad \text{and}$$
$$k_p := k_p - \left|\frac{q_n}{p_m}\right|,$$

(for neutral-type systems) and a segment of the shifted axis:

$$kd + kp(0).$$

In order to compute the distance between (k_p^*, k_d^*) and a crossing curve we only need to identify the points where the vector ($k_p - k_p^*$, $k_d - k_d^*$) and the tangent to the curve are orthogonal.

In other words we have to find the solutions of:

$$f(\omega) = 0,$$

where f is defined by (25).

Taking into account the relation (13), we may write (25) as:

$$f(\cdot) = (k_p - k_p^*)(R_1 I_0 - R_0 I_1) - (k_d - k_d^*)(R_0 I_2 - R_2 I_0). \tag{26}$$

It is noteworthy that $f(\omega)$ is a polynomial function and, therefore, it will have a finite number of roots. Let us consider $\{\omega_1, ..., \omega_M\}$ the set of all the roots of $f(\omega)$ when we take into account all the pieces of crossing curves belonging to the region around (k_p^*, k_d^*). Since the distance from (k_p^*, k_d^*) to the $k_p(\omega)$ axis is given by $|k_d|$, one obtains:

$$d = min\left\{k_{d\infty}, |k_p^* - k_p(0)|, \min_{h=1,...,M}\left\{\left\|\overrightarrow{k(\omega_h)} - \vec{k^*}\right\|\right\}\right\}, \quad (27)$$

that is just another way to express (24).

The explicit computation of the maximum parameter deviation d can be summarized by the following algorithm:

Step 1: First, compute the "degenerate" points of each curve $\mathcal{T}_l$ (i.e. the roots of $R_1 I_2 - R_2 I_1 = 0$ and the multiple solutions of (3)).

Step 2: Second, compute the set Ω_f defined by *Proposition* 6 (i.e. the roots of equation $f(\omega) = 0$, where f is given by (25).

Step 3: Finally, the corresponding maximum parameter deviation d_l is defined by (23).

Remark 4: (On the gains' optimization): It is worth mentioning that the geometric argument above can be easily used for solving other robustness problems. Thus, for instance, if one of the controller's parameters is fixed (prescribed), we can also explicitly compute the maximum interval guaranteeing closed-loop stability with respect to the other parameter. In particular if T_d ("derivative") is fixed, we can derive the corresponding stabilizing maximum gain interval.

IV. Numerical Examples

In this section we present numerical examples.

Example 1: (Gantry crane) We have chosen a gantry crane model with a time delay of 2 seconds used as slave robot in a teleoperation system as suggested by Fernandez *et al.* in [5]:

$$g(s) = \frac{40s^2 + 2s + 400}{200s^3 + 30s^2 + 2401s + 200} e^{-2s}. \quad (28)$$

According to *Proposition 1*, Fig.1 presents the corresponding crossing curves for (28).

Let us define:

$$k_{p0} := |k_p^* - k_p(0)|$$

In table I, we summarize the results obtained after applying the proposed algorithm to analyze the fragility for the controller $(k_p^*, k_d^*) = (3.25, 1.65)$.

Figure 2 illustrates the stability region for the system (28) as well as the *maximum parameter deviation* for the controller (k_p^*, k_d^*).

Example 2 (Sixth order non-minimal phase system): For the second example we consider the following process [2]:

$$g(s) = \frac{-s^4 - 7s^3 - 2s + 1}{(s+1)(s+2)(s+3)(s+4)(s^2+s+1)} e^{-\frac{s}{20}}. \quad (29)$$

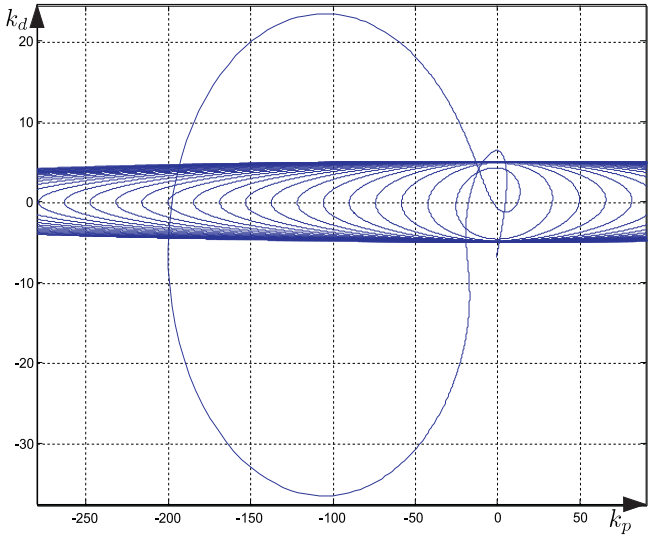


Fig. 1. Corresponding crossing curves for (28).

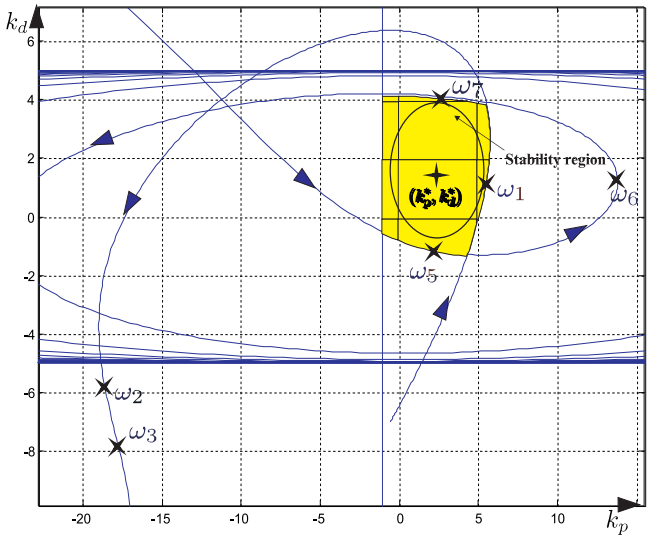


Fig. 2. Stability region of k_p and k_d for (28). $k_p^* = 3.25$, $k_d^* = 1.65$, $d = 2.345980$.

Frequency	$d_{\mathcal{T}_l}$	$k_{d\infty}$	k_{p0}	$\min\{d_T, k_{d\infty}, k_{p0}\}$
$\omega_1 = 0.9017$	2.3459			
$\omega_2 = 2.7292$	23.0540			
$\omega_3 = 2.8228$	23.0158			
$\omega_4 = 3.1625$	203.161			
$\omega_5 = 3.5744$	2.8603			
$\omega_6 = 4.1134$	10.6317	3.35	3.75	2.34598084836201
$\omega_7 = 4.6386$	2.4229			
$\omega_8 = 5.5736$	28.3525			
$\omega_9 = 6.3485$	6.2916			
$\omega_{10} = 7.1127$	30.5030			
$\omega_{11} = 7.8169$	3.13656			

TABLE I

Parameters deviation results without losing the stability for system (28).

In Fig.3 we present the stability region in k_p, k_d for (29).

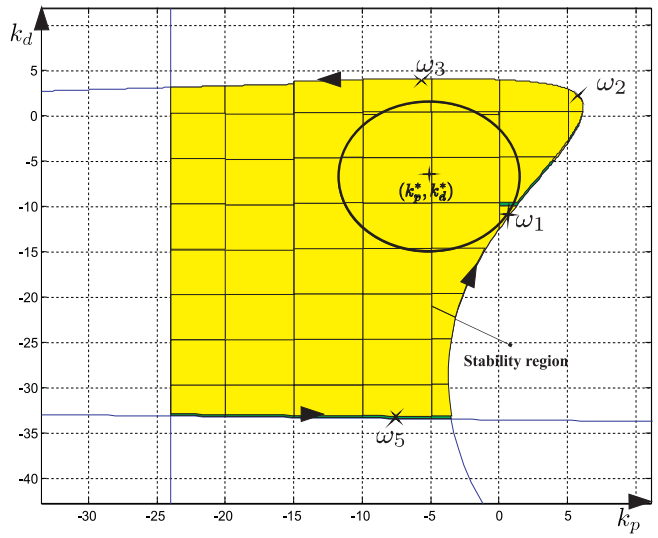


Fig. 3. Stability region of k_p and k_d for (29). $k_p^* = -6.15$, $k_d^* = -6.25$, $d = 8.2917$.

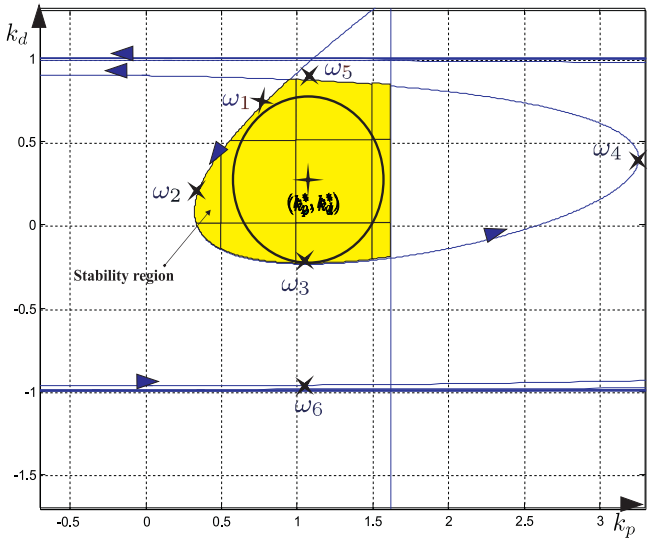


Fig. 4. Maximum parameter deviation $d = 0.4910$ for the controller $(k_p^*, k_d^*) = (1.1021, 0.2514)$.

The fragility analysis for the PD-controller for the system (29) are summarized in table II.

Frequency	$d_{\mathcal{T}_l}$	$k_{d\infty}$	k_{p0}	$\min\{d_T, k_{d\infty}, k_{p0}\}$
$\omega_1 = 0.9352$	8.2917			
$\omega_2 = 1.9489$	14.3176			
$\omega_3 = 3.9725$	9.8893			
$\omega_4 = 23.2533$	288.344		17.85	8.291787839
$\omega_5 = 33.7137$	27.1850			
$\omega_6 = 73.7534$	4792.91			
$\omega_7 = 95.1040$	101.249			

TABLE II

PARAMETERS DEVIATION RESULTS WITHOUT LOSING THE STABILITY FOR SYSTEM (29).

Example 3 (Unstable, non-minimum phase): Consider a second-order, non-minimum-phase and unstable open-loop system, described by the transfer function [16]:

$$g(s) = \frac{s-2}{s^2 - 1/2s + 13/4} e^{-\frac{1}{2}s}. \qquad (30)$$

The particularity of this system is that the use of a PD-controller lead to a closed-loop system of neutral type. In Fig.4 we have depicted the stability region in the (k_p, k_d) parameter space for the system (30).

In table III we summarize the fragility analysis for the controller $(k_p^*, k_d^*) = (1.1021, 0.2514)$ for the system (30).

V. CONCLUSIONS AND FUTURE WORKS

In this paper, we have presented a simple method for choosing a non-fragile PD-controller for a class of strictly proper SISO systems with I/O delays by using geometric arguments. To prove the efficiency of the method, several illustrative examples have been considered.

Frequency	$d_{\mathcal{T}_l}$	$k_{d\infty}$	k_{p0}	$\min\{d_T, k_{d\infty}, k_{p0}\}$
$\omega_1 = 1.0388$	0.562			
$\omega_2 = 1.8131$	0.804			
$\omega_3 = 2.7121$	0.491			
$\omega_4 = 4.7540$	2.164	0.7486	0.5229	0.4910436777
$\omega_5 = 6.3137$	0.609			
$\omega_6 = 12.9703$	1.217			
$\omega_7 = 16.1440$	14.602			

TABLE III

PARAMETERS DEVIATION RESULTS WITHOUT LOSING THE STABILITY FOR SYSTEM (30).

ACKNOWLEDGEMENTS

The work of Bogdan Liacu was financially supported by CEA LIST, Laboratoire de Robotique Interactive CEA, LIST, Interactive Robotics Laboratory BP 6, 18 route du Panorama, F-92265 Fontenay-aux-Roses, France.

The authors wish to thank anonymous reviewers for their comments that helped us for improving overall quality of the paper.

REFERENCES

[1] Alfarom V.M., "PID Controllers Fragility", *ISA Transactions*, Vol.46, pp.555-559, 2007.

[2] Bajcinca, N.: *Computation of stable regions in PID parameter space for time-delay systems*, In W. Michiels(Ed.).,*Proc. of 5th IFAC workshop on time-delay systems*, Oxford:Elsevier, 2005.

[3] Chung, S.J. and Slotine, J-J.E., "Cooperative Robot Control and Concurrent Synchronization of Lagrangian Systems", *IEEE Transactions on Robotics*, VOL. 25, NO. 3, 2009.

[4] El'sgol'ts, L.E. and Norkin, S.B., *"Introduction to Theory and Applications of Differential Equations with Deviating Arguments"*, Academic Press: New York, 1973.

[5] Fernandez, A., Barreiro, A., Banos, A., Carrasco, J., "Reset control for passive teleoperation", *Proc. IECON '08*, pp. 2935-2940, 2008.

[6] Reyesa, F. and Rosadob, A., "Polynomial family of PD-type controllers for robot manipulators", Control Engineering Practice, Vol. 13, No. 4, pp. 441-450, 2005.

[7] Franklin, G. F., Emami-Naeini, A., and Powell, J. D., *"Feedback Control of Dynamic Systems. 3rd."*, Addison-Wesley Longman Publishing Co., 1993.
[8] Goodwin, G. C., Graebe, S. F., and Salgado, M. E., *"Control System Design 1st."*, Prentice Hall PTR, 2000.
[9] Gu, K., Kharitonov, V.L. and Chen, J., *"Stability of time-delay systems"*, Birkhauser: Boston, 2003.
[10] Gu, K., Niculescu, S.-I. and Chen, J., "On stability crossing curves for general systems with two delays" *J. Math. Anal. Appl.*, vol.311, pp.231-253, 2005.
[11] Hale, J.K., and Verduyn Lunel, S.M., "Introduction to Functional Differential Equations" *Applied Mathematical Sciences 99*, Spinger-Verlag, 1993.
[12] Hashimoto, T. and Ishida, Y., "An adaptive I-PD controller based on frequency domain system identification",*ISA Transactions*, Vol. 39, no. 1, pp. 71-78, 2000.
[13] Ho, M.T., "Non Fragile PID Controller Design", *Proceeding of the 39th CDC*, Sidney Australia, 2000.
[14] Keel, L.H. and Bhattacharyya, S.P., "Robust, Fragile or Optimal", *IEEE Trans. Automat. Contr.*, Vol. 42, pp.1098-1105, 1997.
[15] Keel, L.H. and Bhattacharyya, S.P., "Authors Reply", *IEEE Trans. Automat. Contr.*, Vol. 43, pp.1268, 1998.
[16] Méndez-Barrios, C.-F., Niculescu, S.-I., Morărescu, I.-C. and Gu, K., "On the Fragility of PI Controllers for Time-Delay SISO systems", *16th Mediterranean Conference on Control and Automation, MED'08*, 529-534, June 2008, Ajaccio, France.
[17] Michiels, W. and Niculescu, S.-I., "Stability and stabilization of timedelay systems. An eigenvalue-based approach", *SIAM: Philadelphia*, 2007.
[18] Morarescu, C.I., *"Qualitative analysis of distributed delay systems: Methodology and algorithms"*, Ph.D. thesis, University of Bucharest/Universite de Technologie de Compiegne, September 2006.
[19] Morarescu, C.I., Niculescu, S.I. and Gu, K., "On the Stability Crossing Curves of Some Distributed Delay Systems", *SIAM J. Appl. Dyn. System*, vol. 6, pp.475-493, 2007.
[20] Morarescu, C.I., Niculescu S.-I. and Gu, K., "On the Geometry of PI Controllers for SISO Systems with Input Delays", *Proceedings of IFAC Time Delay Systems*, Nantes, France, 2007.
[21] Neimark, J., "D-subdivisions and spaces of quasi-polynomials", *Prikl. Math. Mech.*, Vol. 13, pp.349-380, 1949.
[22] Niculescu, S. -I., *"Delay effects on stability: A robust control approach."*, Springer-Verlag: Heidelberg, LNCIS, vol.**269**, 2001.
[23] Rodriguez-Seda, E. J., Lee, D., and Spong, M.W., "Experimental Comparison Study of Control Architectures for Bilateral Teleoperators", *IEEE Transactions on Robotics*, vol. 25, pp. 1304-1318, 2009.
[24] Tafazoli. S., Salcudean, S.E., Hashtrudi-Zaad, K., and Lawrence, D., "Impedance Control of a Teleoperated Excavator", *IEEE Transactions on Control Systems Technology*, Vol. 10, NO. 3, 2002.
[25] Yongqiang Y., and Peter X. Liu, "Improving Haptic Feedback Fidelity in Wave-Variable-Based Teleoperation Orientated to Telemedical Applications", *IEEE Transactions on Intrumentation and Measurement*, Vol. 58, No. 8, August 2009.

Gateway Software Design for Mobile Patient Monitoring

Robert G. Lupu, *Member IEEE,* Vlad Cehan, Florina Ungureanu and Conrad Ciobanica

***Abstract*— In the recent years, home care services have developed very fast in spite of the increase in the costs of care-giver based services provision for patients and the ageing of caregiver population, which determine a higher request for mobile monitoring devices. Nevertheless, recent technological advances, especially wireless communications, have sustained this request; applications like Wireless Body Area Network (WBAN) have become more attractive offering flexibility and mobility to patients. For such an application, a smart device is needed to transmit the collected data from WBAN to a remote server of care centre, via a wireless network. This paper describes the software which drives the Smartphone/PDA to perform this function. A local signal process is developed to detect emergency situations and to trigger the alarm. As this solution has proved to be reliable, it represents an applicable solution in telemedicine.**

I. Introduction

ACCORDING to the United Nations 2006 Revision of World Population Prospects, by 2045 the number of elderly people in the world (those aged 60 years or over) will likely surpass for the first time in history the number of children (i.e., people under the age of 15). This situation is the consequence of long term reductions in fertility and mortality [1]. In terms of health care, this means that more people will need long-term medical assistance including not only the elderly but also those with disabilities and the chronically diseased. These will overwhelm hospitals and home nursing, leading to higher costs of medical insurance. Although cost reduction is a major goal of governments today, this must not affect the quality of care. A solution tp this problem is the automated care systems like telemedicine, telehomecare, remote patient monitoring, mobile patient monitoring and assistive technologies.

Mobile patient monitoring (MPM) of vital signs offers the potential to provide high quality care to elderly and chronically ill people in their home environment, while making effective use of healthcare resources [2]. It has the potential to lower the total costs of providing healthcare by avoiding unnecessary hospitalization, and ensuring that urgent care is afforded to people who are in need of it [3].

Patient monitoring involves acquisition and measurement of vital signs, such as ECG, blood pressure, heart rate, heart rhythm, respiration rate and oxygen saturation and data transmission to the care centre. Local data processing is needed to detect emergency situations and trigger the alarm. Patient comfort must not be affected during the monitoring process; that is why, monitoring must be wireless both in terms of local data collection and in terms of data transfer over geographical distance. A Wireless Body Area Network (WBAN) based on low cost wireless sensor network technology together with a smart phone/PDA to transmit data using wireless networks could meet the above condition of the MPM system.

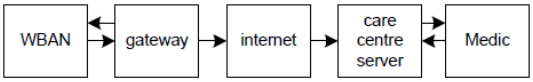


Figure 1. The mobile patient monitoring system.

The architectural block diagram of an MPM system is shown in Fig. 1. Several non invasive wireless sensors are worn by the patients in order to collect data. The acquired data are transmited to the PDA (acting as a gateway) and then forwarded to the care centre server where the data are stored, processed and analysed [4].

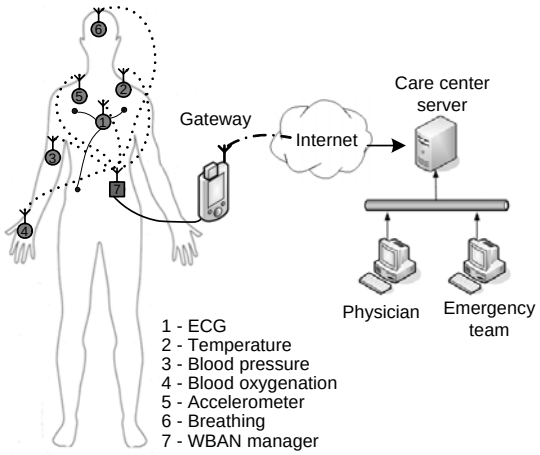


Figure 2. Mobile patient monitoring system.

A more detailed diagram of the MPM system in which the gateway software has been tested is presented in Fig. 2. The ultra low power wireless sensors in the network are all connected to a WBAN network manager, which handles low level sensors events and synchronizes the communication

Manuscript received April 30, 2010. This work was supported by Romanian National Centre of Project Management under partnership project no. 11-069/2007.

R. G. Lupu is with the "Gh. Asachi" Technical University of Iasi, Faculty of Automatic Control and Computer Engineering, 700050 Romania (fax: +402322313431 e-mail: robert@cs.tuiasi.ro).

V. Cehan is with the "Gh. Asachi" Technical University of Iasi, Faculty of Electronics, Telecommunications and Information Technology, 700506 Romania (fax: +40232217720 e-mail: vlcehan@etc.tuiasi.ro).

F. Ungureanu is with the "Gh. Asachi" Technical University of Iasi, Faculty of Automatic Control and Computer Engineering, 700050 Romania (fax: +40232231343; e-mail: fungurea@cs.tuiasi.ro).

C. Ciobănică was with the the "Gh. Asachi" Technical University of Iasi, Faculty of Automatic Control and Computer Engineering, 700050 Romania (fax: +40232231343; e-mail: conradciobanica@gmail.com).

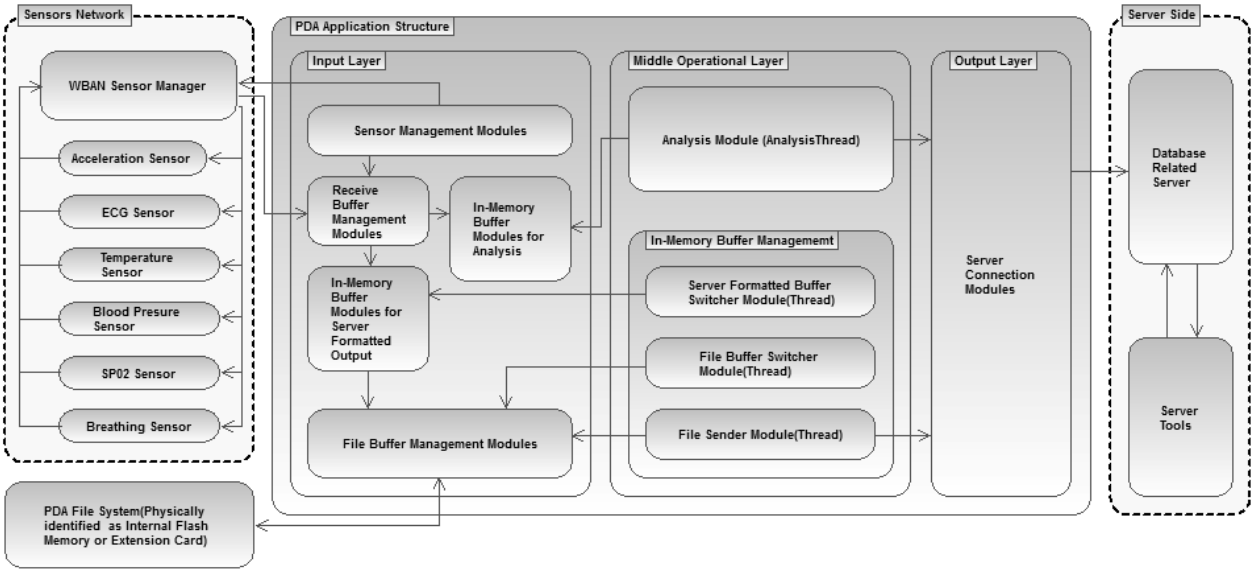


Figure 3. Functional components of the software application

inside the network. The WBAN manager is a proxy module between sensors and the gateway, physically connected via an RS232 or USB. It receives data from sensors at a specified baud rate and forwards them to the PDA. The PDA initialises the START/STOP communication inside the network.

The server has the ability to manage multiple clients, handling data and alarms. The software application is designed to provide a secure access via logging and other techniques and offers the possibility to display the acquired information for different patients at certain time periods.

The main contribution of this paper consists in designing the software for Java enabled 3G mobile phone or a PDA acting as a gateway device and also the implementation in the J2ME framework [5] and PhoneME Features KVM [6]. The developed software can perform the following monitoring functions: WBAN management, data log and transmission, data analysis and alarm notification.

The software has been integrated in the MPM system described above and tested in both laboratory and hospitals/home environments.

II. SOFTWARE ISSUE

A. System architecture

As shown in Fig. 3, the system structure is separated into three major components [7]: sensors network, PDA device (gateway) and server. The PDA application deals with the sensors network and server at the same time, acting as a transmission and analysis bridge between them. The three major layers of the PDA application have to be seen as a logical distribution for this type of application. The input layer deals with all the received data from the sensors via RS232. Each signal is identified using an ID sensor and is distributed to its own buffer group (receiver, in-memory analysis and in-memory output buffers). The received data are formatted into high level protocol in order to make them available for higher level formats (XML, JSON, etc), for more versatility. In our project, JSON [8] is the protocol used between the PDA and server because it is a simple and flexible high level protocol, designed for object serialization and RPC (Remote Procedure Call). The usage of buffers at input (and not only here) is important in order to prevent data losses and to support multiple parallel activities such as transition and writing. Also, the transmission rate and other processings are highly memory consuming while a PDA device has limited resources. That is the reason for creating groups of buffers and different execution threads for each activity. The middle layer comprises the major running threads involved in handling buffers and manages the collaboration at a higher level between the buffer modules and data acquisition. Also, the middle layer contains an analysis module, which is used in conjunction with the buffer modules designed for data analysis, because not all received data are useful in the analysis process. Another important feature in this project is the use of the file system that can be seen as an intermediary buffer between reception and transmission. This prevents a major problem that appears with the use of wireless devices. The application stores received data into the local flash memory and if, for any reason, the connection with the server is not available, this mechanism prevents data looses at server side and memory overloads on the PDA. This software separation between input and output is also useful for avoiding major synchronization between reception and transmission, the first being done in RAM, for speed, and the other on external memory, for size.

B. Sequence Diagram

In Fig. 3 the sequence diagram [9] shows the conceptual

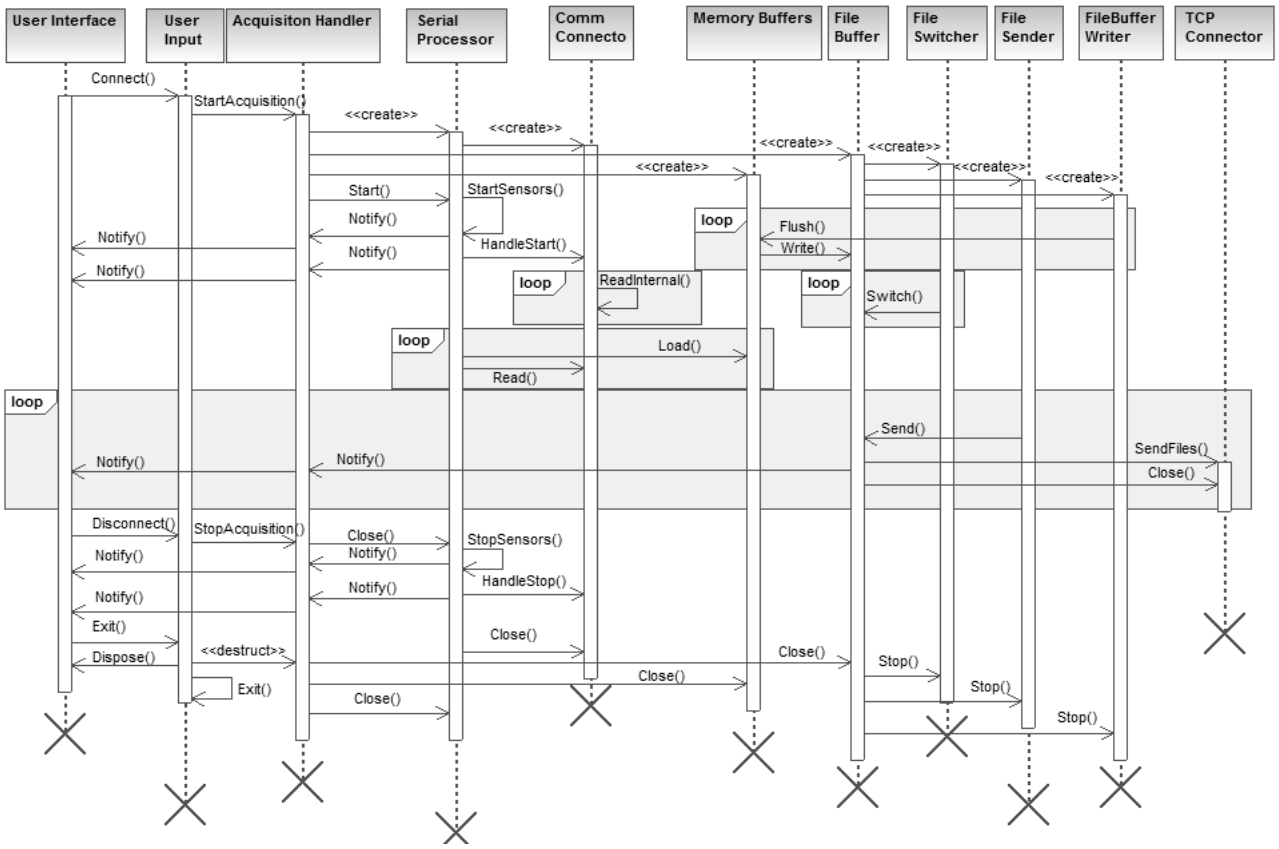


Figure 4. Software behavioral model

behaviour for gateway software. Here can be noticed the same major entities as in the previous diagram. User interaction with the program is modelled through *User Interface* and *User Input*. When a connection event occurs, *Acquisition Handler* creates the needed peers for data processing:

- *Comm. Connector* - used for receiving data;
- *Serial Processor* - which works in conjunction with the serial port interpreting the received bytes and dispatching them into appropriate memory buffers;
- *File Buffer* - used to store signals into a specific file format and then to transmit them to the care centre server.

The entities used are created in cascade, all buffers have their own thread execution, acting in pair, entity - thread controller. So, the File Buffer works with three specific threads, one for switching *File Switcher* (it changes the temporary file, creating in this manner multiple buffer files for the same reception session), one for writing memory buffers into the file *Buffer File Writer* and one for server transmission *File Sender* (this reads the accumulated files and sends them in the order they were created, one by one, validating every file transmission). If an error occurs during transmission, the file sending transaction is rejected and will be retransmitted in the next file transfer. In the diagram above, the sensor handling is represented as a simple function call and the data analysis process is incorporated in the memory buffer management in order to keep the diagram intuitive, easy to read and reusable. In real time execution, these operations are more complicated, with some specific entities which break the handling and analysis into a collaboration of classes. Another important aspect is the manner in which the connection with the server is done. Because the wireless transfer is power consuming, the connection has to be limited only for the time needed to accomplish the file transfer. This mechanism of connection on demand is also in benefit of the server, which has to handle only the active connections and not the idle ones. If data are available and a period of time has passed, the application connects to the server and sends the files. It is also better to make the transfer as fast as possible. If the connection delay is big, the time to accomplish the transfer is also long, and in the case of a PDA device it is resource consuming. The information transfer in small chunks leads to a better synchronization. To obtain a good transfer delay, a deterministic way of establishing the interval between transmissions is necessary, which is done in the manner of binary search method [10].

III. Data Analysis and Alarm Notification

The data analysis module implements analysis buffers for every sensor in which values are accumulated. Every fifteen seconds (the time period can be selected by the user), a processing thread becomes active and analyzes the data

collected from buffers.

For blood pressure, SPO2, breathing and temperature signals no other computing is performed besides reception. Values are received from sensors and compared with the upper limits (thresholds) [11] [12] [13] [14]. Except for blood pressure, the values are filtered by averaging and so any measuring error is eliminated. An alarm is triggered if one of the signals does not fit in the corresponding interval.

Fall (acceleration) and ECG signals are acquired and transmitted unfiltered and additional computation is needed before comparison thresholds are applied. The three signals received from the accelerometer sensor are processed to detect abrupt movement that could be associated with a possible patient fall. Detection is performed using a predefined threshold:

$$ax^2 + ay^2 + az^2 \geq a^2 \quad (1)$$

where *ax*, *ay* and *az* are the three axes accelerations measured by the sensor. If a threshold is reached or exceeded and is followed by a flat line of 1g, an alarm notification is sent [15].

A more complex analysis is made for ECG data to detect the QRS complex. For this we have ported in Java an open source ECG analysis software [16] available in C language based on "Quantitative investigation of QRS detection rules using the MIT/BIH arrhythmia database" [17]. Heart rate and the ST wave are calculated using the same library. Emergency situations are detected only by comparing the heart rate, QRS rate and ST amplitude with the superior thresholds [18] [19].

Generated alarms are sent to the server immediately as they appear using a channel different from that used for data transmission.

IV. In System Gateway Software Testing

All tests have been carried out at the Faculty of Medical Bioengineering, "Gr.T.Popa" University of Medicine and Pharmacy of Iasi, Romania.

A. Technical tests

First of all, a serial software monitor was developed to verify if the data sent by the sensors network correspond to the acquired signals/values. Running on a computer, it acts as an intermediate device able to display waveforms and measured values received from the network through one serial connection and forwards the same data through the second serial connection to the gateway device (Fig. 5). Additionally, it has the possibility to view the information exchanged between the network and PDA (protocol), data logging, protocol debugging and sensor network simulator by opening data log files and send the content via serial connection.

Figure. 5. System configuration for the first technical verification phase

On the other side, a simple tcp/ip server and a json parser enhanced with a graphical user interface to display signals and measured values was developed. All technical tests were made using only Wi-Fi connections.

In the first test phase, the programs mentioned above together with patient simulators and biomedical monitors were used as follows, for each sensor:

- for ecg sensor we used FLUKE PS420 as patient simulator and Schiler CARDIOVIT AT-10 monitor for ecg display. The monitor was connected to a patient simulator and the three main ecg leads were recorded. Then the monitor was replaced by an ecg sensor and the acquired wave forms were displayed with serial monitor and json file view, and compared with those previously recorded with CARDIOVIT. All the tests were validated.

- the oxygen saturation sensor was built using micro power oximeter board from Smiths Medical Company. This board comes with pc software which displays SpO2 waveform, value and heart rate. For the test we used METRON SpO2 Analyzer as signal simulator and wave forms and values from the PC software mentioned above, serial monitor and json file view were also compared and validated.

- for the acceleration sensor we used an oscilloscope for the three axes (x, y, z) in parallel with a serial monitor and json file view. Wave forms have been visually compared and validated.

- the respiration sensor was built using two thermistors for both nostrils; this measures the inspired and expired air temperature. They were used for measuring the difference between inspired and expired air temperature. The signals can be likened with sinusoidal signals with the same frequency. So, for the test we used a HAMEG hm8030 function generator and an oscilloscope. Waveforms signals from the oscilloscope and both the serial monitor and json file view were compared and validated.

- the blood pressure sensor was built using an UA-767PC blood pressure monitor and FLUKE BP Pump 2 non invasive blood pressure simulator. The values from the simulator and monitor were compared with those displayed by the serial monitor and json file view and validated.

- the values from the temperature sensor displayed on the serial monitor and json file view were compared with a medical thermometer and validated.

In the second phase of testing, all sensors were started together in the same conditions as before, but this time the simple tcp/ip server and json file view were replaced by a medical care server and the data visualized with RomSoft MedApp application (Fig. 6). The test lasted for twenty minutes. All results were recorded locally (using serial monitor) and were transmitted to the care server to confirm the quality of the signals. Due to the laboratory tests

conditions, no errors occurred during data transmission.

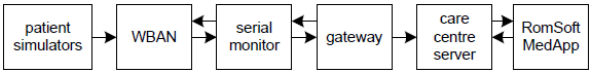


Figure 6. System configuration for the second technical verification phase

The alarm notification was the third phase in the gateway software test. The software has an alarm setting window which enables physicians to set up the alarm thresholds according to the physiological status of the patient. The predefined thresholds are listed in table x. The alarms were tested, using patient simulators, as follows:

- for the ecg sensor, we used FLUKE PS420 as patient simulator. ST amplitude was tested selecting the levels of ST elevation or ST depression from the Simulating Functions → ST Elevation and Depression Waves menu. QRS detection and heart rate alarm thresholds were tested selecting different values from the Stimulating Functions → ECG Rate menu. Notification alarms were received by the server each time when the ST amplitude and heart beats per minute didn't fit in the accepted values (see table I).

- the oxygen saturation threshold alarm was tested using METRON SpO2 Analyzer as signal simulator. It allowed the increase/decrease of the SpO2 values and the changes below the accepted limit were followed by an alarm notification received by the server.

- for the acceleration sensor, the threshold alarm was tested shaking or dropping it and observing whether the alarm notification was received by the server.

- the respiration alarm was tested by switching off the sinusoidal function generator for more then the set threshold. This caused an alarm notification to be received by the server.

- the blood pressure notification alarm was tested using FLUKE BP Pump 2 non invasive blood pressure simulator, from which different standard blood pressure simulations were selected. Depending on the blood pressure values, a corresponding alarm notification was received or not by the server.

- a temperature notification alarm was tested heating or cooling the sensor above or below threshold limits and observing if an alarm was received by the server.

The last technical test of the gateway software was the endurance and power consumption of PDA test. For this, we used two different PDAs, one Fujitsu Siemens N520 and one HTC-7500. We recorded all signals together for 40 minutes with a serial monitor, then the log file content was appended several times to cover nine hours. Then the serial monitors were used as sensor network simulators by opening the log file and sending the content via serial connection to the PDA (Fig. 7). Both PDAs with batteries fully loaded were tested, one at a time. The FS-N520 lasted for about two hours and HTC-7500 for about six hours and no software failure was recorded.

Figure 7. System configuration for endurance test

B. *Subject tests*

Test on human subjects were carried out only on students from the Faculty of Medical Bioengineering. They wore the sensor network and the PDA device on their bodies. The test consisted in moving inside a room first and then inside an area (a floor). Only the Wi-Fi connection was used to see if the data were corrupted when the subject exits/enters the wireless network coverage. We encountered one problem: if the wireless connection dropped during file transmission the same file was retransmitted when the connection became available. At the server end some data were duplicated and data verification was required. Outside the building, tests were carried out using only a GPRS connection.

C. *Clinical test and survey:*

As tests are still under way, we do not have final results yet.

TABLE I
PREDEFINED ALARM THRESHOLDS

Signal	Low	High	Average
Acceleration	-	4 g	No
BP diastolic hypotension	-	60 mmHg	No
BP sistolic hypotension	-	90 mmHg	No
BP diastolic prehypertension	80 mmHg	89 mmHg	No
BP systolic prehypertension	120 mmHg	139 mmHg	No
BP diastolic hypertension level one	90 mmHg	99 mmHg	No
BP systolic hypertension level one	140 mmHg	159 mmHg	No
BP diastolic hypertension level two	100 mmHg	-	No
BP systolic hypertension level two	160 mmHg	-	No
Ecg - ST	-150 uV	200 uV	Yes
Egc - QRS	12 bpm	140 bpm	Yes
Ecg - HR	45 bpm	140 bpm	Yes
SPO2	90 %	-	Yes
Breathing rate	0 for more then 20sec	-	Yes
Temperature	36.1 ºC	37.8 ºC	Yes

BP – blood pressure;
HR – heart rate;
ST – ST wave;
SPO2 – blood oxygenation;
QRS – QRS complex.

V. CONCLUSION

In this paper, a platform independent gateway software design used in mobile patient monitoring is presented. The data received from the WBAN are analyzed, locally processed if necessary, and

forwarded to the care centre server. The test carried out reveals that this software is reliable for data transmission, user-friendly, convenient and feasible for mobile patient monitoring. For future development, standard communication protocols between the gateway-WBAN and gateway - server and a gateway software framework will determine better monitoring devices and systems at a lower price.

ACKNOWLEDGMENT

This work was supported by the Romanian National Centre of Project Management under Partnership Project no. 11069/2007 "Telemon".

REFERENCES

[1] The website of United Nations: http://www.un.org/esa/population/publications/wpp2006/FS_ageing.pdf (accessed April 2010)
[2] T. Bratan, M. Clarke, "Towards the Design of a Generic systems Architecture for Remote Patient Monitoring", *27th Annual Conference of IEEE Engineering in Medicine and Biology Society, 2005, Conference Proceedings,* Shanghai, China (2005)
[3] A. Raman, "Enforcing privacy through security in remote patient monitoring ecosystems", *6th International Special Topic Conference on ITAB,* Tokyo, Japan (2007)
[4] A. Saeed, M. Faezipour, "A scalable wireless body area network for bio-telemetry", *Journal of information processing systems,* vol. 4, no. 2, June 2009
[5] Java 2 Micro Edition Reference: http://java.sun.com/javame/reference/apis.jsp (accessed April 2010)
[6] PhoneMe Project: https://phoneme.dev.java.net/ (accessed April 2010)
[7] C. Kambalyal, "3-Tier Architecture", http://channukambalyal.tripod.com/NTierArchitecture.pdf (accessed April 2010)
[8] JSON Format Reference: http://www.json.org/ (accessed April 2010)
[9] *Unified Modeling Language: Superstructure*, version 2.0, pag. 488 (2005). Available: http://www.omg.org/spec/UML/2.0/
[10] *Algorithms – Design Techniques and Analysis*, M. H. Alsuwaiyel, World Scientific Publishing Co. Pte. Ltd., pag 8-12(1999). ISBN 981-02-3740-5
[11] The website of American Heart Association: http://www.americanheart.org/presenter.jhtml?identifier=4450 (accessed April 2010)
[12] J. M. Haynes, "The Ear as an Alternative Site for a Pulse Oximeter Finger Clip Sensor", *Respiratory Care*, vol. 52. no. 6, June, 2007
[13] Simmers, Louise. "Diversified Health Occupations". 2nd ed. Canada: Delmar, 1988: 150-151
[14] U.S. Department of health and human services, National Institutes of Health, National Heart, Lung, and Blood Institute, "Your guide to healthy sleep"
[15] F. Sposaro, G. Tyson, „iFal: An android application for fall monitoring and response", *31st Annual International Conference of the IEEE EMBS,* Minneapolis, Minnesota, USA, September 2-6, 2009
[16] P.S. Hamilton, "Open source ECG analysis software", 2002, http://www.eplimited.com (accessed April 2010)
[17] Hamilton, Tompkins, W. J., "Quantitative investigation of QRS detection rules using the MIT/BIH arrhythmia database", IEEE Trans. Biomed. Eng., BME-33, pp. 1158-1165, 1987
[18] P. Kligfield, "Principles of simple heart rate adjustment of ST segment depression during exercise electrocardiography", *Cardiology Journal*, vol. 15, no. 2, pp.194-200, 2008. ISSN 1897-5593
[19] American heart association, "Target Heart Rates", http://www.americanheart.org/presenter.jhtml?identifier=4736 (accessed April 2010)

Multi Model Structure Reduction using Nonlinearity Compensators

Ciprian Lupu, Popescu Dumitru, Andreea Udrea, and Catalin Petrescu

***Abstract*— The multi model structures are viable control solutions for the classes of systems with important nonlinearities or different functioning regimes. One of these structures' specific problems is the determination of the models number: an increased number leads to superior performances but very complex structure. The paper proposes an original methodology for reducing the number of models without decreasing the performances.**

This solution is of practical importance allowing facile implementation on PLC and process computers. The experimental results prove the structure's performances.

I. INTRODUCTION

THE multi model systems represent a relative new approach for the nonlinear systems control. Since the 90's, different studies on multi model control strategies have been developed. The Balakrishnan's and Narenda's first papers propose several stabile and robust methods using classical switching and tuning algorithms [1].

Further research in this field determined the extension and improvement of the multi model control concept. Magill and Lainiotis introduce the model representation through Kalman filters. In order to maintain the stability of the minimum phase systems, Middelton improves the switching procedure using an algorithm with hysteresis. Landau and Karimi have important contributions regarding several particular multiple model adaptive structures [2]. Dubois, Dieulot and Borne apply fuzzy procedures for switching and use sliding mode control.

This paper proposes a multi model control structure which contains, for each model/controller pair, a nonlinearity compensator. It is based on the determination of each model's static characteristic. This solution reduces the number of models and a decreases the overall complexity of the global structure.

This structure can be applied in the case of processes with important nonlinearities.

Manuscript received April 30, 2010. This work was supported by IDEI Research Program of Romanian Research, Development and Integration National Plan II, Grant no. 1044/2007.

C. Lupu is with the Department of Automatics and Computers, University "Politehnica" of Bucharest, 313 Splaiul Independentei, Sector 6, 060042-Bucharest, Romania (phone: +40-021-402-9167; fax: +40-021-402-9167; e-mail: cip@ indinf.pub.ro).

D. Popescu, A. Udrea and C. Petrescu are with the Department of Automatics and Computers, University "Politehnica" of Bucharest, 313 Splaiul Independentei, Sector 6, 060042-Bucharest, Romania (e-mail: dpopescu;andreea;catalin@ indinf.pub.ro).

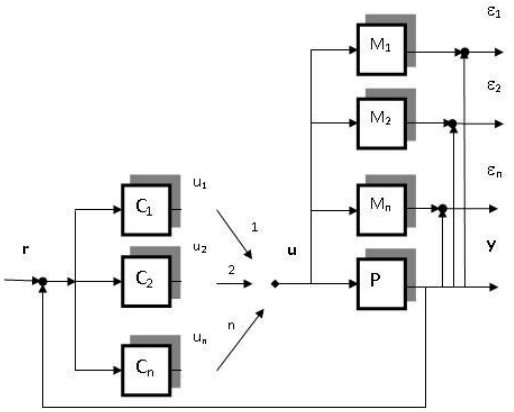


Fig. 1. Multi model control structure.

II. CLASSIC MULTI MODEL APPROACS

The classic control solution implies choosing a set of models *M*, and on a set of the correspondent controllers *C*:

$$M = \{\mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3 \ldots \mathcal{M}_n\}, \; C = \{C_1, C_2, C_3 \ldots C_n\}$$

Based on these model/controller pairs the closed-loop configuration is the one presented in Fig. 1.

The input and output of the process *P* are *u* and *y* respectively, and *r* is the set point of the system. The M_i (*i=1, 2, ... n*) models are determined a priori. For each model M_i a controller C_i is designed in order to assure the nominal performances for the pair (M_i, C_i).

The main idea of the multi model structure construction is based on dividing the process functioning region in *n* small disjoint and adjacent zones, for which the models are simpler and the *n* corresponding control algorithms have low complexity (Figure 2).

One of the principles used in zones' choosing is that the absolute value of the difference between the static characteristic and its linearization has to be smaller than the imposed threshold. This does not impose using a linear model for the region. It is very possible to have a second or third or *m* order model and a complex corresponding control algorithm. A very complex algorithm can determine better performances but uses important hardware resources on real time implementation.

In real situations there must be a balance between complex control algorithms and complex real time hardware/software architectures.

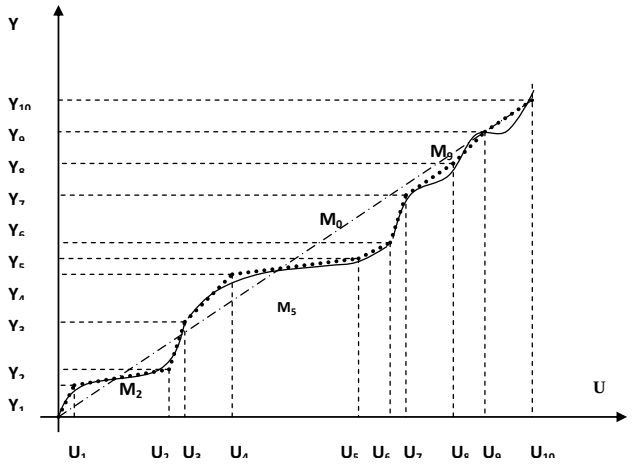


Fig. 2. Construction of the set of process's models

In Figure 2 the continuous line represents the process static characteristic, the dotted line the linear models for a large number of zones and dashed and dotted line is global linear model.

The difference between the global model and the process characteristic is large (maximal distance in *U6, Y6* point).

Using a single controller provides poor performances. For high performances and robust implementation a more complex control strategy must be used.

III. PROPOSED SOLUTION FOR MULTI MODEL STRUCTURE

This paper proposes a multi model control structure which, for each controller, provides a nonlinearity compensator [3]. This solution allows a reduced number of models and a reduced complexity for each control algorithm. This solution is named "control system with inverse model" [10].

In literature proposes a lot of inverse model structures. For the presented control solution a very simple and efficient structure, presented in Figure 3, is employed. This solution sums two commands: the first one "a direct command" generated by the feed forward command generator, and the second generated by a classic and very simple algorithm (PID, RST etc.).

This structure is added to all the model/controller pairs of the multi model structure. For each controller, the first command, based on the process static characteristics, is dependent on set point value and is designed to generate a corresponding value to drive the process's output close to imposed set point value. The second (classic feedback) algorithm generates a command that corrects the difference caused by external disturbances and, accordingly to the set point, by eventual bias error caused by mismatches between calculated inverse process characteristic and the situation from real process.

The presented solution proposes treating these "inverse model" mismatches that "disturb" the first command as a

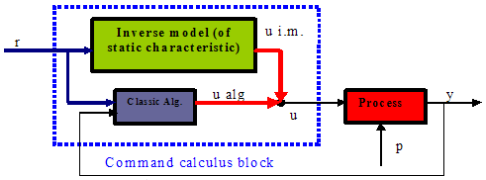


Fig. 3. Proposed scheme for inverse model structure

second command classic algorithm's model mismatches. This solution imposes designed a classic algorithm with robustness reserves. For this reason, designing the second algorithm has in two steps:

- design of a classic algorithm based on a model identified in a functioning point – selected fortuitously or, on the middle of the corresponding segment process characteristic;
- verification of algorithm's robustness and improving, if necessary - (re)designing procedure;

In Figure 3, the blocks and variables are as follows: Process – physical system to be controlled; Command calculus – unit that computes the process control law; Classic Alg. – control algorithm (PID, RST); y – output of the process; u – output of the Command calculus block; u alg. – output of the classic algorithm; u i.m. – output of the inverse model block; r – system's set point or reference trajectory; p – disturbances of physical process.

This solution used in the context of a multi model structure has three important aspects:

- Selection of a reduced number of zones where the nonlinearity is important but lower than an imposed threshold.
- Construction of the compensator block for each zone.
- Designing the correspondent controller for each zone.

All three will be presented in next sections.

A. *Zones selection*

The number of zones must be reduced (*2, 3* or maximum *4*) and these can consist in the medium or "local" tendencies of the nonlinear characteristic [4], [5]. Figure 4 presents an example for this aspect.

It can be imposed that the difference between the tendency and the real characteristic must be less or equal to an imposed margin.

In Figure 4 the continuous line represents the process static characteristic, the dotted line – the linear models and dashed and dotted line the global linear model.

B. *Construction of nonlinear compensator blocks*

This operation is based on several experiments. The command *u(k)* is increasing and decreasing and the corresponding stabilized process output *y(k)* is measured. The command *u(k)* covers all possible values (*0 to 100%* in percentage representation). Because the process is disturbed by noises, the measurements of the static characteristics are

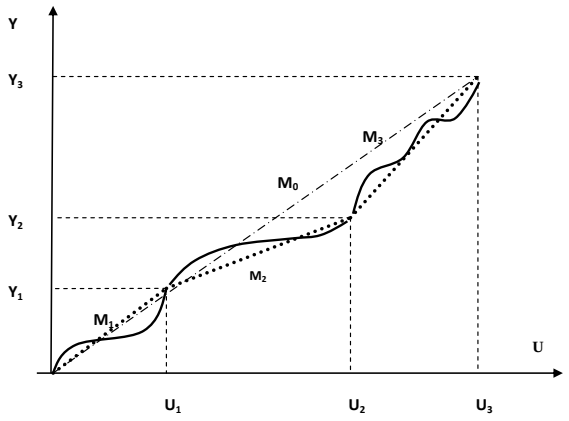


Fig. 4. Selection of major zones

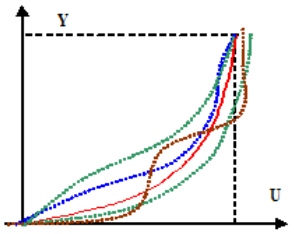


Fig. 5. Determination of static characteristic of the process. Red (continuous) line represents the final characteristic

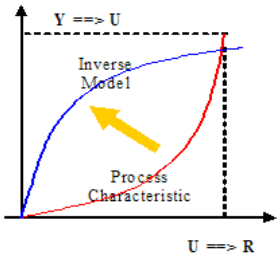


Fig. 6. Construction of inverse model

not identically. The final static characteristic is obtained by meaning these experiments. Figure 5 presents this operation. The graphic between two "mean" points can be obtained using an extrapolation procedure.

According to system identification theory [4] the dispersion of process trajectory can be found using expression (1):

$$\sigma^2[n] \cong \frac{1}{n-1}\sum_{i=1}^{n} y^2[i],\ \forall n \in N^* \setminus \{1\} \tag{1}$$

This can express a measure of noise, process's nonlinearity etc. and is important for robustness perspective during the design of the control algorithm [7], [8].

The next step in obtaining the nonlinear compensator block deals with inverting the process's static characteristic. Figure 6 presents this construction. According to this, *u(k)* is dependent to *r(k)*. This characteristic is stored in a table; thus, for the inverse model based controller, selecting a new set point *r(k)* means searching in this table the corresponding command *u(k)* that determines a process output *y(k)* close to the reference value.

C. Designing of controllers

The zones control algorithm's duty is to eliminate the disturbances and differences between inverse models' computed command and real process behavior. A large

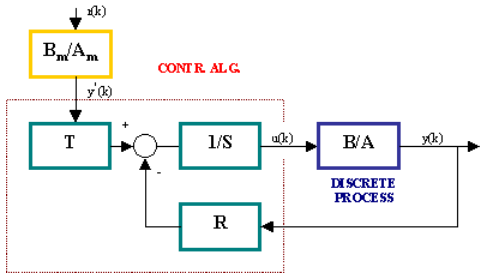


Fig. 7. RST control algorithm structure

variety of control algorithms can be used here, PID, RST, fuzzy etc., but the goal is to have a very simplified one.

For this study we use a RST algorithm. This is designed using a pole placement procedure [2]. Figure 7 presents the RST algorithm:

Where R, S, T polynoms are:

$$\begin{aligned} R(q^{-1}) &= r_0 + r_1 q^{-1} + \ldots + r_{nr} q^{-nr} \\ S(q^{-1}) &= s_0 + s_1 q^{-1} + \ldots + s_{ns} q^{-ns} \\ T(q^{-1}) &= t_0 + t_1 q^{-1} + \ldots + t_{nt} q^{-nt} \end{aligned} \tag{2}$$

Pole placement algorithm makes use of the identified model.

$$y(k) = \frac{q^{-d} B(q^{-1})}{A(q^{-1})} u(k) \tag{3}$$

where

$$\begin{aligned} B(q^{-1}) &= b_1 q^{-1} + b_2 q^{-2} + \ldots + b_{nb} q^{-nb} \\ A(q^{-1}) &= 1 + a_1 q^{-1} + \ldots + a_{na} q^{-na} \end{aligned} \tag{4}$$

The identification is made in a specific process operating point and can use recursive least square algorithm developed in [2].

This approach allows the users to verify, and if is necessary, to calibrate the algorithm's robustness. The following expression and Figure 8 present the "disturbance-output" sensitivity function.

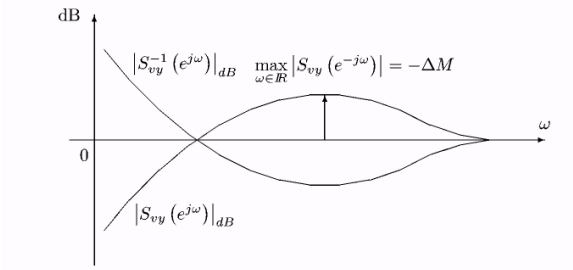


Fig. 8. Sensitivity function graphic representation

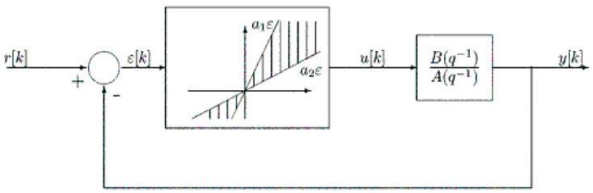


Fig. 9. Robust control design procedure

$$S_{vy}(e^{j\omega}) \overset{def}{=} H_{vy}(e^{j\omega}) = \\ = \frac{A(e^{j\omega})S(e^{j\omega})}{A(e^{j\omega})S(e^{j\omega}) + B(e^{j\omega})R(e^{j\omega})}, \quad \forall \omega \in R \tag{5}$$

The negative maximum value of sensitivity function represents the module margin.

$$\Delta M\big|_{dB} = -\max_{\omega \in R} \left| S_{vy}(e^{j\omega}) \right|_{dB} \tag{6}$$

Based on this value [2], in a "input-output" representation, process nonlinearity can be bounded inside of "conic" sector, presented in Figure 9, where a_1 and a_2 are calculated using next expression:

$$\frac{1}{1-\Delta M} \geq a_1 \geq a_2 \geq \frac{1}{1+\Delta M} \tag{7}$$

Finally, if it is imposed that all nonlinear characteristics should be (graphically) bounded by the two gains, or if the gain limit should be greater or equal to process static characteristic maximal distance $\Delta G \geq md$, then a controller that has sufficient robustness was designed.

D. *Multi model global architectures*

Partitioning the nonlinear characteristic like in Figure 4 and combining the multi model structure (presented in Figure 1) with the control structure (presented in Figure 4 determines) the global architecture of multi model control system presented in Figure 10.

On Figure 10, the blocks and variables are as follows: Process – physical system to be controlled: Command calculus – unit that computes the process control law; Alg. *i* – i control algorithms (PID, RST); *y* – output of the process; *u* – output of the Command calculus block; *u i.* – output of the *i* control algorithm; *r* – system's set point or reference trajectory; *p* – disturbance of physical process.

IV. Advantages and Disadvantages of the Proposed Structure

A. *Advantages of proposed structure*

The main advantage consists in using a simplified and

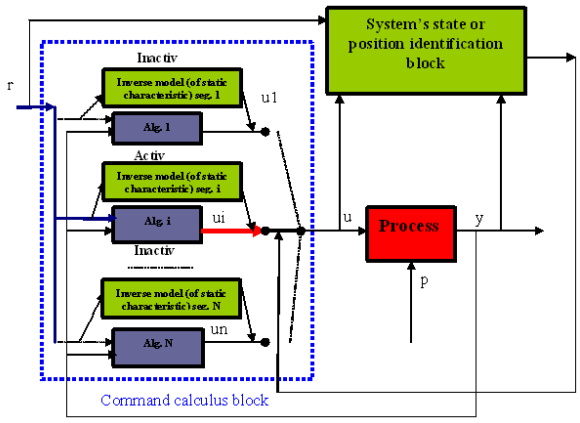


Fig. 10. RST control algorithm structure

performance operating control structure. Designing procedure is based on classic pole placement and determination of inverse command blocks. Well known procedures are used for dynamic and static models' identification.

Because the global command contains a "constant" component generated by an inverse model command block, the system has good stability margin.

The inverse model command generator can be replaced by a fuzzy logic bloc or neural network that can "contain" human experience about some nonlinear processes.

Due to the fact that the control laws are not very complex, real time software and hardware implementation doesn't need important resources.

B. *Disadvantages or limitations of the structure*

The main limitation is that this procedure can be applied only for the processes that permit the construction of the static characteristic.

This structure is very difficult to use for systems with non bijective characteristic and for systems with different functioning regimes.

Another limitation is that this structure can be used only for stable processes. In situations where the process is "running", the direct (feed forward) command is very possible to not have enough flexibility to control it.

The increased number of experiments for determination of the mean static characteristic can be another disadvantage of the structure.

C. *Possible developing*

For special situations, the direct command generators (feed forward) included in multi model structure can be constructed as a single general block. This block compensates the process nonlinearity and allows using simplified control laws in multiple controller structure.

These systems can be easily implemented on PLC structures particularly, and real time control systems, generally.

V. EXPERIMENTAL RESULTS

We evaluate the achieved performances of the proposed control structure using an experimental installation presented in Figure 11, where the position of an object contained in the vertical tube must be controlled using an air flow generator.

The nonlinear process static characteristic is presented in Figure 12.

For this, there are selected 12 zones (Figure 12) for a classic multi model structure (Figure 1). The models and corresponding area (output %) are: M1: 0-35%, M2: 35-50%, M3: 50-54%, M4: 54-60%, M5: 60-69%, M6: 69-72%, M7: 72-75%, M8: 75-78%, M9: 78-84%, M10: 84-86%, M11: 86-95%, M12: 95-100%. All 12 are first order models. For example, for M1 using Te=0.2 s sampling time and Least Square identification method from Adaptech/WinPIM the model is:

$$M_1 = \frac{0.487180}{1 - 0.79091 q^{-1}}$$

The corresponding controller (for Tracking performances: second order dynamic system with w0=2.0, x=0.95, Disturbance rejection performances: second order dynamic system with w0=1.1, x=0.8, using WinReg) is:

$$R(q^{-1}) = 0.263281 - 0.179872\ q^{-1}$$
$$S(q^{-1}) = 1.000000 - 1.000000\ q^{-1}$$
$$T(q^{-1}) = 2.052629 - 3.412794\,q^{-1} + 1.443573\ q^{-2}$$

The RST control algorithm can be written as follows:

Fig. 11. Experimental installation

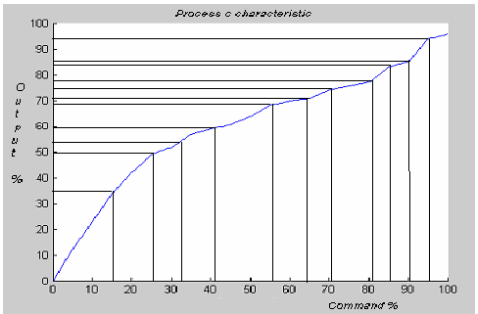


Fig. 12. Nonlinear process characteristic

$$u(k) = \frac{1}{s_0}[-\sum_{i=1}^{n_S} s_i u(k-i) - \sum_{i=0}^{n_R} r_i y(k-i) + \sum_{i=0}^{n_T} t_i y^*(k-i)]$$

where *R*, *S*, *T* polynomials are presented in relation (2) and n_S, n_R, n_T express the corresponding degrees and also the memory dimension for the software implementation of the algorithm. For example, if $n_R=2$, then three memory locations must be reserved for the process's output: *y(k), y(k-1), y(k-2)*. Respectively, the same rule applies for *u(k)* and $y^*(k)$.

To calculate the corresponding command the controller presented before, there are used 7 multiplication and 7 addition or subtraction operations.

Because the multi models control structure must assure no bump commutations, all of the 12 control algorithms must work in parallel [6]. This condition gives the total number of operations: 12 x 7 = 84 multiplications and 12 x 7 = 84 additions/ subtractions.

For the proposed control structure, with nonlinear blocks there are selected 3 zones Z1: 0-50%, Z2: 50-80% and Z3: 80-100%, presented in Figure 13.

The models are:

$$M_1 = \frac{0.0964 - 0.19647 q^{-1}}{1 - 1.06891 q^{-1} + 0.22991 q^{-2}}$$

$$M_2 = \frac{0.01297 + 0.05397 q^{-1} + 0.03674 q^{-2}}{1 - 0.76251 q^{-1}}$$

$$M_3 = \frac{0.02187 + 0.05668 q^{-1} + 0.06048 q^{-2}}{1 - 0.93161 q^{-1} + 0.02741 q^{-2} + 0.09863 q^{-3}}$$

In this case, we have computed three corresponding RST algorithms using a pole placement procedure from Adaptech/WinREG platform. The same nominal performances are imposed to all systems, through a second order system, defined by the dynamics $\omega_0 = 1.25$, $\xi = 1.2$ (tracking performances) and $\omega_0 = 2$, $\xi = 0.8$ (disturbance rejection performances) respectively, keeping the same sampling period as for identification.

$$R_1(q^{-1}) = 1.863259 - 2.027113 q^{-1} + 0.520743 q^{-2}$$
$$S_1(q^{-1}) = 1.000000 - 0.554998 q^{-1} + 0.445002 q^{-2}$$
$$T_1(q^{-1}) = 3.414484 - 4.931505 q^{-1} + 1.873910 q^{-2}$$

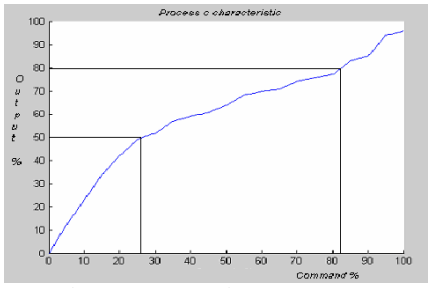


Fig. 13. Selection of the three zones of nonlinear characteristic

$$R_2(q^{-1}) = 2.309206 - 1.624937q^{-1}$$
$$S_2(q^{-1}) = 1.0 - 0.815278q^{-1} - 0.106427q^{-2} - 0.078295q^{-3}$$
$$T_2(q^{-1}) = 9.645062 - 14.928993q^{-1} + 5.968200\,q^{-2}$$

$$R_3(q^{-1}) = 1.72482 - 1.611292q^{-1} - 0.03784q^{-2} + 0.292903q^{-3}$$
$$S_3(q^{-1}) = 1.0 - 0.725187q^{-1} - 0.095205q^{-2} - 0.179608q^{-3}$$
$$T_3(q^{-1}) = 7.192692 - 11.645508q^{-1} + 4.821405\,q^{-2}$$

To calculate the corresponding command for the *C1* controller there are used: 9 multiplications and 9 additions or subtractions, for *C2*: 9 multiplications and 9 additions or subtractions and for C3 11 multiplications and 11 additions or subtractions, giving a total number of 29 multiplications and 29 additions or subtractions.

For the proposed control structure, in addition to the command calculus operation, here is the calculus for the direct command. This depends on the software implementation. For PLC, particular and real time process computers, where (C) code programming can be used, the implementation is:

```
// segment determination
segment = (int)(floor(rdk/10));
// segment gain and difference determination
panta = (tab_cp[segment+1] - tab_cp[segment]) * 0.1;
// linear value calculus
val_com_tr = uk + 1.00 * (panta * (rdk - segment*10.0) +
tab_cp[segment]);
```

One needs 10 multiplications and 4 additions or subtractions. The total operations number for the proposed structure is: 59 multiplications and 41 additions or subtractions.

It is obvious that the proposed structure needs a diminished number of multiplications if compared to the classic multi model solutions and a comparative value for the number of additions and subtractions. This means that the system with nonlinear compensators is faster or needs simplified hardware and software architecture.

VI. Conclusions

In this paper there is proposed a multi model control structure which contains, for each model/controller, a nonlinearity compensator. This solution allows a reduced number of models and a reduced complexity for global structure. The analysis on the advantages and disadvantages of proposed structure is made.

The experimental results done on laboratory installation present a case where the proposed structure is a faster solution then the classic multi model structure.

This structure can be easily implemented on PLC and real time process computer.

Acknowledgment

This work was supported by IDEI Research Program of Romanian Research, Development and Integration National Plan II, Grant no. 1044/2007.

References

[1] K. S. Narendra and J. Balakrishnan, "Adaptive Control using multiple models", *IEEE Transactions on Automatic Control*, vol. 42, no. 2, Feb., 1997, pp. 171–187

[2] I. D. Landau., R. Lozano and M. M'Saad, *Adaptive Control*, Springer Verlag, London, ISBN 3-540-76187-X, 1997.

[3] L. Ljung, T. Soderstroom, *Theory and Practice of Recursive Identification*, MIT Press, Cambridge, Massashusetts, 1983

[4] D. Stefanoiu, J. Culita and P. Stoica, - *Fundamentele Modelarii si Identificarii Sistemelor*, Editura Printech, 2005, ISBN : 973-718-368-1.

[5] G. Tao and P. Kokotovic, *Adaptive control of systems with actuator and sensor nonlinearities*, Wiley, N.Y. 1996

[6] Lupu, D. Popescu, B. Ciubotaru, C. Petrescu, G. Florea ,- *Switching Solution for Multiple Models Control Systems*, MED06, paper WLA2-2, 28-30 June, Ancona, Italy. 2006.

[7] I. Dumitrache, *Ingineria Reglarii Automate*, Politehnica Press, Bucuresti, 2005, ISBN 973-8449-72-3.

[8] J.M. Flaus, *La Regulation Industrielle*, Editure Herms, Paris, 1994, ISBN 2-86601-441-3.

[9] G. Pajunen, "Adaptive control of wiener type nonlinear system", *Automatica*, no. 28, 1992, pp. 781-785

[10] J. Richalet, *Practique de la Commande Predictiv*, Editura Herms, Paris, 1993, ISBN 978-2212115536.

A Security Infrastructure for the University-based IT Services

Marius MARIAN[1], *Department of Automation, University of Craiova*

***Abstract*—this paper deals with the design of a secure IT infrastructure on a University-wide basis. The security solution is based on public-key cryptography and the related technologies. Digital certificates The system will provide security at two main levels, for applications via SSL/TLS channels or by means of S/MIME messages and also for networking via IPsec protocol suite.**

***Keywords*: PKI, X.509v3 digital certificates, security infrastructure.**

I. INTRODUCTION

IN any university environment a major part of the information exchange is taking place in an electronic form. Document flows from and towards the administrative services and also from the educational process are also eventually processed or stored in or on digital form/support. Members of the university are using all other IT services in their daily activities. As everyone knows there are security risks associated to the use of IT applications and services. We will provide just some few examples to support the precedent statement. Communication without confidentiality is dangerous because third parties can easily read, modify, block, or even replace entire messages. One can easily imagine that certain information – for example, salary information – within a university environment must be transmitted/stored confidentially. Another important topic to be considered is the integrity of information exchanged. Similarly, access control must also be achieved in a networked environment. Students and professors have different requirements and constraints, therefore they should posses different privileges on a university IT infrastructure. They must be authenticated first and then authorized based on their credentials to access particular IT resources (i.e. electronic documents, services, infrastructure, etc.). Without authentication the traceability (logging) of and responsibility for events related to a particular user movements and actions would be difficult to implement. We can see thus that these basic security features are all necessary for a healthy, prudent and judicious IT environment.

The purpose of this paper is to present a plan of a security infrastructure for our university IT environment dealing with the issues presented above and mainly with user authentication and authorization. It will not go further into topics such as detailed analysis of security risks involved and risk management policies. No central security policy has been defined and consequently enforced at university IT infrastructure level, so far. This obviously has profound implications for the entire IT infrastructure but the subject will not be investigated here.

[1] Marius MARIAN is with the Department of Automation, Faculty of Automation, Computers and Electronics of University of Craiova, Romania (e-mail: marius.marian@cs.ucv.ro).

II. KEY CONCEPTS

A. Security services

The most important security service to be achieved within the IT infrastructure of an educational institution is identification and authentication of users. The system needs the means to verify the entity requesting access. So far this step is achieved by means of usernames and passwords. This approach is not as safe as it could be since most users ignore the risks associated with choosing a weak password. Moreover, identifiers are not chosen in a coherent and unitary manner at university level and consequently within the same IT environment a user may posses many different IDs. Since the process of authentication implies only identification and verification of the identity of a particular user, a secure system must also be able to allow or refrain the user actions inside the system after having her authenticated. This process is called authorization.

The concept of single sign-on (SSO) is important and highly desirable within a university from our perspective. This concept implies that a user authenticates only once to the system and then she is transparently granted access to a well-specified set of resources inside the IT infrastructure. From an administrative perspective, the advantage of SSO is that it prevents users from choosing bad passwords (and hopefully, usernames).

Data confidentiality is also essential to the system. Confidentiality services restrict access to the content of sensitive data to only those individuals who are authorized to view the data [1]. Confidentiality measures prevent the unauthorized disclosure of information to unauthorized individuals or processes. In the academic context it worths mentioning that failing to respect and protect the confidentiality of students, teaching staff and their families not only erodes public trust in the professionalism and ethics of the institution and its employees, but it also puts the involved employees at grave risk of both disciplinary and legal action. Therefore peoples' privacy is a matter of utmost concern for the university IT infrastructure.

Data integrity services address the unauthorized or accidental modification of data. To ensure data integrity, a system must be able to detect unauthorized data modification. The goal is for the end-user of the data to

verify that the data has not been altered during transmission or archival. One-way hashing and digital signatures are a common way to ensure both data integrity and data origin.

B. *Public-key infrastructure*

To deploy security within closed user groups, symmetric key cryptography could be a solution, but when such groups grow up considerably or we have to deal with open groups of users public-key (asymmetric) cryptography (PKC) is by far the most used technique. It is a verified fact that in the last decade, asymmetric cryptography has become the fundamental technology for building security systems in Internet: each participant to PKC has a pair of keys (consisting of a private key and a public key). The most important characteristic of PKC is that computing one key from another is theoretically impossible.

PKC is very well suited for performing digital signatures, secure transmission (exchange) of secret keys, and data confidentiality. Well-known examples of PKC standards include DSS (NIST Digital Signature Standard) [2], RSA (based on Rivest, Shamir, Adleman cryptographic algorithm) [3], PGP (Philip Zimmermann's Pretty Good Privacy system) [4], Diffie-Hellman key exchange [5], etc.

In practice, PKC is established by means of a public-key infrastructure which is an ensemble of hardware, software, people, policies and procedures needed to create, manage, distribute, use, store and revoke digital certificates [1]. The reason for establishing PKIs is born from the main disadvantage of PKC: public keys must be introduced in a trustworthy manner to all users and this can only be accomplished through the trusted third party (TTP) model. Within this model, a third party – commonly trusted by all interested parties – authenticates user identities and the corresponding public keys to each other. Here it is worthed a short introduction to X.509 standard [6]. This ITU-T[2] standard introduces public-key infrastructures for single sign-on and privilege management infrastructure and also defines a framework for data origin authentication and peer entity authentication services, including formats for public-key (digital) certificates, attribute certificates and certificate revocation lists (CRL). X.509 version 3 (X.509v3) certificates [8], [7] are cryptographic data structures used for binding together a user identity, her public key, and a digital signature performed by a TTP. The key role of trust in X.509v3 is to describe the relationship between an entity and a certification authority (CA); an entity shall be certain that it can trust a CA to issue only valid and reliable public-key certificates. Prior to creating an actual digital certificate, a CA will properly identify the entity to which a certificate is issued. Moreover, a CA will not only create but also distribute and when necessary revoke the issued certificates. Identification of applying entities can be delegated by the CA to a so-called registration authority (RA) that can record and verify some or all of the information needed by the CA to issue a certificate or a CRL and to perform other certificate management functions [7]. Distribution of certificates and CRLs can be achieved in several ways: the CA will publish them via a directory service (e.g. LDAP[3]) or via some HTTP/FTP-based repositories, or certificate subscribers can directly share certificates among each other. Revocation and suspension of public-key certificates is vital whenever a subscriber detects or suspects that her private key is compromised. The CA have the obligation to issue the so-called CRLs – black-lists on which revoked certificates are published so that PKC participants can be informed about the date and reason of revocation and also help them avoid using compromised certificates.

The documents that govern the activity of a PKI is called the certificate policy (CP) and the corresponding certification practice statement (CPS). They contain the declared and assumed set of rules and procedures followed by the CA and its subordinated CAs and RAs during their activities. These documents indicate the applicability of a certificate issued within such a PKI with respect to a particular community and class of application with common security requirements. A CP can help for example a user to decide whether such a certificate should be trusted or not for electronic data interchange (EDI) transactions [9].

As mentioned, Single Sign-On is the mechanism whereby a single action of authentication and authorization can permit a user to access all applications, computers and systems where she is allowed to, without the need to enter multiple IDs and passwords. A PKI is most suited for implementing SSO since it provides a solution for the need of authentication: it can be used to authenticate people, avoiding the need to remember dozens of PINs and passwords. Automatic authentication and authorization can be obtained by a cryptographic challenge based on users' public keys: one public key (i.e. digital certificate) per user, per network node, per application server.

Moreover, a PKI side-effect is the legal value it provides for electronic documents [10].

III. PKI PLANNING AND DEPLOYMENT

Deploying a successful public-key infrastructure requires a great deal of analysis, planning and preparation. There are several steps to be performed.

A. *Analysis*

The analysis must be driven by two major vectors: first, the scope and objectives to be achieved; second, the resources at hand.

The current security status of the university IT infrastructure must be the first question to be answered. This analysis must take into account what are the current risks

[2] ITU-T = The Telecommunication standardization sector coordinates standards for telecom on behalf of ITU (International Telecommunications Union).

associated to identity fraud in the set of applications used by the academic community. This means that a survey of applications must first be performed and then observed the potential security vulnerabilities. The security analyst must also understand if the organization will be able to respect the applicable privacy laws concerning the sensitive information about the people – informationa that might be required to be stored over large periods of time within the PKI.

It is equally important to investigate the community and to determine its reaction to new technologies and change of security paradigm. It is not a rare case that reluctant users become a barrier to the succes of a new/different security infrastructure/paradigm. Trainings and awareness sessions must be devised in order to draw the attention of community to the security risks and consequently to convince them that the change is positive since it eliminates (or at least reduces) the existing vulnerabilities. Also the initiative must be stimulated for those people that want to know more about security topics.

During this phase and still about the community, it is a good oportunity to understand how well the administration of the university tackles the problems of authentication (identification and verification) and authorization of the academic community (students, teaching, technical and administrative staff). One thing to find out is if there are already in place and if so, how strong they are, the controls used for strong authentication purposes. Of course it is also useful to determine if there exists a written central security policy enforced over the entire IT infrastructure of the university.

Last but not least important are the financial costs for planning, implementing and operating such a security infrastructure. Of course, given that we focus on an educational institution free-of-charge open-source PKI solutions must have precedence over customized commercial solutions. Operational costs can be reduced if part of the academic personnel (Master and Ph.D. students) is properly motivated to get involved in making the project work.

B. Technical requirements

From the administrative point of view a certain inertia will always be present and must be expected. Technologies other than PKI may also offer *good enough* safety. The power of PKI is the fact that most security requirements (data confidentiality, data integrity, data and user authentication, non-repudiation, authorization management) are delivered within a single technology. It is also worth pointing out that a common security infrastructure is easier to administer and more convenient to operate.

A supplementary effort must be performed to make clear what is the set of applications that will take advantage of the new security mechanisms and how many of them need to be engineered (e.g. modified, re-build, re-configured) in order to align with the new security infrastructure. Perhaps creating a focus group of students, teaching, administrative and technical staff to manage expectations and to get a level of commitment to the initiative will only improve the chances of success for the project.

The interoperability of different PKIs is also a interesting technical subject. There are several PKI structures to choose from. First, the single or stand-alone model in which we have one CA, one or more associated RAs, one or more public repositories, its certificate subscribers, and other supporting servers. Second, there is the hierarchical model in which the model above is replicated within a hierarchy. Third, there is the network model in which the stand-alone PKIs for interoperability purposes will issue to each other the so-called cross-certificates (a PKI will create one certificate for each PKI with which it wants to establish a trust relationship). Fourth, there is the PGP approach. PGP is a freeware e-mail system that uses an unstructured authentication framework in which users are free to decide whom they trust.

We consider that the hierarchical model is the most adequate one for an academic organization. University of Craiova is clearly a hierarchic organization with 17 faculties, 2 colleges and 9 special-purpose departments all subordinated to a central administration. In a hierarchical model the trust relationship between users of different CAs is easily resolved as long as the CAs are placed under a common top level CA. This model is flexible also when the community is large since authority and trust delegation is a clear advantage of such an architecture. Additionally, placing a new CA inside a certain hierarchy would implicitly offer the interoperability advantage with all the other CAs placed under the same hierarchy eliminating the need for cross-certification as described above. From an operational point of view, there is another advantage for a hierarchical model. The divide et impera approach can be used to delegate tasks from a top-level CA to a newly created subordinate CA dedicated for a particular faculty or department community of certificate subscribers.

C. Supported Services

Once the security infrastructure is designed, implemented and in operation, applications using it will also become more secure and also the institution's network communications will be safer.

It is a fact that most application-level protocols used in Internet today are transmitting by default user data in clear. This data may sometimes be sensitive for the user. If a user would have a choice about it, she will certainly prefer that confidentiality of her data be preserved at all times. Furthermore, since data is transmitted most of the time in clear, this makes it relatively easy for people who have acces to the network through which your data passes to install special programs (e.g. *sniffers*, *packet tracers*) and consequently grab all data available on the wire. Suppose

[3] The Lightweight Directory Access Protocol (LDAP) is used to query and modify data using directory services running over TCP/IP.

that among this data usernames and passwords may be intercepted. An adversary holding this information could read your e-mail box, login into different applications, in short, he controls all your data and access rights. To avoid this you must use an encrypted form of communication. The encryption can be achieved by means of SSL/TLS protocol [13]. This protocol provides confidentiality and data integrity between two communicating applications/entities. Moreover, the TLS handshake protocol allows the server and the client to authenticate each other and to negotiate the cryptographic setup of their private communication. Using asymmetric cryptography it is extremely easy to validate peer's identity.

Safe IT systems must be capable to support security also at network level. This could represent an alternative for the channel-oriented applications when the SSL/TLS protocol is not present. The IP Security Architecture (IPSec) [14] defines basic security mechanisms at network level, adding

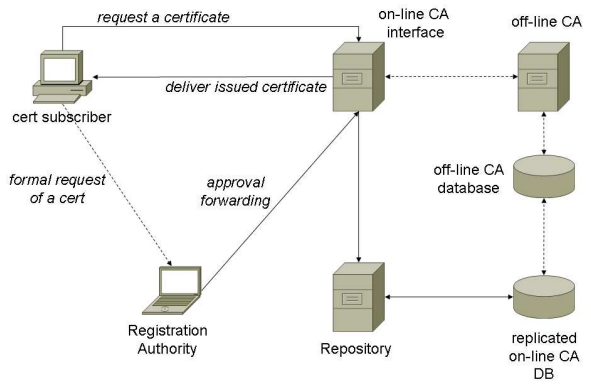


Fig. 1. A traditional Certification Authority architecture

two dedicated and easily insertable extensions to the IP packet: the authentication header (AH) [15] and the encapsulating security payload (ESP) [16]. AH is used to ensure authenticity and integrity of the IP packet, while ESP provides packet integrity, anti-replay service, and most importantly, data encapsulation with encryption ensuring thus that only the destination node can read the payload conveyed by the IP packet. The related Internet key exchange (IKE) [17] is a protocol designed to provide authenticated keys for use with the Internet security association and key management protocol (ISAKMP) [17]. IKE and ISAKMP define a generic architecture for authenticated security associations set up and key exchange without specifying the actual algorithms to be used. As far as the public-key cryptography is concerned, IKE imposes neither the use of a particular digital signature algorithm nor a particular key distribution method. The use of X.509 certificates has been largely accepted and the most commonly implemented standard.

For message-oriented applications a different approach can be considered. By means of access control lists (ACL) and public-key cryptography, authentication and consequently authorization can be achieved within these applications requiring no longer a password, instead the username and the digital certificate will suffice. The popular MIME standards [18] describe how electronic mail messages are structured. The S/MIME standards provide a consistent way to send and receive secure MIME data specifying how the encrypted information and the digital certificates can be included in the MIME message body. In doing that S/MIME respects the syntax provided by the RSA's Public-Key Cryptography Standard 7 (PKCS#7) [20]. PKCS#7 (a.k.a. cryptographic message syntax) describes the general syntax for data that may have elements of cryptography applied to it, such as digital signatures and digital envelopes. S/MIME uses digital certificates and provides the following crypto services to electronic messaging applications: data origin authentication, message integrity, non-repudiation of message signature, and data privacy. S/MIME can be used by traditional mail user agents to add cryptographic security services to mail that is sent and also to interpret them upon receipt. S/MIME is not restricted to e-mail; it can be used with any transport mechanism that carries MIME data, e.g HTTP. Furthermore, S/MIME can be used in automated message transfer agents that use cryptographic security services that do not require any human intervention, such as the signing of software-generated documents and the encryption of fax messages sent over the Internet.

Other important services that need to be supported by the security infrastructure concern certificate status verification, time-stamping, notarization and perhaps server-delegated certificate validation. Applications based on PKC must take care to properly retrieve and verify the validity and revocation status of the digital certificates they use. A certificate can be used trustfully only when it is proved to be valid. Valid means not only that is correctly issued and still in the valid lifetime frame, but also that it has not been revoked yet. CAs traditionally issue CRLs (black lists with revoked/suspended certificates), but there is also another related service that provides just in time revocation status for a certificate. This service can be implemented by respecting the OCSP standard [21]. In order to automate the process for those applications that have already implemented support for this protocol (such as for example, mail user agents and web browsers), a PKI designer must provide within the extension set of the certificate support for the CRL distribution point and also for the OCSP responder URL. The advantage of OCSP over CRL consists in the fact that the information that is passed between the clients and the server is composed of small messages containing certificate status requests and responses concerning singular certificate(s) and moreover, the service is on-line. There is no longer necessary that an entire CRL (certainly larger than an OCSP response) to be downloaded and consulted.

A time-stamping service [22] allows to prove that a datum existed before a particular time and can be used as one component in building reliable non-repudiation services. Suppose for example that within an automated distributed document processing system pertaining to the university, there exists the necessity to check whether a digital signature applied to a document/message was valid or not before the corresponding certificate was revoked. In order to associate a datum with a particular moment of time, a time-stamping authority (TSA) needs to be designed and implemented within the PKI. The TSA uses a trustworthy source of time in order to include a reliable time value for each time-stamp token. To preserve confidentiality and data privacy the TSA will only stamp (via a digitally signature) a digest value of the document and will not examine in any way the imprint being stamped. The TSA can also be used to indicate the time of submission when an enforced deadline is critical, or to indicate transaction times within operational logs.

In today's not only academic but also commercial environments establishing a framework for the authentication of electronic information or a legal framework for electronic signature requires understanding the concepts and principles of both juridical and computer security fields. The problem rises from the fact that definitions in the two field for entities' signature and authentication are different. A need for compromise is obvious. On a nation-wide basis legal provisions have been produced for supporting electronic signature [10] and this can be introduced into daily document processings. PKI are the first step for supporting the actual creation of digital signatures on documents. Of course, decision must be taken from an administrative point of view whether to adopt or not the use of digital signature and e-documents in current document processing workflows.

D. Service deployment

Given the precedent analysis, the architecture proposed will provide a series of secure services such as : user authentication via public-key certificates, a personal security environment containing both the user's private key and the corresponding public-key certificate stored on personal computer filesystem or on smart-cards, S/MIME for message-oriented applications, either IPSec using IKE and X.509 certificates or SSL/TLS for channel-oriented applications, time stamping, certificate revocation status notification, electronic signature.

When choosing to implement a PKI-based system management people have to consider a series of factors that will be detailed in what follows. The service deployment must include the following steps. First, a root CA must be deployed. Figure 1 presents schematically the interactions and the architecture of a certification authority. The CA is designed as follows: on-line there will be the CA public interface that will gather certificate issuance and certificate revocation requests; a repository will be provided for subscribers and end-users to allow retrieval of both issued certificates and certificate revocation lists. The actual machine that performs public-key certificate and CRL signing will be kept off-line at all times in order to reduce potential attacks through the net. The link between the repository, the on-line interface and the off-line CA machine will be insured by the CA administrator. Certificate requests arrived on the on-line CA interface (directly or via a RA) will be collected by the CA administrator, loaded manually on the off-line CA machine, examined, and if successfully validated new certificates will be issued. Then these new certificates are collected by the administrator and manually transferred into the public repository. At this point the PKI subscribers will be in possession of their security information and can decide in what applications their personal security

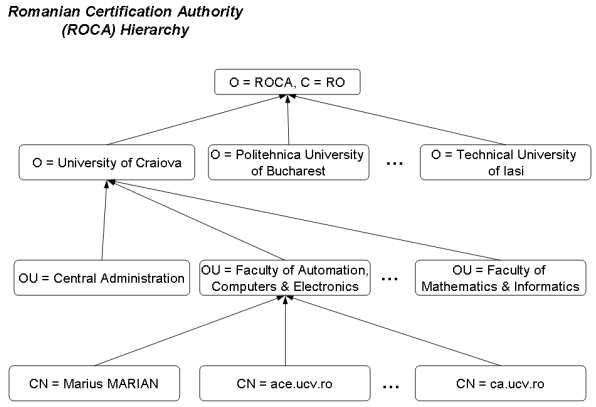


Fig. 2. Example of the envisioned hierarchy of CAs within the Romanian Certification Authority PKI

environment will be included: web browsers, mail user agents (e.g. MS Outlook, Mozilla Thunderbird, etc.), channel-oriented applications that make use of IPSec, SSL/TLS, message-oriented applications (e.g. using S/MIME) or applications able to perform digital signature creation and verification.

To solve future interoperability requirements, we propose to establish a hierarchic PKI called ROCA (standing for Romanian Certification Authorities for the academic sector). The proposed structure of this PKI is depicted in Figure 2. Under the root CA of ROCA, most universities and other education-related institutions interested may adhere by creating a top-level CA for each such institution. Under each top-level CA, subordinate CAs may be established. For example, under the top-level CA of University of Craiova, there might be established 18 subordinate CAs, one for each faculty of the University and also for the central administration. This delegation of tasks from a top-level CA to a subordinate one makes sense if we think for example that the Faculty of Automation, Computers and Electronics has a community formed of approximately 1600 students (for bachelor, master and doctoral directions of study), 66 persons for the teaching staff and 34 persons for the administrative and technical positions.

In order to delegate the process of registering users and verifying their identities, the activation of one or more RAs should be considered within each CA of the PKI hierarchy. In what concerns the users' identities a specific procedure must be enforced: the certificate will only be issued upon receipt of a formal written request accompanied by a copy of a legal identity card that authenticates the request submitter. A decision must be taken to include an identifier in the certificate for each user – a good candidate is the matriculation number used for each person registered with the University.

For the CA set-up, any software can be used. As detailed in the analysis section above, to reduce costs it would be good to use an open-source one. A first alternative for our PKI is the OpenCA[4] implementation which is based on the well-known OpenSSL[5] cryptographic library.

IV. Conclusion

PKI represents today a robust technology that provides a complete security solution to any type of organization. It delivers strong authentication, data confidentiality and integrity and gives excellent support for non-repudiation. It also enables and facilitates a centralized management of access rights.

The proposed design will be implemented as part of a secure IT infrastructure of the University of Craiova. The first pilot implementation will be accomplished for the community of the Faculty of Automation, Computers and Electronics.

Acknowledgment

This work was supported by the strategic grant POSDRU/89/1.5/S/61968, Project ID61968 (2009), co-financed by the European Social Fund within the Sectorial Operational Program Human Resources Development 2007 – 2013.

References

[1] D. R. Kuhn, V.C. Hu, W. T. Polk, S. J. Chang, *Introduction to public-key technology and the federal PKI infrastructure*, National Institute of Standards and Technologies, SP 800-32, 26 February 2001.

[2] NIST – National Institute of Standards and Technologies, *Digital Signature Standard*, FIPS PUB (Federal Information Processing Standards Publication) 186-3, June 2009.

[3] R. Rivest, A. Shamir, L. Adleman, *RSA Cryptographic Standard*, RSA Laboratories – Public-Key Cryptographic Standard (PKCS) #1 v2.1, 2002.

[4] J. Callas, L. Donnerhacke, H. Finney, R. Thayer, *OpenPGP Message Format*, RFC (Request For Comments) 4880, Internet Engineering Task Force (IETF), November 2007.

[5] W. Diffie, M. E. Hellman, *New Directions in Cryptography*, IEEE Transactions on Information Theory, vol. IT-22, Nov. 1976, pp: 644–654.

[6] ITU-T Recommendation X.509, *Information Technology - Open Systems Interconnection - The Directory: Public-key and attribute certificates frameworks*, 2005.

[7] C. Adams, S. Farrell, T. Kause, T. Mononen, *Internet X.509 Public Key Infrastructure: Certificate Management Protocol*, IETF 4210, September 2005.

[8] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, W. Polk, *Internet X.509 Public Key Infrastructure Certificate and CRL Profile*, IETF RFC 5280, May 2008.

[9] NIST, *Electronic Data Interchange*, FIPS PUB 161-2, April 1996.

[10] Parlamentul României, *Legea nr. 455 din 18 iulie 2001 privind semnătura electronică*, publicată în Monitorul Oficial, Partea I nr. 429 din 31 iulie 2001.

[11] C. Adams, M. Burmester, Y. Desmedt, M. Reiter, P. Zimmermann, *Which PKI is the right one?*, Proceedings of the 7th ACM conference on Computer and communications security CCS'00, Athens, Greece, November 2000.

[12] Organization for the Advancement of Structured Information Standards (OASIS), *PKI Action Plan*, OASIS PKI Technical Committee Report, February 2004.

[13] T. Dierks, E. Rescorla, *The Transport Layer Security (TLS) Protocol v1.2*, IETF RFC 5246, August 2008.

[14] S. Kent, *Security Architecture for the Internet Protocol*, IETF RFC 4301, December 2005.

[15] S. Kent, *IP Authentication Header (AH)*, IETF RFC 4302, December 2005.

[16] S. Kent, *IP Encapsulating Security Payload (ESP)*, IETF RFC 4303, December 2005.

[17] C. Kaufman, *Internet Key Exchange Protocol (IKEv2) Protocol*, IETF RFC 4306, December 2005.

[18] N. Freed, N. Borenstein, *Multipurpose Internet Mail Extensions (MIME), Parts one and two*, IETF RFC 2045 – 2046, November 1996.

[19] B. Ramsdell, S. Turner, *Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.2 Message Specification*, IETF RFC 5751, January 2010.

[20] R. Rivest, A. Shamir, L. Adleman, *RSA Cryptographic Message Syntax Standard*, RSA Laboratories – Public-Key Cryptographic Standard (PKCS) #7 v1.5, November 1993.

[21] M. Myers, R. Ankney, A. Malpani, S. Galperin, C. Adams, *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*, IETF RFC 2560, June 1999.

[22] C. Adams, P. Cain, D. Pinkas, P. Zuccherato, *Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP)*, IETF RFC 3161, August 2001.

[4] OpenCA project available on-line at http://www.openca.org.

[5] OpenSSL project available on-line at http://www.openssl.org.

Switched linear systems in discrete-time: Criteria for the existence of invariant sets

Mihaela-Hanako Matcovschi, Octavian Pastravanu*, and Adrian Alecu

***Abstract*—This paper focuses on the existence of exponentially contractive sets that are invariant with respect to the trajectories of discrete-time switched linear systems. These sets are described by the following *attributes*: shape, state-variable scaling factors and decreasing rate. We are interested both in analysis (to test if the sets characterized by certain attributes are invariant with respect to the trajectories of a given switched system) and synthesis (to construct state feedbacks so as the sets characterized by certain attributes are invariant with respect to the trajectories of the closed-loop switched system). For the usual Holder p-norms corresponding to $p \in \{1, 2, \infty\}$ we also discuss the numerical tractability of the theoretical results. A numerical example is included for practical illustration.**

I. Introduction

A. Notations

The following notations will be used throughout the paper:

• For a vector $\boldsymbol{x} \in \mathbb{R}^n$:

$\|\boldsymbol{x}\|_p$ is the Hölder vector p-norm and $1 \le p \le \infty$;

$\|\boldsymbol{x}\|_p^{\boldsymbol{D}} = \|\boldsymbol{D}^{-1}\boldsymbol{x}\|_p$ for $\boldsymbol{D} = \mathrm{diag}\{d_1, \ldots, d_n\}$ a positive definite diagonal matrix, i.e. $d_i > 0$, $i = 1, \ldots n$;

• For a matrix $\boldsymbol{M} \in \mathbb{R}^{n \times n}$:

$\|\boldsymbol{M}\|_p$ is the matrix norm induced by the vector norm $\|\bullet\|_p$ through $\|\boldsymbol{M}\|_p = \sup_{\boldsymbol{x} \in \mathbb{R}^n, \boldsymbol{x} \neq 0} \dfrac{\|\boldsymbol{M}\boldsymbol{x}\|_p}{\|\boldsymbol{x}\|_p} = \max_{\boldsymbol{x} \in \mathbb{R}^n, \|\boldsymbol{x}\|_p = 1} \|\boldsymbol{M}\boldsymbol{x}\|_p$;

$\|\boldsymbol{M}\|_p^{\boldsymbol{D}} = \|\boldsymbol{D}^{-1}\boldsymbol{M}\boldsymbol{D}\|_p$ is the matrix norm induced by the vector norm $\|\bullet\|_p^{\boldsymbol{D}}$;

$\boldsymbol{M}^T$ denotes the transpose of matrix $\boldsymbol{M}$;

$S(\boldsymbol{M}) = \{z \in \mathbb{C} \mid \det(z\boldsymbol{I} - \boldsymbol{M}) = 0\}$ is the spectrum of $\boldsymbol{M}$, and $\lambda_i(\boldsymbol{M}) \in S(\boldsymbol{M})$, $i = 1, \ldots, n$, denote its eigenvalues.

If $\boldsymbol{M} \in \mathbb{R}^{n \times n}$ is symmetric, then "$\boldsymbol{M} \succ 0$" ("$\boldsymbol{M} \prec = 0$") means "$\boldsymbol{M}$ is positive definite" (negative semidefinite).

• If $\boldsymbol{X} \in \mathbb{R}^{n \times m}$, then $|\boldsymbol{X}|$ represents the nonnegative matrix (for $m \ge 2$) or vector (for $m = 1$) defined by taking the absolute values of the entries of $\boldsymbol{X}$.

• If $\boldsymbol{X}, \boldsymbol{Y} \in \mathbb{R}^{n \times m}$, then "$\boldsymbol{X} \le \boldsymbol{Y}$", "$\boldsymbol{X} < \boldsymbol{Y}$" mean componentwise inequalities.

Manuscript received September 1, 2010. This work was supported by CNMP, Romania, under Grant 12100/1.10.2008 - SICONA.

The authors are with the Department of Automatic Control and Applied Informatics from the Technical University "Gh. Asachi" of Iasi, Blvd. Mangeron 27, 700050 Iasi, Romania (e-mail: opastrav@ac.tuiasi.ro).

B. Theoretical prerequisites on switched systems and invariant sets

A *switched linear system* (SLS) consists of a finite family of linear subsystems and a set of rules that regulates the switching between these subsystems. The commutation events can be classified into time-driven or state-driven, being autonomous (uncontrolled) or controlled [1].

This paper focuses on discrete-time SLSs which are mathematically described by a collection of indexed linear difference equations:

$$\begin{aligned} &\boldsymbol{x}(t+1) = \boldsymbol{A}_{\sigma(t)}\boldsymbol{x}(t), \;\; \boldsymbol{x}(t_0) = \boldsymbol{x}_0 \in \mathbb{R}^n, \\ &t, t_0 \in \mathbb{Z}_+, \;\; t \ge t_0, \end{aligned} \tag{1}$$

where $\mathbb{Z}_+$ stands for the set of non-negative integers. The logical rule that commands the switching between the subsystems is represented by the switching signal $\sigma : \mathbb{Z}_+ \to \{1, \ldots, N\}$. At any time instant $t \in \mathbb{Z}_+$, the index $s = \sigma(t)$ determines the *active mode* of the SLS

$$\boldsymbol{x}(t+1) = \boldsymbol{A}_s\boldsymbol{x}(t), \;\; s \in \{1, \ldots, N\}, \tag{1a}$$

characterized by the $n \times n$ matrix $\boldsymbol{A}_s \in \boldsymbol{\mathcal{A}} = \{\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_N\}$. Starting at t_0 from the initial state $\boldsymbol{x}_0$, the trajectory of SLS (1) depends on the switching sequence σ and is denoted in the sequel by $\boldsymbol{x}(t) = \boldsymbol{x}(t; t_0, \boldsymbol{x}_0, \sigma)$.

SLSs have attracted increasingly more attention since the 1990s and have been extensively discussed by recent monographs such as [1] and [2]. For switched systems different stability problems can be approached, depending on whether stability should hold for arbitrary switching signals, for a certain class of switching signals (constrained switching) or for a given switching signal (stabilizability through switching). In a general setting, the main aspects regarding the stability theory of switched and hybrid systems are approached in [3]. The latest survey paper [4] presents an up-to-date synthesis of results in the stability and stabilizability of switched linear systems, in both continuous- and discrete-time, based on an extensive reference list.

The investigation of *invariant sets* emerged as a research trend in mathematics at the middle of the 20-th century, as pointed out by the monograph [5]. In the 1980s, the theoretical results were applied in control engineering for developing new analysis and design techniques. The progress achieved until the end of the century is discussed in the tutorial paper [6] that concentrates on linear systems.

The applicability of the flow-invariance concepts in the qualitative theory of dynamical systems is amply discussed by the recent monograph [7] whose reference list contains more than 300 works representing noticeable contributions to this research trend.

Generally speaking, a *set* (constant or time-dependent) is (*positively*) *invariant* with respect to the trajectories of a dynamical system, if any trajectory initiated inside that set does not leave it any longer. A particular type of invariant sets investigated by many researches is the so called "contractive sets" (for the exact definition see [6]), the interest focusing on sets with polyhedral shapes. Unlike constant invariant sets, the contractive ones constrain system trajectories to approach the equilibrium point (always existing for linear systems). By considering exponentially contractive invariant sets, with rectangular shapes, papers such as [8], [9] defined and characterized a stronger type of exponential stability, called *componentwise exponential asymptotic stability* (abbreviated CWEAS). Later on, paper [10] showed that, similarly to CWEAS but aiming at a general picture, a stronger type of exponential stability can be defined for any Hölder *p*-norm, $1 \le p \le \infty$. The proposed terminology was *diagonally invariant exponential stability* (abbreviated DIES) relative to *p*-norms, and CWEAS represented the particular case $p = \infty$. Actually, the *p*-norm configures the shapes of the contractive sets that decrease exponentially to the equilibrium $\{0\}$.

The current paper explores the exponentially contractive sets corresponding to the DIES of discrete-time switched linear systems. Since DIES is a stronger property than standard stability, the results on DIES for switched systems are more conservative than those referring to the stability / stabilizability of switched systems (reported in the papers cited by [1] or [2], and in the more recent ones, such as [3], [11], [4]. However, from the mathematical point of view, this conservativeness is motivated by the fact that the results on DIES we are going to present express necessary and sufficient conditions. These results are in full accordance with the continuous-time case we have addressed in [12].

In our work, the exponentially contractive sets are described by the following *attributes*: shape, state-variable scaling factors and decreasing rate. We are interested both in analysis (to test if the sets characterized by certain attributes are invariant with respect to the trajectories of a switched system) and synthesis (to construct state feedbacks so as the sets characterized by certain attributes are invariant with respect to the trajectories of the closed-loop switched system).

This paper is organized as follows. Section II provides analysis techniques for testing the existence of contractive invariant sets with given attributes. Section III considers the design of a closed loop switched system with state feedback that exhibits invariant sets with given attributes. In both Sections II and III, the first part is devoted to the theoretical development and the second part discusses the numerical tractability. Section IV illustrates the applicability of our results.

II. CONTRACTIVE INVARIANT SETS IN THE DISCRETE TIME DYNAMICS

A. Theoretical development

Definition 1.

Let $1 \le p \le \infty$, $\boldsymbol{D} \succ 0$ diagonal, and $0 < r < 1$. The SLS (1) is called *diagonally invariant exponentially stable relative to the p-norm and parameters* $\boldsymbol{D}$, *r* (abbreviated as $\mathrm{DIES}_p^{\boldsymbol{D},r}$) under arbitrary switching if

$$\forall \varepsilon > 0, \forall t, t_0 \in \mathbb{Z}_+, t \ge t_0, \forall \boldsymbol{x}_0 = \boldsymbol{x}(t_0) \in \mathbb{R}^n, \forall \sigma : \mathbb{Z}_+ \to \{1, \ldots, N\}:$$
$$\| \boldsymbol{x}_0 \|_p^{\boldsymbol{D}} \le \varepsilon \Rightarrow \| \boldsymbol{x}(t; t_0, \boldsymbol{x}_0, \sigma) \|_p^{\boldsymbol{D}} \le \varepsilon r^{(t-t_0)}. \quad \blacksquare \quad (2)$$

Remark 1.

In terms of invariant sets Definition 1 is equivalent to the existence of a positive definite diagonal matrix $\boldsymbol{D} \succ 0$ and a constant $0 < r < 1$ ensuring that the exponentially decreasing time-dependent sets:

$$X_p^{\boldsymbol{D},r}(\varepsilon; t, t_0) = \left\{ x \in \mathbb{R}^n \,\middle|\, \| x \|_p^{\boldsymbol{D}} \le \varepsilon\, r^{(t-t_0)} \right\}, \quad t, t_0 \in \mathbb{Z}_+,\ t \ge t_0,\ \varepsilon > 0, \quad (3)$$

are invariant with respect to the solutions (state-space-trajectories) of system (1) for any switching sequence σ.

For the usual norms corresponding to $p \in \{1, 2, \infty\}$, the sets defined by (3) have well-known geometric shapes (i.e. hyper-diamonds for $p = 1$, ellipses for $p = 2$, and rectangles for $p = \infty$) scaled in accordance with the diagonal entries of matrix $\boldsymbol{D}$. The constant $0 < r < 1$ expresses the contraction rate of the considered sets. $\blacksquare$

Our study of $\mathrm{DIES}_p^{\boldsymbol{D},r}$ for discrete-time SLSs is based on Theorem 2 in [10], stated in the sequel for a time-invariant linear system:

$$\boldsymbol{x}(t+1) = \boldsymbol{A}\boldsymbol{x}(t),\ \ \boldsymbol{x}(t_0) = \boldsymbol{x}_0 \in \mathbb{R}^n,\ \ t, t_0 \in \mathbb{Z}_+,\ \ t \ge t_0, \quad (4)$$

whose solution is denoted by $\boldsymbol{x}(t) = \boldsymbol{x}(t; t_0, x_0)$.

Theorem 1.

Let $1 \le p \le \infty$, $\boldsymbol{D} \succ 0$ diagonal and $0 < r < 1$.

The following statements are equivalent:

(*i*) System (4) is $\mathrm{DIES}_p^{\boldsymbol{D},r}$, i.e.

$$\forall \varepsilon > 0, \forall t, t_0 \in \mathbb{Z}_+, t \ge t_0, \forall \boldsymbol{x}_0 = \boldsymbol{x}(t_0) \in \mathbb{R}^n:$$
$$\| \boldsymbol{x}_0 \|_p^{\boldsymbol{D}} \le \varepsilon \Rightarrow \| \boldsymbol{x}(t; t_0, x_0) \|_p^{\boldsymbol{D}} \le \varepsilon r^{(t-t_0)}. \quad (5)$$

(*ii*) $V(\boldsymbol{x}) = \| \boldsymbol{x} \|_p^{\boldsymbol{D}}$ is a strong Lyapunov function for system (4), with the decreasing rate *r*, i.e.

$$\forall t \in \mathbb{Z}_+,\ \forall \text{ solution } \boldsymbol{x}(t) \text{ to (4)},\ V(\boldsymbol{x}(t+1)) \le r\, V(\boldsymbol{x}(t)). \quad (6)$$

(*iii*) $\| \boldsymbol{A} \|_p^{\boldsymbol{D}} \le r$. $\blacksquare$ (7)

The following result represents a natural extension of Theorem 1 to the case of SLSs.

Theorem 2.

Let $1 \le p \le \infty$, $\boldsymbol{D} \succ 0$ diagonal and $0 < r < 1$.

The following statements are equivalent:

(*i*) The SLS (1) is $\text{DIES}_p^{\boldsymbol{D},r}$ under arbitrary switching.

(*ii*) $V(\boldsymbol{x}) = \|\boldsymbol{x}\|_p^{\boldsymbol{D}}$ is a common strong Lyapunov function for all the subsystems of (1), with the decreasing rate *r*.

(*iii*) $\|\boldsymbol{A}_s\|_p^{\boldsymbol{D}} \le r$, $s = 1,\ldots,N$. (8)

Proof: (*i*) $\Rightarrow$ (*ii*) $\Rightarrow$ (*iii*). For every $s \in \{1,\ldots,N\}$, consider the particular switching signals $\sigma(t) \equiv s$, then apply Theorem 1.

(*iii*) $\Rightarrow$ (*i*). We construct a proof by contradiction. We assume there exist a positive $\varepsilon > 0$ and a switching signal σ so that the set $X_p^{\boldsymbol{D},r}(\varepsilon; t, t_0)$ is not invariant with respect to the solution $\boldsymbol{x}(t; t_0, \boldsymbol{x}_0, \sigma)$ initialized inside it. This means there exists a moment t^* when the considered solution leaves the set, i.e. $\|\boldsymbol{x}(t; t_0, \boldsymbol{x}_0, \sigma)\|_p^{\boldsymbol{D}} \le \varepsilon r^{(t-t_0)}$, for $t_0 \le t \le t^*$, and $\|\boldsymbol{x}(t^*+1; t_0, \boldsymbol{x}_0, \sigma)\|_p^{\boldsymbol{D}} > \varepsilon r^{(t^*+1-t_0)}$. Let $s^* = \sigma(t^*)$ be the index of the active mode of the SLS at t^*. Taking $\boldsymbol{x}(t^*+1; t_0, \boldsymbol{x}_0, \sigma) = \boldsymbol{A}_{s^*} \boldsymbol{x}(t^*; t_0, \boldsymbol{x}_0, \sigma)$ into account, we get $\|\boldsymbol{x}(t^*+1; t_0, \boldsymbol{x}_0, \sigma)\|_p^{\boldsymbol{D}} \le \|\boldsymbol{A}_{s^*}\|_p^{\boldsymbol{D}} \, \|\boldsymbol{x}(t^*; t_0, \boldsymbol{x}_0, \sigma)\|_p^{\boldsymbol{D}} \le r\varepsilon r^{(t^*-t_0)}$, which contradicts the hypothesis and completes the proof. ■

Remark 2.

As pointed out in [4], the existence of a common quadratic Lyapunov function is only sufficient for the asymptotic stability of switched linear systems under arbitrary switching signals, and could be rather conservative. Theorem 2 shows that the existence of a common Lyapunov function defined by $V(\boldsymbol{x}) = \|\boldsymbol{x}\|_p^{\boldsymbol{D}}$ with a positive definite diagonal matrix $\boldsymbol{D}$ is a necessary and sufficient condition for the $\text{DIES}_p^{\boldsymbol{D},r}$ of SLSs under arbitrary switching signals. ■

B. Numerical tractability

In this section we are interested in pointing out the numerical approach to $\text{DIES}_p^{\boldsymbol{D},r}$ analysis for the switched system (1) in the particular cases of the usual Hölder norms corresponding to $p \in \{1, 2, \infty\}$.

The results are presented as follows.

Corollary 1.

(*i*) For $p = 1$, inequalities (8) are equivalent to

$$|\boldsymbol{A}_s|^T \boldsymbol{\delta} \le r\boldsymbol{\delta}, \; s = 1,\ldots,N, \tag{9}$$

where

$$\boldsymbol{\delta} = [\delta_1 \ldots \delta_n]^T \in \mathbb{R}^n, \; \delta_i = 1/d_i, \; i = 1,\ldots,n, \tag{10}$$

is a positive vector formed with the inverses of the diagonal entries of matrix $\boldsymbol{D}$.

(*ii*) For $p = 2$, inequalities (8) are equivalent to

$$\boldsymbol{A}_s^T \boldsymbol{P} \boldsymbol{A}_s - r^2 \boldsymbol{P} \preceq 0, \; s = 1,\ldots,N, \tag{11}$$

where $\boldsymbol{P} = (\boldsymbol{D}^{-1})^2$.

(*iii*) For $p = \infty$, inequalities (8) are equivalent to

$$|\boldsymbol{A}_s| \, \boldsymbol{d} \le r\boldsymbol{d}, \; s = 1,\ldots,N, \tag{12}$$

where $\boldsymbol{d} = [d_1 \ldots d_n]^T \in \mathbb{R}^n$ is a positive vector formed with the diagonal entries of matrix $\boldsymbol{D}$.

Proof: For a simultaneous handling of all matrices $\boldsymbol{A}_s$, $s = 1,\ldots,N$, involved in (9), (11) and (12), let us consider a generic notation $\boldsymbol{M} = [m_{ij}]$, $i, j = 1,\ldots,n$, when referring to a real square matrix.

(*i*) For $p = 1$ we have

$$\|\boldsymbol{M}\|_1^{\boldsymbol{D}} \le r \Leftrightarrow \max_{1 \le j \le n} \left(|m_{ii}| + \sum_{i=1, i \ne j}^{n} |m_{ij}| \frac{d_j}{d_i} \right) \le r \Leftrightarrow$$

$$\max_{1 \le j \le n} \left(|m_{ii}| + \sum_{i=1, i \ne j}^{n} |m_{ij}| \frac{\delta_i}{\delta_j} \right) \le r \Leftrightarrow$$

$$|m_{jj}| \delta_j + \sum_{i=1, i \ne j}^{n} |m_{ij}| \delta_i \le r\delta_j, \; j = 1,\ldots,n \Leftrightarrow |\boldsymbol{M}|^T \boldsymbol{\delta} \le r\boldsymbol{\delta},$$

where vector $\boldsymbol{\delta}$ is defined by (10). Therefore, for $p = 1$ inequalities (8) are equivalent to (9).

(*ii*) For $p = 2$ inequalities (8) become $\|\boldsymbol{M}\|_2^{\boldsymbol{D}} \le r \Leftrightarrow (\lambda_{\max}[\boldsymbol{D}\boldsymbol{M}^T (\boldsymbol{D}^{-1})^2 \boldsymbol{M}\boldsymbol{D}])^{1/2} \le r$. Consequently we obtain $\lambda_{\max}[\boldsymbol{D}\boldsymbol{M}^T (\boldsymbol{D}^{-1})^2 \boldsymbol{M}\boldsymbol{D} - r^2 \boldsymbol{I}] \le 0$, which is equivalent to $\boldsymbol{D}\boldsymbol{M}^T (\boldsymbol{D}^{-1})^2 \boldsymbol{M}\boldsymbol{D} - r^2 \boldsymbol{I} \preceq 0$. By left and right multiplication by the positive definite diagonal matrix $\boldsymbol{D}^{-1}$, we get the Stein matrix inequality $\boldsymbol{M}^T \boldsymbol{P} \boldsymbol{M} - r^2 \boldsymbol{P} \preceq 0$ written for $\boldsymbol{P} = (\boldsymbol{D}^{-1})^2$, which completes the proof.

(*iii*) For $p = \infty$ we have

$$\|M\|_\infty^D \le r \Leftrightarrow \max_{1 \le i \le n} \left(|m_{ii}| + \sum_{j=1, j \ne i}^{n} |m_{ij}| \frac{d_j}{d_i} \right) \le r \Leftrightarrow$$

$$|m_{ii}| d_i + \sum_{j=1, \, j \ne i}^{n} |m_{ij}| d_j \le r d_i, \;\; j = 1,\ldots,n \Leftrightarrow |\boldsymbol{M}| \, \boldsymbol{d} \le r\boldsymbol{d}.$$

This proves that for $p = \infty$ inequalities (8) are equivalent to (12). ■

Remark 3.

According to Corollary 1, for the usual matrix norms corresponding to $p \in \{1, 2, \infty\}$, the analysis of $\text{DIES}_p^{\boldsymbol{D},r}$ of SLS (1) implies the usage of simple numerical tools. For $p \in \{1, \infty\}$ testing the inequalities (9) and (12) requires only matrix (vectors) multiplications, whereas for $p = 2$ inequality (11) involves eigenvalue computation. ■

III. FEEDBACK ALLOCATION OF CONTRACTIVE INVARIANT SETS

A. Theoretical development

Consider a switched linear system that commutes between $N \in \mathbb{N}$ subsystems:

$$\begin{aligned} &\boldsymbol{x}(t+1) = \boldsymbol{A}_{\sigma(t)}\boldsymbol{x}(t) + \boldsymbol{B}_{\sigma(t)}\boldsymbol{u}(t),\ \boldsymbol{x}(t_0) = \boldsymbol{x}_0, \\ &\boldsymbol{A}_\sigma \in \mathbb{R}^{n\times n},\ \boldsymbol{B}_\sigma \in \mathbb{R}^{n\times m},\ m \le n,\ t, t_0 \in \mathbb{Z}_+,\ t \ge t_0, \end{aligned} \tag{13}$$

governed by the switching signal $\sigma : \mathbb{Z}_+ \to \{1,\dots,N\}$. Let us choose a state-feedback built with a unique gain matrix:

$$\boldsymbol{u}(t) = -\boldsymbol{K}\boldsymbol{x}(t),\ \boldsymbol{K} \in \mathbb{R}^{m\times n}, \tag{14}$$

and define the closed-loop switched system

$$\begin{aligned} &\hat{\boldsymbol{x}}(t+1) = (\boldsymbol{A}_{\sigma(t)} - \boldsymbol{B}_{\sigma(t)}\boldsymbol{K})\hat{\boldsymbol{x}}(t),\ \hat{\boldsymbol{x}}(t_0) = \boldsymbol{x}(t_0), \\ &\quad t, t_0 \in \mathbb{Z}_+,\ t \ge t_0. \end{aligned} \tag{15}$$

Definition 2.

Let $1 \le p \le \infty$, $\boldsymbol{D} \succ 0$ diagonal and $0 < r < 1$.

If there exists a feedback (14) so that the closed-loop switched system (15) is $\mathrm{DIES}_p^{\boldsymbol{D},r}$ under arbitrary switching, then the switched system (13) is called $\mathrm{DIES}_p^{\boldsymbol{D},r}$-*stabilizable* through state-feedback. ■

Theorem 3.

Let $1 \le p \le \infty$, $\boldsymbol{D} \succ 0$ diagonal and $0 < r < 1$.

The switched system (13) is $\mathrm{DIES}_p^{\boldsymbol{D},r}$-stabilizable through the state-feedback (14) if and only if matrix $\boldsymbol{K}$ is a solution to the inequalities

$$\| \boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K} \|_p^{\boldsymbol{D}} \le r,\ \ s = 1,\dots,N. \tag{16}$$

Proof: We apply Theorem 2 to the closed-loop switched system (15). ■

Remark 4.

Let $1 \le p \le \infty$. Any $\mathrm{DIES}_p^{\boldsymbol{D},r}$-feedback of the switched system (13) ensures the assignment of the closed-loop eigenvalues in the Schur-stability region of the complex plane, $\mathbb{C}_S = \{ z \in \mathbb{C} \mid |z| < 1 \}$, for each subsystem of (15), $\hat{\boldsymbol{x}}(t+1) = (\boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K})\hat{\boldsymbol{x}}(t)$, $s = 1,\dots,N$. Indeed, if $\boldsymbol{M}$ denotes a real $n \times n$ matrix and $\lambda_i = \lambda_i(\boldsymbol{M})$, $i = 1,\dots,n$, are its eigenvalues, then $|\lambda_i| = |\lambda_i(\boldsymbol{D}^{-1}\boldsymbol{M}\boldsymbol{D})| \le \| \boldsymbol{M} \|_p^{\boldsymbol{D}}$ for all $i = 1,\dots,n$, according to [14, p.41]. Hence, for any $s = 1,\dots,N$, and $i = 1,\dots,n$, we have

$$|\lambda_i(\boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K})| = |\lambda_i(\boldsymbol{D}^{-1}(\boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K})\boldsymbol{D})| \le \| \boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K} \|_p^{\boldsymbol{D}} \le r < 1. \ ■$$

B. Numerical tractability

In this section we are interested in pointing out the numerical approach to $\mathrm{DIES}_p^{\boldsymbol{D},r}$-stabilizability of the switched system (13) in the particular cases of the usual Hölder norms corresponding to $p \in \{1, 2, \infty\}$.

Corollary 2.

(*i*) For $p = 1$, inequalities (16) are equivalent to the linear inequalities:

$$\begin{aligned} &-\boldsymbol{B}_s\boldsymbol{K} - \boldsymbol{G}_s \le -\boldsymbol{A}_s, \\ &\boldsymbol{B}_s\boldsymbol{K} - \boldsymbol{G}_s \le \boldsymbol{A}_s, \\ &\boldsymbol{G}_s^T\boldsymbol{\delta} \le r\boldsymbol{\delta}, \quad s = 1,\dots,N, \end{aligned} \tag{17}$$

where the vector $\boldsymbol{\delta} = [\delta_1 \dots \delta_n]^T \in \mathbb{R}^n$ has the same meaning as in Corollary 1 (*i*) and $\boldsymbol{K} \in \mathbb{R}^{m\times n}$, $\boldsymbol{G}_s \in \mathbb{R}^{n\times n}$, $s = 1,\dots,N$, are unknown matrices.

(*ii*) For $p = 2$, inequalities (16) are equivalent to the linear matrix inequalities

$$(\boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K})^T\boldsymbol{P}(\boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K}) - r^2\boldsymbol{P} \preceq 0,\ s = 1,\dots,N, \tag{18}$$

where $\boldsymbol{K} \in \mathbb{R}^{m\times n}$ is an unknown matrix and $\boldsymbol{P} = (\boldsymbol{D}^{-1})^2$.

(*iii*) For $p = \infty$, inequalities (16) are equivalent to the linear inequalities

$$\begin{aligned} &-\boldsymbol{B}_s\boldsymbol{K} - \boldsymbol{G}_s \le -\boldsymbol{A}_s, \\ &\boldsymbol{B}_s\boldsymbol{K} - \boldsymbol{G}_s \le \boldsymbol{A}_s, \\ &\boldsymbol{G}_s\boldsymbol{d} \le r\boldsymbol{d}, \quad s = 1,\dots,N, \end{aligned} \tag{19}$$

where the vector $\boldsymbol{d} = [d_1 \cdots d_n]^T \in \mathbb{R}^n$ has the same meaning as in Corollary 1 (*iii*), and $\boldsymbol{K} \in \mathbb{R}^{m\times n}$, $\boldsymbol{G}_s \in \mathbb{R}^{n\times n}$, $s = 1,\dots,N$, are unknown matrices.

Proof: (*i*) Corollary 1 (*i*) written for $\boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K}$ instead of $\boldsymbol{A}_s$ shows that inequalities (16) with $p = 1$ are equivalent to the existence of a matrix $\boldsymbol{K} \in \mathbb{R}^{m\times n}$ such that

$$|\boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K}|^T\boldsymbol{\delta} \le r\boldsymbol{\delta},\ s = 1,\dots,N. \tag{20}$$

First we prove the implication "(20) $\Rightarrow$ (17)". If there exists $\boldsymbol{K} \in \mathbb{R}^{m\times n}$ such that (20) is satisfied, then matrices $\boldsymbol{G}_s = |\boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K}|$, $s = 1,\dots,N$, satisfy $\boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K} \le \boldsymbol{G}_s$ and $-\boldsymbol{G}_s \le \boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K}$. Therefore, matrices $\boldsymbol{K} \in \mathbb{R}^{m\times n}$ and $\boldsymbol{G}_s \in \mathbb{R}^{n\times n}$, $s = 1,\dots,N$, satisfy inequalities (17).

Now we prove the converse statement "(17) $\Rightarrow$ (20)". If inequalities (17) are true for $\boldsymbol{K} \in \mathbb{R}^{m\times n}$, $\boldsymbol{G}_s \in \mathbb{R}^{n\times n}$, $s = 1,\dots,N$, then $\boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K} \le \boldsymbol{G}_s$ and $-\boldsymbol{G}_s \le \boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K}$ hold, implying $|\boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K}| \le \boldsymbol{G}_s$. Consequently we can write the inequalities $|\boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K}|^T\boldsymbol{\delta} \le (\boldsymbol{G}_s)^T\boldsymbol{\delta} \le r\boldsymbol{\delta}$, $s = 1,\dots,N$, from which we conclude that inequalities (20) have solutions.

(*ii*) Corollary 1 (*ii*) written for $\boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K}$ instead of $\boldsymbol{A}_s$ shows that inequalities (16) with $p = 2$ are equivalent to the existence of $\boldsymbol{K} \in \mathbb{R}^{m\times n}$ such that inequalities (18) are satisfied.

(*iii*) Corollary 1 (*iii*) written for $\boldsymbol{A}_s - \boldsymbol{B}_s\boldsymbol{K}$ instead of $\boldsymbol{A}_s$

shows that inequalities (16) with $p=\infty$ are equivalent to the existence of $\boldsymbol{K}\in\mathbb{R}^{m\times n}$ such that

$$|\boldsymbol{A}_s-\boldsymbol{B}_s\boldsymbol{K}|\boldsymbol{d}\le r\boldsymbol{d},\ s=1,\ldots,N. \tag{21}$$

First we prove the implication "(21) $\Rightarrow$ (19)". If inequalities (21) are true, there exists $\boldsymbol{K}\in\mathbb{R}^{m\times n}$ such that $|\boldsymbol{A}_s-\boldsymbol{B}_s\boldsymbol{K}|\boldsymbol{d}\le r\boldsymbol{d}$, $s=1,\ldots,N$. By taking $\boldsymbol{G}_s=|\boldsymbol{A}_s-\boldsymbol{B}_s\boldsymbol{K}|$, $s=1,\ldots,N$, we get $-\boldsymbol{G}_s\le\boldsymbol{A}_s-\boldsymbol{B}_s\boldsymbol{K}$ and $\boldsymbol{A}_s-\boldsymbol{B}_s\boldsymbol{K}\le\boldsymbol{G}_s$. Therefore matrices $\boldsymbol{K}\in\mathbb{R}^{m\times n}$, $\boldsymbol{G}_s\in\mathbb{R}^{n\times n}$, $s=1,\ldots,N$, satisfy inequalities (19).

Next we prove the converse statement "(19) $\Rightarrow$ (21)". If inequalities (19) are true for $\boldsymbol{K}\in\mathbb{R}^{m\times n}$, and $\boldsymbol{G}_s\in\mathbb{R}^{n\times n}$, then $-\boldsymbol{G}_s\le\boldsymbol{A}_s-\boldsymbol{B}_s\boldsymbol{K}$ and $\boldsymbol{A}_s-\boldsymbol{B}_s\boldsymbol{K}\le\boldsymbol{G}_s$ hold, which imply $|\boldsymbol{A}_s-\boldsymbol{B}_s\boldsymbol{K}|\le\boldsymbol{G}_s$, for all $s=1,\ldots,N$. Consequently we can write $|\boldsymbol{A}_s-\boldsymbol{B}_s\boldsymbol{K}|\boldsymbol{d}\le\boldsymbol{G}_s\boldsymbol{d}\le r\boldsymbol{d}$, $s=1,\ldots,N$, which proves that inequalities (21) have solutions. ■

Remark 5.

According to Corollary 2 the design of a $\mathrm{DIES}_p^{\boldsymbol{D},r}$-feedback for the switched system (9) relies on simple numerical tools. Indeed, If $p=1,\infty$, the resolution of inequalities (11) or (13) can be seen as a linear programming problem. If $p=2$, the linear matrix inequalities (12) are solved along the guidelines in [15]. ■

IV. Numerical example

In this section we illustrate the usage of the numerical tools developed for both $\mathrm{DIES}_p^{\boldsymbol{D},r}$ analysis and $\mathrm{DIES}_p^{\boldsymbol{D},r}$-feedback synthesis in the dynamics of switched systems. To this end, we consider exponentially contractive sets $X_p^{\boldsymbol{D},r}(\varepsilon;t,t_0)$ defined by (3), with shapes defined for $p\in\{1,2,\infty\}$ by the following attributes:

$$\boldsymbol{D}=\mathrm{diag}\{2,\ 2\},\ r=0.99. \tag{22}$$

A. Invariant set analysis in the discrete time dynamics

Let us consider a switched system of form (1) with $N=2$, whose matrices were randomly generated as:

$$\boldsymbol{A}_1=\begin{bmatrix}2.59 & -2.35\\ -1.23 & 2.46\end{bmatrix},\ \boldsymbol{A}_2=\begin{bmatrix}0.63 & -0.77\\ 0.26 & -0.40\end{bmatrix}. \tag{23}$$

For $p=1$, we apply Corollary 1 (*i*); thus inequalities (9) for the attributes (22) are:

$$\begin{bmatrix}2.59 & 1.23\\ 2.35 & 2.46\end{bmatrix}\cdot\begin{bmatrix}0.5\\ 0.5\end{bmatrix}\le 0.99\begin{bmatrix}0.5\\ 0.5\end{bmatrix},\ \text{for } s=1,$$
$$\begin{bmatrix}0.63 & 0.26\\ 0.77 & 0.40\end{bmatrix}\cdot\begin{bmatrix}0.5\\ 0.5\end{bmatrix}\le 0.99\begin{bmatrix}0.5\\ 0.5\end{bmatrix},\ \text{for } s=2. \tag{24}$$

Since inequalities (24) are not true, the conditions (8) in Theorem 2 are not fulfilled. Therefore the considered switched system is not $\mathrm{DIES}_1^{\mathrm{diag}\{2,2\},0.99}$, i.e. the contractive sets of form (3) for $p=1$ and the attributes in (22) are not invariant with respect to system (1)&(23).

For $p=2$, we apply Corollary 1 (*ii*) and construct the matrices $\boldsymbol{M}_s=\boldsymbol{A}_s^T\boldsymbol{P}\boldsymbol{A}_s-r^2\boldsymbol{P}$, $s=1,2$, with $\boldsymbol{P}=(\boldsymbol{D}^{-1})^2$, i.e.

$$\boldsymbol{M}_1=\begin{bmatrix}1.82 & -2.28\\ -2.28 & 2.65\end{bmatrix},\ \boldsymbol{M}_2=\begin{bmatrix}-0.12 & -0.09\\ -0.09 & -0.05\end{bmatrix}. \tag{25}$$

Because matrices (25) do not fulfill the inequalities (11) we conclude that the switched system is not $\mathrm{DIES}_2^{\mathrm{diag}\{2,2\},0.99}$.

For $p=\infty$, we apply Corollary 1 (*iii*); thus inequalities (12) for the attributes (22) can be written as follows:

$$\begin{bmatrix}2.59 & 2.35\\ 1.23 & 2.45\end{bmatrix}\cdot\begin{bmatrix}2\\ 2\end{bmatrix}\le 0.99\begin{bmatrix}2\\ 2\end{bmatrix},\ \text{for } s=1,$$
$$\begin{bmatrix}0.63 & 0.77\\ 0.26 & 0.40\end{bmatrix}\cdot\begin{bmatrix}2\\ 2\end{bmatrix}\le 0.99\begin{bmatrix}2\\ 2\end{bmatrix},\ \text{for } s=2. \tag{26}$$

Since inequalities (26) are not satisfied we conclude that the considered switched system is not $\mathrm{DIES}_\infty^{\mathrm{diag}\{2,2\},0.99}$.

B. Invariant set allocation by feedback synthesis

Let us consider a switched system of form (9) with $N=2$ and the same matrices $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ as in the previous subsection, i.e. defined by (23). The matrices $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$ were also randomly generated as:

$$\boldsymbol{B}_1=\begin{bmatrix}-2.25\\ 1.59\end{bmatrix},\ \boldsymbol{B}_2=\begin{bmatrix}-0.53\\ -0.35\end{bmatrix} \tag{27}$$

For the attributes in (22), we study the $\mathrm{DIES}_p^{\boldsymbol{D},r}$-stabilizability under arbitrary switching through state feedback for the switched system (13)&(23)&(27).

For $p=\infty$, the design of a $\mathrm{DIES}_\infty^{\mathrm{diag}\{2,2\},0.99}$ state-feedback for the switched system relies on Theorem 3 and Corollary 2 (*iii*).The resolution of the inequalities (13) was approached as a linear programming problem minimizing the const function:

$$J(\boldsymbol{G}_1,\boldsymbol{G}_2,\boldsymbol{K},\lambda)=\lambda \tag{28}$$

under the constraints:

$$\begin{aligned}&-\boldsymbol{B}_1\boldsymbol{K}-\boldsymbol{G}_1\le-\boldsymbol{A}_1, && -\boldsymbol{B}_2\boldsymbol{K}-\boldsymbol{G}_2\le-\boldsymbol{A}_2\\ &\boldsymbol{B}_1\boldsymbol{K}-\boldsymbol{G}_1\le\boldsymbol{A}_1, && \boldsymbol{B}_2\boldsymbol{K}-\boldsymbol{G}_2\le\boldsymbol{A}_2\\ &\boldsymbol{G}_1\boldsymbol{d}\le\lambda\boldsymbol{d}, && \boldsymbol{G}_2\boldsymbol{d}\le\lambda\boldsymbol{d},\ \lambda\le r\end{aligned} \tag{29}$$

The usage of the **linprog** function from the Matlab Optimization Toolbox provided the following results:

$$\lambda=0.81,\ \boldsymbol{K}=[-0.80\ \ 1.06],$$
$$\boldsymbol{G}_1=\begin{bmatrix}0.77 & 0.03\\ 0.05 & 0.76\end{bmatrix},\ \boldsymbol{G}_2=\begin{bmatrix}0.33 & 0.33\\ 0.02 & 0.78\end{bmatrix}. \tag{30}$$

This solution is also a solution for (16) with $p=\infty$ since the inequality $\lambda=0.81\le r=0.99$ implies $\boldsymbol{G}_1\boldsymbol{d}\le\lambda\boldsymbol{d}<r\boldsymbol{d}$ and

$\boldsymbol{G}_2\boldsymbol{d} \le \lambda\boldsymbol{d} < r\boldsymbol{d}$, for $\boldsymbol{d} = [2,\ 2]^T$. Therefore, the state feedback (14) constructed with matrix $\boldsymbol{K}$=[−0.80 1.06], ensures the invariance of the contractive sets defined by (3) with respect to the closed-loop switched system (15). Moreover, the designed feedback ensures the allocation in the region $\{z \in \mathbb{C} \mid |z| < \lambda\}$ of the eigenvalues for both subsystems defining the dynamics of the closed-loop switched system. As expected, we obtain $S(\boldsymbol{A}_1 - \boldsymbol{B}_1\boldsymbol{K}) = \{0.81,\ 0.72\}$ and $S(\boldsymbol{A}_2 - \boldsymbol{B}_2\boldsymbol{K}) = \{0.19,\ 0.78\}$.

For $p = 1$, the approach to the design of a $\mathrm{DIES}_1^{\mathrm{diag}\{2,2\},0.99}$ feedback for the switched system (relying on Theorem 2 and Corollary 2 (*i*)) is similar to the case when $p = \infty$. However, for the attributes in (22), the linear programming problem minimizing the cost function (28) under the constraints

$$\begin{aligned} &-\boldsymbol{B}_1\boldsymbol{K} - \boldsymbol{G}_1 \le -\boldsymbol{A}_1, -\boldsymbol{B}_2\boldsymbol{K} - \boldsymbol{G}_2 \le -\boldsymbol{A}_2 \\ &\boldsymbol{B}_1\boldsymbol{K} - \boldsymbol{G}_1 \le \boldsymbol{A}_1, \quad \boldsymbol{B}_2\boldsymbol{K} - \boldsymbol{G}_2 \le \boldsymbol{A}_2, \\ &\boldsymbol{G}_1^T\boldsymbol{\delta} \le \lambda\boldsymbol{\delta}, \quad \boldsymbol{G}_2^T\boldsymbol{\delta} \le \lambda\boldsymbol{\delta}, \ \lambda \le r, \end{aligned} \tag{31}$$

is not feasible. This shows that, under arbitrary switching, the considered switched system is not $\mathrm{DIES}_1^{\boldsymbol{D},r}$-stabilizable.

For $p = 2$, the design of a $\mathrm{DIES}_2^{\mathrm{diag}\{2,2\},0.99}$ feedback for the switched system relies on Theorem 2 and Corollary 2 (*ii*). By finding a solution to the linear matrix inequalities:

$$\begin{aligned} &(\boldsymbol{A}_1 - \boldsymbol{B}_1\boldsymbol{K})^T\boldsymbol{P}(\boldsymbol{A}_1 - \boldsymbol{B}_1\boldsymbol{K}) - r^2\boldsymbol{P} \prec = 0 \\ &(\boldsymbol{A}_2 - \boldsymbol{B}_2\boldsymbol{K})^T\boldsymbol{P}(\boldsymbol{A}_2 - \boldsymbol{B}_2\boldsymbol{K}) - r^2\boldsymbol{P} \prec = 0 \end{aligned} \tag{32}$$

we will prove that the closed-loop switched system (9) is $\mathrm{DIES}_2^{\mathrm{diag}\{2,2\},0.99}$-stabilizable. For solving this problem we used the Multi Parametric Toolbox for MATLAB [16] that is freely available. The solution to (32) is $\boldsymbol{K} = [-0.98\ \ 1.15]$. The designed feedback ensures the allocation in the region $\{z \in \mathbb{C} \mid |z| < r\}$ of the eigenvalues for both subsystems defining the dynamics of the closed-loop switched system. As expected, we obtain $S(\boldsymbol{A}_1 - \boldsymbol{B}_1\boldsymbol{K}) = \{0.8,\ 0.2\}$ and $S(\boldsymbol{A}_2 - \boldsymbol{B}_2\boldsymbol{K}) = \{0.09,\ 0.82\}$.

V. Conclussions

The existence of invariant sets represents an important property for switched systems, since such sets allow formulating different type of constraints for the state-space trajectories. In our approach, set invariance is defined as a special type of stability, called diagonally invariant exponential stability, which can be characterized by necessary and sufficient conditions.

This paper provides theoretical and numerical instruments for the use of invariant sets in analysis and synthesis. For analysis we have developed procedures that test the existence of invariant sets with given attributes. For synthesis we have proposed techniques that construct feedback laws ensuring the invariance of certain sets with pre-assigned attributes. Our theoretical results cover all shapes of sets defined by *p*-norms, with $1 \le p \le \infty$. The numerical tractability is discussed for both open- and closed-loop systems, by considering the usual *p*-norms ($p = 1, 2, \infty$). A numerical example is included for practical illustration.

VI. References

[1] D. Liberzon, *Switching in Systems and Control*, Birkhauser, Boston, 2003.

[2] Z. Sun, S. S. Ge, *Switched Linear Systems. Control and Design*, Springer-Verlag, London, 2005.

[3] R. N. Shorten, F. Wirth, O. Mason, K. Wulff, and C. King, Stability theory for switched and hybrid systems, *SIAM Review*, vol. 49, no. 4, pp. 545-592, 2007.

[4] H. Lin, and P. J. Antsaklis, Stability and Stabilizability of Switched Linear Systems: A Survey of Recent Results, *IEEE Trans. Aut. Control*, vol. 54, no. 2, pp. 308-322, 2009.

[5] H. N. Pavel, *Differential Equations: Flow Invariance and Applications*, Pitman, Boston, 1984.

[6] F. Blanchini, Set invariance in control - Survey paper. *Automatica*, vol. 35, pp. 1747-1767, 1999.

[7] F. Blanchini, and S. Miani. *Set-Theoretic Methods in Control,* Birkhäuser: Boston, Basel, Berlin, 2008.

[8] M. Voicu, Componentwise asymptotic stability of linear constant dynamical systems. *IEEE Trans. Aut. Control*, 10 (1984) 937–939.

[9] A. Hmamed, Componentwise stability of 1-D and 2-D linear discrete systems, *Automatica*, vol. **33**, pp. 1759-1762, 1997.

[10] O. Pastravanu, and M. Voicu, Generalized matrix diagonal stability and linear dynamical systems, *Linear Algebra and its Applications*, vol. **419**, pp. 299-310, 2006.

[11] F. Blanchini, and C. Savorgnan, Stabilizability of switched linear systems does not imply the existence of convex Lyapunov functions. *Automatica*, vol. **44**, pp. 1166 – 1170, 2008.

[12] M.-H. Matcovschi, and O. Pastravanu, Contractive invariant sets in the dynamics of switched linear systems, *Proc. of the 8-th Int. Conf. on Technical Informatics CONTI 2008*, Timişoara (2008), CD-ROM.

[13] A. Bhaya, and F. Mota, Equivalence of stability concepts for discrete time-varying systems, *Int. Jrnl. Robust Nonlin. Control*, vol. **4**, pp. 725-740, 1994.

[14] W. A. Coppel, *Stability and Asymptotic Behavior of Differential Equations*, D.C. Heath and Company, Boston, 1965.

[15] S. Boyd, E. Feron, L. El Ghaoui, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, SIAM, Philadelphia, 1994.

[16] M. Kvasnica, *Multi Parametric Toolbox* (MPT), http://control.ee.ethz.ch/~mpt/, ETH Zurich, 2007.

Eigenvalue ranges of interval matrices - On the practical use of a theorem estimating right outer bounds

Mihaela-Hanako Matcovschi, Octavian Pastravanu*, and Mihail Voicu

***Abstract*—This paper discusses the practical utilization of a mathematical result reported in the interval matrix literature, which provides estimations for the right-end points of the eigenvalue ranges. Our investigation is motivated by the fact that very few results of this type are available for the analysis of interval matrices; at the same time, many concrete applications cannot be addressed within the framework developed around the discussed mathematical result. We show that the main limitations are a restrictive hypothesis and a rather modest accuracy of the estimations. We also propose alternatives to this mathematical result, which rely on more convenient hypotheses and provide more precise numerical information.**

I. Introduction

The current paper considers interval matrices and interval systems defined as follows.

A family (set) of real square matrices

$$\mathcal{A}=[\boldsymbol{A}^-,\boldsymbol{A}^+]=\boldsymbol{A}^0+[-\boldsymbol{R},\boldsymbol{R}]\,,\ \boldsymbol{A}^-,\boldsymbol{A}^+,\boldsymbol{A}^0,\boldsymbol{R}\in\mathbb{R}^{n\times n}\,, \tag{1}$$

where $\boldsymbol{A}^-\le\boldsymbol{A}^+$, $\boldsymbol{R}\ge 0$, are componentwise inequalities, is called an "*interval matrix*". The notation $\mathcal{A}$ preserves this meaning throughout the paper.

A continuous-time linear system with parameter uncertainties, of the form

$$\dot{\boldsymbol{x}}(t)=\boldsymbol{A}\boldsymbol{x}(t)\,,\ \boldsymbol{A}\in\mathcal{A}\,,\ \boldsymbol{x}(t_0)=\boldsymbol{x}_0\,,\ t,t_0\in\mathbb{R}_+\,,\ t\ge t_0\,, \tag{2}$$

is called an "*interval system*", "*interval matrix system*" or "*dynamical interval system*". The usage of an interval system (2) assumes that the entries of $\boldsymbol{A}$ are fixed (not time-varying), but the knowledge of their values is limited to intervals, instead of precise numbers.

For applications, the *eigenvalue range* of the interval matrix $\mathcal{A}$ (1) presents a great interest, since it creates an algebraic portrait corresponding to the different dynamics that may be exhibited by the interval system (2). In guaranteeing the stability of the interval system (2), the crucial role is played by the *right end-point* of the *eigenvalue range* of the interval matrix $\mathcal{A}$ (1), defined as

$$I^+(\mathcal{A})=\max_{\boldsymbol{A}\in\mathcal{A}}\max_{i=1,\dots,n}\mathrm{Re}\{\lambda_i(\boldsymbol{A})\}\,, \tag{3}$$

where $\lambda_i(\boldsymbol{A})$, $i=1,\dots,n$, denote the eigenvalues of $\boldsymbol{A}$.

Taking the importance of the information offered by $I^+(\mathcal{A})$ into account, several papers such as [1] – [4], proposed techniques for computing an *estimation* of $I^+(\mathcal{A})$, denoted by $I_e^+(\mathcal{A})$, that satisfies the inequality

$$I^+(\mathcal{A})\le I_e^+(\mathcal{A})\,, \tag{4}$$

or, equivalently, $I_e^+(\mathcal{A})$ is a *right outer bound* of the eigenvalue range. Obviously, for any interval dynamic system of form (2) with $I_e^+(\mathcal{A})<0$, the positive quantity $-I_e^+(\mathcal{A})$ represents an estimation of the stability margin.

The most recent of the aforementioned papers, namely [4], proposes a procedure for the estimation of $I^+(\mathcal{A})$, recommended as an efficient computational tool, providing better results (more refined estimations) than [1] – [3]. A deeper examination of Theorem 1 in [4], that sustains the estimation principle, proves the existence of some weak points that limits its applicability (rather drastically). The current paper analyses these limitations and proposes alternative instruments, able to avoid them. Our interest for such an investigation is motivated by the fact that Theorem 1 [4] represents a nice mathematical result, almost singular in the interval matrix literature, but, at the same time, many concrete problems cannot be addressed within the framework developed by [4] and the previous researches [1] – [3].

The remainder of the text is organized in four sections playing the roles described below. Section II presents a brief overview of the estimation principle proposed by Theorem 1 [4]. Section III points out two main categories of drawbacks encountered in the exploitation of Theorem 1 [4], namely a restrictive hypothesis and a modest accuracy for the estimation of $I^+(\mathcal{A})$. Section IV proposes an optimization procedure for the numerical computation of $I^+(\mathcal{A})$, which circumvents the usage of Theorem 1 [4] and presents a larger applicability since the restrictive hypothesis of Theorem 1 [4] is avoided. Section V focuses only on some particular types of interval matrices, showing that their right end-pont $I^+(\mathcal{A})$ can be calculated directly, as an eigenvalue of a constant matrix. Section VI formulates some concluding remarks on the importance of our work.

It is worth mentioning that a similar construction can be oriented towards interval systems with discrete-time dynamics.

Manuscript received September 1, 2010.

The authors are with the Department of Automatic Control and Applied Informatics from the Technical University "Gh. Asachi" of Iasi, Blvd. Mangeron 27, 700050 Iasi, Romania (e-mail: opastrav@ac.tuiasi.ro).

II. Brief Overview of the Estimation Principle Proposed by Paper [4]

The procedure in [4] relies on the following assumption. Any matrix $\boldsymbol{A}$ belonging to the interval matrix $\mathcal{A}$ (1) has a real (simple or multiple) eigenvalue, denoted by $\lambda_{\max}(\boldsymbol{A})$, that dominates the spectrum of A, i.e.

$$\mathrm{Re}\{\lambda_i(\boldsymbol{A})\} \le \lambda_{\max}(\boldsymbol{A}),\ i=1,...,n. \tag{5}$$

Although rather conservative at a first glance, this assumption applies to important classes of interval matrices (such as essentially non-negative, or symmetrical), as well as to other cases (see, for instance, the interval matrix in Example 3 [4]).

Our presentation uses the same notations as in [4].

The procedures starts with the computation of the pair $(\lambda^0, \boldsymbol{x}^0)$, where $\lambda^0 = \lambda_{\max}(\boldsymbol{A}^0)$ is the dominant eigenvalue of $\boldsymbol{A}^0$, and $\boldsymbol{x}^0$ is its associated eigenvector, i.e. $\boldsymbol{A}^0\boldsymbol{x}^0 = \lambda^0\boldsymbol{x}^0$. The eigenvector $\boldsymbol{x}^0$ is taken normalized (scaled), such that one of its components is 1. Consider the last component normalized, i.e. $x_n^0 = 1$. It is assumed that the normalization of the last component applies to each eigenvector $\boldsymbol{x}(\boldsymbol{A}) \in \mathbb{R}^n$ associated with $\lambda_{\max}(\boldsymbol{A})$, for all $\boldsymbol{A} \in \mathcal{A}$. Introduce a generic vector $\boldsymbol{y} \in \mathbb{R}^n$, $y_i = x_i(\boldsymbol{A})$, $i=1,\ldots,n-1$, $y_n = \lambda_{\max}(\boldsymbol{A})$. Thus, for all $\boldsymbol{A} \in \mathcal{A}$, the equality

$$\boldsymbol{A}\cdot\boldsymbol{x}(\boldsymbol{A}) = \lambda_{\max}(\boldsymbol{A})\cdot\boldsymbol{x}(\boldsymbol{A}), \tag{6}$$

which defines $\lambda_{\max}(\boldsymbol{A})$ and its associated eigenvector $\boldsymbol{x}(\boldsymbol{A})$, can be written as

$$\begin{aligned} &\sum_{j=1}^{n-1} a_{ij}y_j - y_n y_i + a_{in} = 0,\ i=1,\ldots,n-1, \\ &\sum_{j=1}^{n-1} a_{nj}y_j - y_n + a_{nn} = 0, \end{aligned} \tag{7}$$

where

$$a_{ij} = a_{ij}^0 + u_{ij},\ -R_{ij} \le u_{ij} \le R_{ij},\ i,j=1,\ldots,n, \tag{8}$$

and

$$\begin{aligned} &y_i = y_i^0 + v_i,\ v_i^- \le v_i \le v_i^+,\ i=1,\ldots,n, \\ &y_1^0 = x_1^0/x_n^0,\ldots,\ y_{n-1}^0 = x_{n-1}^0/x_n^0,\ y_n^0 = \lambda^0. \end{aligned} \tag{9}$$

In (8), the values a_{ij}^0, R_{ij}, $i,j=1,\ldots,n$, are known as the entries of the constant matrices $\boldsymbol{A}^0, \boldsymbol{R} \in \mathbb{R}^{n\times n}$ used in equality (1). In (9), the values y_i^0, $i=1,\ldots,n$, are known (as explained by the second row), but the margins v_i^-, v_i^+, $i=1,\ldots,n$, are unknown.

Now, it is obvious that the knowledge of the precise value for v_n^+ would allow the computation of the right end point of the eigenvalue range as

$$I^+(\mathcal{A}) = y_n^0 + v_n^+ = \lambda^0 + v_n^+. \tag{10}$$

Paper [4] aims to find a majorant V_n^+ of v_n^+ ($V_n^+ \ge v_n^+$) such that the value $y_n^0 + V_n^+$ represents an outer estimation of $I^+(\mathcal{A})$.

Define the matrix $\boldsymbol{U} \in \mathbb{R}^{n\times n}$ and the vector $\boldsymbol{v} \in \mathbb{R}^n$, with the elements u_{ij}, $i,j=1,\ldots,n$, and v_i, $i=1,\ldots,n$. Then system (7) where a_{ij}, y_i are explicitly written by the help of (8) and, respectively (9), becomes a nonlinear system with the compact form

$$\tilde{\boldsymbol{A}}_0\boldsymbol{v} = \boldsymbol{b}(\boldsymbol{U},\boldsymbol{v}). \tag{11}$$

In (11), the elements of the vector $\boldsymbol{v} \in \mathbb{R}^n$ are variables (unknowns), the elements of matrix $\boldsymbol{U} \in \mathbb{R}^{n\times n}$ are constrained parameters

$$-R_{ij} \le u_{ij} \le R_{ij},\ i,j=1,\ldots,n, \tag{12}$$

$\tilde{\boldsymbol{A}}_0 \in \mathbb{R}^{n\times n}$ is a constant matrix, and $\boldsymbol{b}(\boldsymbol{U},\boldsymbol{v}): \mathbb{R}^{n\times n}\times\mathbb{R}^n \to \mathbb{R}^n$ is a vector valued function.

For $\tilde{\boldsymbol{A}}_0 \in \mathbb{R}^{n\times n}$ nonsingular, system (11) is equivalent to the system $\boldsymbol{v} = (\tilde{\boldsymbol{A}}_0)^{-1}\boldsymbol{b}(\boldsymbol{U},\boldsymbol{v})$. Let $\boldsymbol{r} = [r_1 \ldots r_n]^T \in \mathbb{R}^n$, $r_i > 0$, $i=1,\ldots,n$, be an arbitrary positive vector. If $-\boldsymbol{r} \le \boldsymbol{v} \le \boldsymbol{r}$, then the components of the vector $(\tilde{\boldsymbol{A}}_0)^{-1}\boldsymbol{b}(\boldsymbol{U},\boldsymbol{v}) = \boldsymbol{\Theta} = [\theta_1 \ldots \theta_n]^T \in \mathbb{R}^n$ have their absolute values majorized as below:

$$|\theta_i| \le \delta_i + \sum_{j=1}^{n-1} d_{ij}r_j + r_n\sum_{j=1}^{n-1} c_{ij}r_j,\ i=1,\ldots n. \tag{13}$$

In inequalities (13), the coefficients δ_i, d_{ij}, c_{ij}, $i,j=1,\ldots,n$, have non-negative values that are calculated from the known values a_{ij}^0, R_{ij}, $i,j=1,\ldots,n$, and y_i^0, $i=1,\ldots n$, in accordance with relations (15) – (19) in [4]. Majorizations (13) hold for any values of the parameters u_{ij} satisfying constraints (12).

Starting from (13), let us constructs the nonlinear system

$$r_i = \delta_i + \sum_{j=1}^{n-1} d_{ij}r_j + r_n\sum_{j=1}^{n-1} c_{ij}r_j,\ i=1,\ldots n, \tag{14}$$

which has the same right-hand side as (13) and r_i, $i=1,\ldots,n$, are unknowns.

If $\boldsymbol{r}\in\mathbb{R}^n$, $\boldsymbol{r}>0$, is a positive solution of system (14), then the symmetrical rectangular set $-\boldsymbol{r}\le\boldsymbol{v}\le\boldsymbol{r}$ permits the usage of *Brouwer's fixed point theorem* for the function $\boldsymbol{f}_U(\boldsymbol{v}):[-\boldsymbol{r},\boldsymbol{r}]\to[-\boldsymbol{r},\boldsymbol{r}]$, $\boldsymbol{f}_U(\boldsymbol{v})=(\tilde{\boldsymbol{A}}_0)^{-1}\boldsymbol{b}(\boldsymbol{U},\boldsymbol{v})$, for any values of the parameters u_{ij} satisfying constraints (12). Thus, all the solutions $\boldsymbol{v}\in\mathbb{R}^n$ of system (7) (equivalent to the system $\boldsymbol{v}=(\tilde{\boldsymbol{A}}_0)^{-1}\boldsymbol{b}(\boldsymbol{U},\boldsymbol{v})$) are fixed points of $\boldsymbol{f}_U(\boldsymbol{v})$. This result is formulated in [4] with no proper proof and a rather poor motivation, from which the key element of the fixed point theorem is missing.

Theorem 1 (see Theorem 1 in [4])
If the nonlinear system (14) has a positive solution $\boldsymbol{r}=[r_1\dots r_n]^T$, $r_i>0$, $i=1,\dots,n$, then the value

$$I_e^+(\mathcal{A})=y_n^0+r_n=\lambda^0+r_n \tag{15}$$

is an outer bound of the eigenvalue range of the interval matrix $\mathcal{A}$ defined by (1). ■

The main idea of Theorem 1 consists in using the component r_n of the positive solution of system (14) as a majorant for the unknown exact value v_n^+. Despite the nice mathematical innovation of connecting system (7) to system (14), the existence of positive solutions for system (14) represents a strong condition that limits the practical use of Theorem 1, as commented in the next section.

III. The Practical Use of the Estimation Principle in [1] Is Affected by Severe Limitations

Paper [4] does not place any emphasis on the applicability of Theorem 1, which is unfortunately strongly limited by two important causes. First, the hypothesis "system (14) has a positive solution" represents a severe condition, which when not met makes Theorem 1 useless. Second, the accuracy of $I_e^+(\mathcal{A})$ offered by (15) is debatable, since the majorizations (13) are rather coarse and the solutions to system (14) may significantly overestimate a decent outer bound for $\lambda_{\max}(\boldsymbol{A})$.

A. Existence of positive solutions to system (14)

Example 1
Consider the interval matrix given in Example 1 in [4]

$$\mathcal{A}=\boldsymbol{A}^0+[-\boldsymbol{R},\boldsymbol{R}],\quad \boldsymbol{A}^0=\begin{bmatrix}-3.8 & 1.6\\ 0.6 & -4.2\end{bmatrix},\ \boldsymbol{R}=\begin{bmatrix}0.17 & 0.17\\ 0.17 & 0.17\end{bmatrix}. \tag{16}$$

It is obvious that $\mathcal{A}$ is an essentially non-negative matrix and assumption (5) is fulfilled for all $\boldsymbol{A}\in\mathcal{A}$. We normalize the eigenvector $\boldsymbol{x}^0$ with respect to its first component. System (14) has the second order form

$$\begin{aligned} r_1&=0.3570+0.2380r_2+0.8000r_1r_2,\\ r_2&=0.1913+0.1275r_2+0.5000r_1r_2. \end{aligned} \tag{17}$$

The solutions to system (17) are

$$r_1=0.5941,\ r_2=0.3323; \tag{18-a}$$
$$r_1=1.2020,\ r_2=0.7044. \tag{18-b}$$

Since the eigenvector $\boldsymbol{x}^0$ was normalized with respect to its first component, Theorem 1 uses r_1 for the estimation (15). Thus, the best estimation $I_e^+(\mathcal{A})$ is obtained for the smallest value $r_1=\min\{0.5941,1.2020\}$ in the two positive solutions (18-a), (18-b), providing

$$I_e^+(\mathcal{A})=\lambda_1^0+r_1=-3+0.5941=-2.4059. \quad ■ \tag{19}$$

Example 2
Consider a slightly modified form of the interval matrix $\mathcal{A}$ (16) used by Example 1, namely

$$\mathcal{A}^*=\boldsymbol{A}^0+[-\boldsymbol{R}^*,\boldsymbol{R}^*],\quad \boldsymbol{A}^0=\begin{bmatrix}-3.8 & 1.6\\ 0.6 & -4.2\end{bmatrix},\ \boldsymbol{R}^*=\begin{bmatrix}0.19 & 0.19\\ 0.19 & 0.19\end{bmatrix}. \tag{20}$$

We intend to apply Theorem 1 to $\mathcal{A}^*$ (20) and thus we construct the second order system corresponding to the generic nonlinear system (14):

$$\begin{aligned} r_1&=0.3990+0.2660r_2+0.8000r_1r_2,\\ r_2&=0.21375+0.1425r_2+0.5000r_1r_2. \end{aligned} \tag{21}$$

System (21) has the pairs of complex solutions

$$r_1=0.8860+0.1140i,\ r_2=0.5061+0.0696i; \tag{22-a}$$
$$r_1=0.8860-0.1140i,\ r_2=0.5061-0.0696i. \tag{22-b}$$

Since system (21) has no positive solution, we cannot apply Theorem 1 and we cannot find any estimation $I_e^+(\mathcal{A}^*)$ for $\mathcal{A}^*$ (20), despite the minor differences between the two interval matrices $\mathcal{A}^*$ (20) and $\mathcal{A}$ (16). ■

This limitation of Theorem 1 is explained by the hypothesis requirement with respect to the solutions of the nonlinear system (14). Roughly speaking, the existence of positive solutions for system (14) depends on the magnitude of the coefficients in the right hand side of the system, since all these coefficients are nonnegative. At the same time, the magnitude of the coefficients is related to the entries of the nonnegative matrix $\boldsymbol{R}$ (the radius of the interval matrix $\mathcal{A}$ defined by (1)), as resulting from equalities (15) – (19) in [4]. Therefore, as a "rule of thumb" confirmed by $\mathcal{A}^*$ (20), we can say: the larger the entries of $\boldsymbol{R}$, the more unlikely the existence of positive solutions for system (14).

B. *The fixed point theorem may yield large overestimations*

The estimation $I_e^+(\mathcal{A}) = -2.4059$ calculated in Subsection 3.1 for the interval matrix $\mathcal{A}$ (16) is less accurate than the estimation $I_e^+(\mathcal{A}) = -2.5931$ obtained by [2] for the same $\mathcal{A}$ (16). Nevertheless, paper [4] (relying on some incorrect calculations) claims to have obtained better results than reported in [2].

As a general remark, when the hypothesis of Theorem 1 is fulfilled and Theorem 1 can be used, the accuracy expected from (15) for $I_e^+(\mathcal{A})$ is just moderate. One can say that the role of the positive solutions of system (14) (as used by Theorem 1) is rather qualitative than quantitative, in the sense that such a solution ensures the existence of a set where the fixed point theorem is operational. From the quantitative point of view (meaning the accuracy of the outer bounds provided by Theorem 1), the right hand side of system (14) represents a rough majorization (induced by (15) – (19) in [4]) for $|(\tilde{\boldsymbol{A}}_0)^{-1}\boldsymbol{b}(\boldsymbol{U},\boldsymbol{v})|$, which enlarges (although roughly) the actual set containing the fixed points.

IV. Nonlinear Optimization Approach as an Alternative to the Estimation Principle in [1]

The purpose is to avoid solving the nonlinear system (14) because of the severe limitations commented by the previous section. Therefore, we focus on the usage of equality (10) providing the right-end point of the eigenvalue range as $I^+(\mathcal{A}) = \lambda^0 + v_n^+$.

The positive value v_n^+ can be computed as a solution of the maximization problem for the *objective function* $J(\boldsymbol{U},\boldsymbol{v}) = v_n$, subject to the *nonlinear equality constraints* (11) and the *interval constraints* (12), starting from the *initial values* $v_i = 0$, $u_{ij} = 0$, $i,j = 1,\ldots,n$ (which satisfy the nonlinear constraints as corresponding to the center matrix $\boldsymbol{A}^0$).

We implemented this computation procedure for v_n^+ in MATLAB, by the help of the function **fmincon.** The numerical exploitation of this function is assisted by the report on the equality constraints automatically provided at the end of the optimization. A solution returned by **fmincon** is considered feasible, whenever the associate report proves the fulfillment of the nonlinear equality (11) within the accuracy limits of the numerical computation. Thus, for the interval matrix $\mathcal{A}^*$ (20) where the application of Theorem 1 failed (as commented in Example 2), we obtained the numerical values $v_n^+(\mathcal{A}^*) = 0.3959$ and $I^+(\mathcal{A}^*) = -2.6041$. We also resumed Example 1 with the interval matrix $\mathcal{A}$ (16) and we found the numerical values $v_n^+(\mathcal{A}) = 0.3544$ and $I^+(\mathcal{A}) = -2.6456$. Obviously, the calculation of the stability margin of $\mathcal{A}$ (16) yields a better result for the value $I^+(\mathcal{A}) = -2.6456$, than the estimation $I_e^+(\mathcal{A}) = -2.4059$ calculated in Example 1 via Theorem 1. The value $I^+(\mathcal{A}) = -2.6456$ is also more accurate than the estimation $I_e^+(\mathcal{A}) = -2.5931$ reported in [2]. The correctness of $I^+(\mathcal{A}) = -2.6456$ and $I^+(\mathcal{A}^*) = -2.6041$ as the precise values of the right end points for the eigenvalue ranges of $\mathcal{A}$ (16) and, respectively, $\mathcal{A}^*$ (20), will be confirmed by the help of a different procedure in the following section.

V. Estimation Principle in [1] Can Be Avoided for Some Types of Interval Systems Where $I^+(\mathcal{A})$ Is Directly Calculable

For the interval matrix $\mathcal{A}$ (1), let us introduce the *majorant matrix*, denoted by $\boldsymbol{\Omega}$,

$$\boldsymbol{\Omega} = \left[\omega_{ij}\right]_{i,j=1,\ldots,n},\ \begin{cases} \omega_{ii} = a_{ii}^+,\ i = 1,\ldots,n, \\ \omega_{ij} = \max\{|a_{ij}^-|, |a_{ij}^+|\},\ i \neq j,\ i,j = 1,\ldots,n. \end{cases} \tag{23}$$

The matrix $\boldsymbol{\Omega}$ is essentially nonnegative, i.e all its off-diagonal entries are nonnegative. Consequently $\boldsymbol{\Omega}$ has a simple or multiple real eigenvalue denoted by $\lambda_{\max}(\boldsymbol{\Omega})$ that fulfills the following inequalities (e.g. Lemma 2 in [5]).

$$\mathrm{Re}\{\lambda_i(\boldsymbol{\Omega})\} \leq \lambda_{\max}(\boldsymbol{\Omega}),\ i = 1,\ldots,n. \tag{24}$$

The objective of the current section is to show that, for some classes of interval systems, the eigenvalue $\lambda_{\max}(\boldsymbol{\Omega})$ of the constant matrix $\boldsymbol{\Omega}$ (23) provides the right end-point of the eigenvalue range corresponding to the interval matrix $\mathcal{A}$, i.e.

$$I^+(\mathcal{A}) = \lambda_{\max}(\boldsymbol{\Omega}). \tag{25}$$

Equality (25) holds true for the following types of interval systems:

a) The majorant matrix $\boldsymbol{\Omega}$ belongs (as a constant matrix) to the interval matrix, i.e. $\boldsymbol{\Omega} \in \mathcal{A}$.

(*b*) The interval matrix $\mathcal{A}$ is either lower or upper triangular for all possible values of its entries.

(*c*) There exists a vertex $\boldsymbol{V}^*$ of $\mathcal{A}$ (regarded as a polytopic matrix) that satisfies

$$\boldsymbol{V}^* = \boldsymbol{V}_D^* + \boldsymbol{V}_O^* \text{ with } \boldsymbol{V}_D^* = \mathrm{diag}\{\boldsymbol{\Omega}\},\ |\boldsymbol{V}_O^*| = \mathrm{offdiag}\{\boldsymbol{\Omega}\}, \tag{26}$$

where $\boldsymbol{V}_O^*$ is a Morishima matrix, $|\boldsymbol{V}_O^*|$ is a non-negative matrix built by taking the absolute values of the entries of $\boldsymbol{V}_O^*$, $\mathrm{diag}\{\boldsymbol{\Omega}\}$ is a diagonal matrix with the entries ω_{ii}, $i = 1,\ldots,n$, and $\mathrm{offdiag}\{\boldsymbol{\Omega}\}$ is a non-negative matrix with 0 on the diagonal and ω_{ij}, $i \neq j$, $i,j = 1,\ldots,n$, as off-diagonal

entries.

Indeed, the value $\lambda_{\max}(\boldsymbol{\Omega})$ dominates the spectrum of any matrix belonging to the interval matrix $\mathcal{A}$, i.e.

$$\forall \boldsymbol{A} \in \mathcal{A},\ \mathrm{Re}\{\lambda_i(\boldsymbol{A})\} \le \lambda_{\max}(\boldsymbol{\Omega}),\ i = 1,\ldots,n, \tag{27}$$

in accordance with Lemma 3 in [5]. Hence

$$I^+(\mathcal{A}) = \max_{\boldsymbol{A} \in \mathcal{A}} \max_{i=1,\ldots,n} \mathrm{Re}\{\lambda_i(\boldsymbol{A})\} \le \lambda_{\max}(\boldsymbol{\Omega}). \tag{28}$$

Moreover, for conditions (*a*) - (*c*), the following inequality holds:

$$\lambda_{\max}(\boldsymbol{\Omega}) \le I^+(\mathcal{A}), \tag{29}$$

as motivated below for each case.

(*a*) If $\boldsymbol{\Omega} \in \mathcal{A}$ then (29) is obvious.

(*b*) If the interval matrix $\mathcal{A}$ is triangular, then the matrix $\boldsymbol{\Omega}$ is also triangular and its diagonal elements satisfy $\omega_{ii} = a_{ii}^+$, $i = 1,\ldots,n$. Therefore $\lambda_i(\boldsymbol{\Omega}) = \lambda_i(\boldsymbol{A}^+)$, $i = 1,\ldots,n$, and $\boldsymbol{A}^+ \in \mathcal{A}$ implies (29).

(*c*) If $\boldsymbol{V}_O^*$ is a Morishima matrix, then there exists a signature matrix $\boldsymbol{S} = \mathrm{diag}\{s_1,\cdots,s_n\}$, $s_i \in \{-1, +1\}$, $i = 1,\ldots,n$, such that $\boldsymbol{S}\boldsymbol{V}_O^*\boldsymbol{S} = |\boldsymbol{V}_O^*|$. Since $\boldsymbol{S}\boldsymbol{V}_D^*\boldsymbol{S} = \boldsymbol{V}_D^*$, we can write $\boldsymbol{S}\boldsymbol{V}^*\boldsymbol{S} = \boldsymbol{S}(\boldsymbol{V}_D^* + \boldsymbol{V}_O^*)\boldsymbol{S} = \boldsymbol{V}_D^* + |\boldsymbol{V}_O^*| = \boldsymbol{\Omega}$, showing that matrices $\boldsymbol{V}^*$ and $\boldsymbol{\Omega}$ are similar and $\lambda_i(\boldsymbol{\Omega}) = \lambda_i(\boldsymbol{V}^*)$, $i = 1,...,n$. Thus, (29) is satisfied, because $\boldsymbol{V}^* \in \mathcal{A}$.

In is worth mentioning that the three classes of interval systems defined by the above conditions (*a*) - (*c*) were studied by ([6], Theorem 3.1), ([7], Corollary 2.3), ([8], Theorem 5.2). The cited papers focused on *qualitative theory* aspects, namely the equivalence between the stability of $\boldsymbol{\Omega}$ and the stability of the interval matrix $\mathcal{A}$ (1). The discussion developed in this section has placed emphasis on quantitative information, namely the precise value of the right end point of the eigenvalue range .

Both interval matrices $\mathcal{A}$ (16) and $\mathcal{A}^*$ (20) used as numerical examples in Subsection 3.1 satisfy condition (*a*). Thus we can apply equality (25). For $\mathcal{A}$ (16) we have $\boldsymbol{\Omega} = \boldsymbol{A}_0 + \boldsymbol{R} \in \mathcal{A}$ and $I^+(\mathcal{A}) = \lambda_{\max}(\boldsymbol{\Omega}) = -2.6456$, whereas for $\mathcal{A}^*$ (20) we get $\boldsymbol{\Omega}^* = \boldsymbol{A}_0 + \boldsymbol{R}^* \in \mathcal{A}^*$ and $I^+(\mathcal{A}^*) = \lambda_{\max}(\boldsymbol{\Omega}^*) = -2.6041$. These values are in full accordance with those calculated by the optimization procedure presented in the previous section.

VI. Conclusion

We have developed a thorough analysis of the limitations of Theorem 1 that represents one of the very few numerical instruments built for interval matrices in order to provide outer bounds of the eigenvalue ranges. Despite the mathematical novelty of the approach, we have shown that (i) the practical use of Theorem 1 is drastically reduced by a strong condition requested for the solutions of system (14) and (ii) the accuracy of the estimations is modest because the right hand side of system (14) gives a rough majorization of the right hand side of the system directly related to the eigenvalue problem. To avoid these drawbacks of Theorem 1, we propose two procedures that compute the right-end point of the eigenvalue range, namely (i) a method with large applicability, based on numerical optimization with constraints and (ii) a method covering some classes of interval matrices whose spectral characterization can be address by using constant matrices.

VII. References

[1] Y. Juang and C. Shao, Stability analysis of dynamic interval systems, *Int. J. Control*, vol. **49**, no. 4, pp. 1401–1408, 1989.

[2] S. S. Wang and W. G. Lin, A new approach to the stability analysis of interval systems, *Control-Theory Adv. Technol*., vol. **7**, no. 2, pp. 271–284, 1991.

[3] J. Rohn, Stability of interval matrices: The real eigenvalue case, *IEEE Trans. on Automatic Control*, vol. **37**, no. 10, pp. 1604–1605, 1992.

[4] L. Kolev, and S. Petrakieva, Assessing the stability of linear time-invariant continuous interval dynamic systems, *IEEE Trans. on Automatic Control*, vol. **50**, no. 3, pp. 393-397, 2005.

[5] O. Pastravanu, and M. Voicu, Necessary and sufficient conditions for componentwise stability of interval matrix systems. *IEEE Trans. on Automatic Control*, vol. **49**, no. 6, pp. 1016-1021, 2004.

[6] L.X. Xin, Necessary and sufficient conditions for stability of a class of interval matrices, *Int. Journal of Control*, vol. **45**, no. 1, pp. 211-214, 1987.

[7] M.E. Sezer, and D.D. Šiljak, On stability of interval matrices, *IEEE Trans. on Automatic Control*, vol. **39**, no. 2, pp. 368-371, 1994.

[8] D. Liu and A. Molchanov, Criteria for robust absolute stability of time-varying nonlinear continuous-time systems, *Automatica,* vol. **38**, no.4, pp.627-637, 2002.

Comparative study between an alternative fuel and a classical one - scenarios of using pure biodiesel (B100) or biodiesel/diesel blends (B5, B20) in the public transport network of Craiova city

Monica Mateescu, Liana Simona Sbirna, Gabriel Vladut and Sebastian Sbirna

***Abstract*—This paper aims to present the results of a recent study regarding total emission of harmful pollutants, such as CO, NO_x, PM and SO_2 registred in Craiova during 2007, 2008 and 2009. The main purpose of this study is to compare the real data gathered during these three years about the emission caused by the buses in the public transport fleet with data obtained from different scenarios, in which the classic fuel of the buses would have been replaced by an alternative one, namely biodiesel. This is a clean burning alternative fuel, produced from domestic, renewable resources. Biodiesel is the only alternative fuel to have fully completed the health effects testing requirements of the 1990 Clean Air Act Amendments, meeting ASTM D6751, and it is legally registered as a legal motor fuel for sale and distribution. It can be used in compression-ignition (diesel) engines with little or no modifications. The alternative fuel contains no petroleum, but it can be blended at any level with petroleum diesel to create a biodiesel blend. Each biodiesel/diesel blend is designated by the letter B followed by the extent of biodiesel in it.**

I. Introduction

IN the European Union, urban areas constitute now the life environment of the vast majority of the population and therefore life quality should be high. That is why it is nowadays necessary to reflect over the urban mobility problem.

All the European cities are different, but they are confronting with similar challenges and are looking for new common solution.

In the entire Europe, the intensification of the traffic in the centers of the cities led to a phenomenon of chronic agglomeration, with various negative consequences on the environment.

The re-thinking of the urban mobility supposes the optimization of the use of the entire variety of means of transport and the organization of a „co-modality" between different means of public transport (train, tramway, metro, bus) and different means of individual transport (car, motorcycle, bicycle, as well as walking).

Manuscript received June 15, 2010.

M. Mateescu is with the Faculty of Chemistry, University of Craiova, which is situated on 107I Calea Bucuresti Street, Craiova (e-mail: monica.daniela.mateescu@gmail.com).

L. S. Sbirna is with the Faculty of Chemistry, University of Craiova, 107I Calea Bucuresti Street, Craiova (e-mail: simona.sbirna@gmail.com).

G. Vladut is with the S. C. IPA CIFATT Craiova company, 12 Stefan cel Mare Street, Craiova. He is an IEEE Member (e-mail: office@ipacv.ro).

S. Sbirna is with Dolj Environment Protection Agency, 1 Petru Rares Street, Craiova (e-mail: s.sbirna@gmail.com, phone: (+40).721.329779).

II. Importance of Introducing Ecological Driving

IN the context of the sustainable development, the urban areas are confronting with a huge challenge: that to reconcile the economic development of the cities and the accessibility, on one hand, with the amelioration of the quality of life and with the environmental protection, on the other hand.

When confronting with these problems that have multiple and various implications, a common effort will allow the encouragement of seeking innovative and ambitious solutions concerning the urban transport in order to get in the situation in which cities are less polluted and more accessible, and the traffic more fluid.

We must find together means of attaining a better urban and surrounding mobility, a sustainable mobility and mobility in the benefit of all the citizens that should allow the economic operators to play their part in our cities.

As solutions to the existent environmental and pollution problems one might consider the expansion, rehabilitation and the modernization of the public transport, i.e. of the trolleybuses, buses, tramways, metros and suburban railways, as well as other sustainable urban transport projects that should be promoted and sustained by the European Union.

Certain authorities have improved the ecologic performance of their auto park for the public transport acquiring clean vehicles and offering economic stimulants to the common transport operators. The public financial sustaining for the new alternative fuel distribution infrastructure was also decisive in several cities. The common acquisitions of clean vehicles and with reduced energy and fuel consumption by the public authorities might accelerate the creation of a new market for technologies and would assure their economic viability.

The ecological driving is also an important factor that reduces the energy and fuel consumption by a changing in the habits of the drivers that should be encouraged especially by the auto schools when forming professional drivers. The urban public transport must attain and attract and also maintain personnel with high qualification.

The formation programs can improve the competences of the personnel in public transport and can reduce the emissions and consequently pollution.

One of the most reliable alternative fuels that can be used in the public transport in order to introduce ecological driving is biodiesel, used as such or as blends with classical diesel.

III. Biodiesel as an Alternate Fuel

BIODIESEL is the only alternative fuel to have fully completed the health effects testing requirements of the 1990 Clean Air Act Amendments, meeting ASTM D6751, and it is legally registered as a legal motor fuel for sale and distribution.

Biodiesel can be used as a pure fuel or blended with petroleum in any percent [1].

Each biodiesel/diesel blend is designated by the letter B followed by the extent of biodiesel in it (e.g. B10, B25 etc).

IV. Biodiesel Blends Involved in the Present Study

FOR the scenarios we imagined, we have dealt with B5, B20 and B100 (pure biodiesel). Let us present each of them.

B5 (a blend of 5 percent by volume biodiesel with 95 percent by volume petroleum diesel) is the blend which maximize the engine life and does not require any engine adjustment.

B20 (a blend of 20 percent by volume biodiesel with 80 percent by volume petroleum diesel) has demonstrated the best environmental benefits with a minimum increase in cost for fleet operations and other consumers.

Finally, B100 presents the advantages of being nontoxic, very easy biodegradable and significantly reducing toxic emissions – among others, it takes to level zero the sulfur dioxide and the aromatics – but it seems to be less advantageous than B20 taking into account the costs involved by a major engine modification that would be required in this case [2].

The scenarios of using these types of biodiesel will be concretely applied on the public transport network in Craiova, one of the most developed cities of the Romania, situated at 44°20′00″ North latitude and 23°49′00″ East longitude.

Before applying these scenarios, let us reveal its actual situation regarding public transport.

V. Present Situation of Public Transport in Craiova

BEFORE starting this study, the local public transport network in Craiova city had a park of approximately 200 auto-vehicles that were a great source of environmental pollution, and a great number of them did not offer a degree of comfort and safety necessary to passengers. At least 100 buses had more than 10 years when the Environmental Protection Agency imposed their removal from circulation in the following years.

Due to practical reasons of material type, an important percent from the public transport fleet is composed of vehicles found in a bad condition, either because of long exploitation, or because they were bought second-hand.

This fact leads progressively to the degradation of the quality and image of the urban public transport, proving that is imperatively necessary to reform the system of public transport, by acquiring vehicles with an increased capacity and more efficient from the point of view of fuel consumption and transport quality.

VI. Emissions Caused by the Buses in Craiova

THE main purpose of this study is to compare the real data gathered between 2007 and 2009 about the emissions caused by the buses in the public transport fleet of Craiova with data obtained from different scenarios, in which the classic fuel of the buses would have been replaced by an alternative one, namely biodiesel.

The air quality in Craiova urban area (including its surrounding suburban zone) is continuously monitored.

The concentration of different air pollutants is recorded by five modern monitoring station provided by the European Community (one of them is shown in Figure 1). Their symbols are DJ1, DJ2, …, DJ5, DJ being the abbreviation for Dolj county, whose main city is Craiova.

Whereas DJ3 measures the pollutants' concentration in a high-traffic zone (it is placed on Calea Severinului near Billa hypermarket), DJ2 measures the urban background (it is positioned next to the City Hall) and DJ5 measures the suburban background (it is located in the village Breasta). The other ones - DJ1 (Calea Bucuresti) and DJ4 (Isalnita) - are industrial type stations.

As noticed above, this paper aims to present the results of a recent study regarding total emission of CO, NO_x, PM and SO_2 registred in Craiova during 2007, 2008 and 2009, a special atention being payed to the apport of the buses to it.

Concrete results are given in Table I and represented in Fig. 1 („Mg" meaning „megagrams").

Table I. Total emissions for four main air pollutants

Air pollutant	Total emission (Mg)		
	2007	2008	2009
CO	32313	35688	22492
PM	75	80	175
NO_x	1824	1970	2242
SO_2	304	330	500

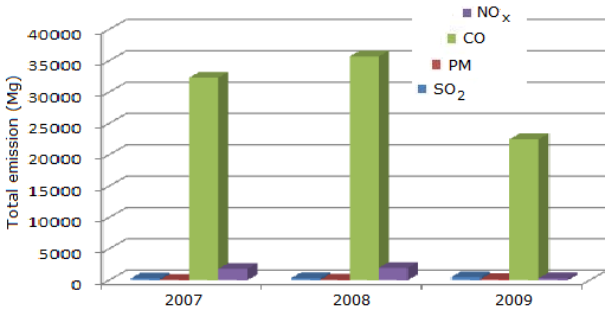


Fig. 1. Comparison between the amounts of CO, NO_x, PM and SO_2 emitted in 2007, 2008 and 2009

For each pollutant a detailed study may be performed, taking into account the origin of the emission.

Let us present, as an appropriate example, the detailed study for carbon monoxide (CO).

The main part of the CO emission is originated from transport. Continous air quality measurements, together with classical means (the Corinair method, for instance) have been used by the experts of the Dolj Environment Protection Agency to establish which part of the total CO emission is originated from transport.

Moreover, the experts of the Dolj Environment Protection Agency have processed input data provided by the local transport authorities, in order to establish which part of these is caused by the buses.

The results may be suggestively shown as in Fig. 2 (detailed for the year 2009) and Fig. 3 (comparatively presenting the situation during the three years taken into study).

In the figures bellow, „Gg" means „gigagrams".

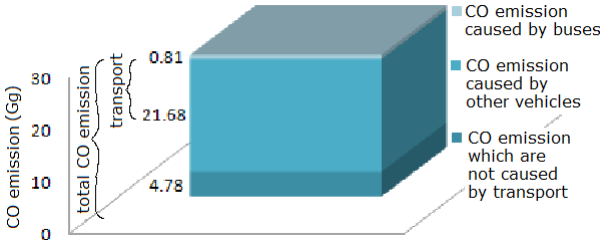


Fig. 2. The origin of the CO emission

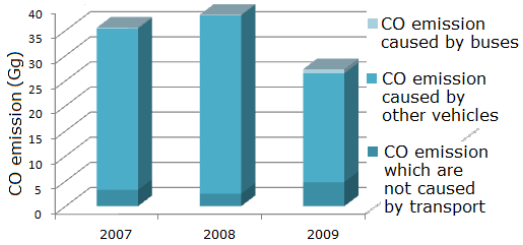


Fig. 3. Comparison between the amounts of CO emitted in 2007, 2008 and 2009, also taking into account their origin

It shows that a major part of the total CO emission is originated in the transport activities.

VII. Scenarios of Using Biodiesel in Public Transport

As mentioned above, in order to compare the emission caused by diesel with the ones caused by biodiesel, three scenarios have been imagined and investigated:

a) the buses would have been operated on B5;

b) the buses would have been operated on B20;

c) the buses would have been operated on B100.

The way emissions would have changed (decreased or increased) by using B5, B20 or B100 instead of classical fuel is expressed in Table II and then represented in Fig. 4 as a changing trend (expressed in percents).

Table II. Percentage by which emissions change in the three scenarios

Air pollutant	Emission change obtained by replacing diesel with biodiesel (%)		
	B5	B20	B100
CO	-4,5	-12	-48
PM	-4	-12	-47
NO_x	+0,5	+2	+10
SO_2	-5	-20	-100

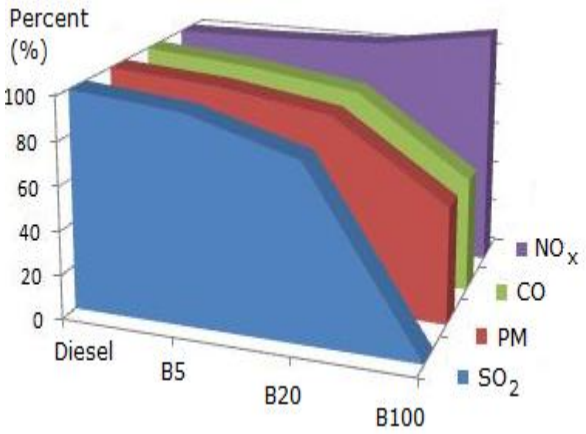


Fig. 4. Changing trend for each air pollutant (percents)

More specifically, taking into account the real values, exhibited during the period we are dealing with, for each of these pollutants, the study leads to interesting results, that will be presented in what follows.

The values of carbon monoxide (CO) emission would have been respectively decreased at 95.5%, 88% and 52% from the real value, which might be shown in Table III and even more suggestive in Fig. 5.

Table III. Results obtained for CO emission for each scenario

Fuel	Total CO emission (Mg)		
	2007	2008	2009
diesel	32313	35688	22492
B5	30859	34082	21480
B20	28435	31405	19793
B100	16803	18558	11696

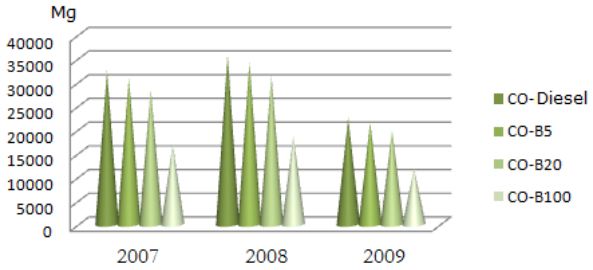


Fig. 5. Total CO emission, hipotetically using all four fuels

Similarly, the values of particulate matter (PM) emission would have been decreased at 96%, 88% and 53% respectively, which is presented in Table IV and then shown in Fig. 6.

Table IV. Results obtained for PM emission for each scenario

Fuel	Total PM emission (Mg)		
	2007	2008	2009
diesel	75	80	175
B5	72	76,8	168
B20	66	70,4	154
B100	39,8	42,4	92,8

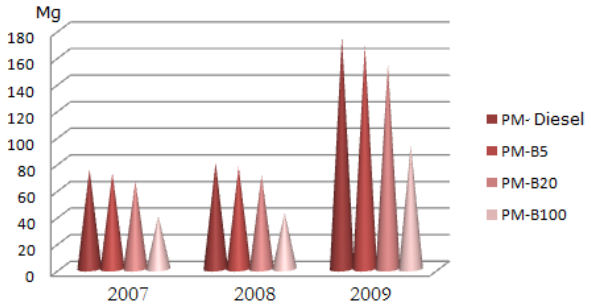


Fig. 6. Total PM emission, hipotetically using all four fuels

As far as sulfur dioxide (SO_2) emission is concerned, the values would have been decreased at 95%, 80% and finally at 0% from the real value, as resulting from Table V and Fig. 7. It shows that pure biodiesel would have totally eliminated this pollutant, which is important to note.

Table V. Results obtained for SO_2 emission for each scenario

Fuel	Total SO_2 emission (Mg)		
	2007	2008	2009
diesel	304	330	500
B5	288,8	313,5	475
B20	243,2	264	400
B100	0	0	0

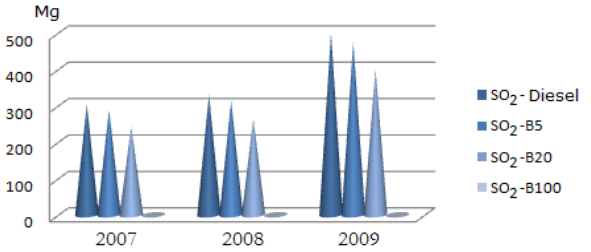


Fig. 7. Total SO_2 emission, hipotetically using all four fuels

A problem still remains, namely the increasing values of nitrogen oxides (NO_x) emission at 100.5%, 102% and 110% from the real value. This fact obviously shows by consulting Table VI and regarding Fig. 8.

Table VI. Results obtained for NO_x emission for each scenario

Fuel	Total NO_x emission (Mg)		
	2007	2008	2009
diesel	1824	1970	2242
B5	1833	1979,5	2253,2
B20	1860,5	2009,4	2287
B100	2010,8	2167	2466,2

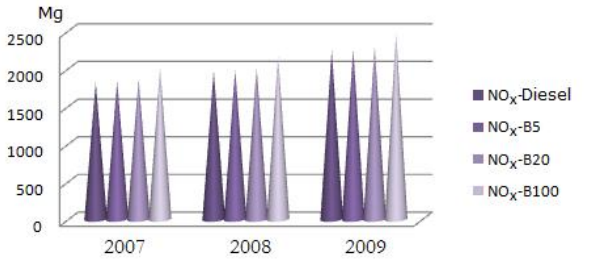


Fig. 8. Total NO_x emission, hipotetically using all four fuels

However, this disadvantage seems to be less important than the advantages presented before it.

Beside the advantages presented above, introducing biodiesel in public transport fleet would lead to a significant decrease of the obscurity level, as it might be seen from Fig. 9 [1].

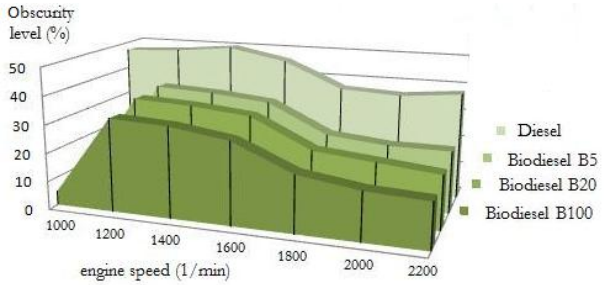


Fig. 9. The way the obscurity level changes with the engine speed

More specifically, this parameter roughly changes at a very low engine speed (lower than 1000 min^{-1}), as well as at a high engine speed (higher than 1800 m^{-1}).

If a vehicle uses traditional diesel, the vehicle emits black, stinky smoke; with biodiesel, it really becomes clean.

VIII. Conclusion

BIODIESEL can be considered as a new technology, that is less harmful to the environment, producing fewer emissions [3].

1. Moreover, another argument in favor of trying to introduce biodiesel in the engines of the public transport vehicles is that biodiesel can be made from domestically produced, renewable oilseed crops [4].
2. If these crops are grown without the use of petroleum byproducts like artificial fertilizers, they do not contribute to any use of fossil fuels. When biodiesel is derived from reused vegetable oil or animal fat, it does not contribute to fossil fuel use and it utilizes waste products as well, whereas petroleum based fuels (fossil fuels), are not renewable [5].

The study presented in this paper – showing that a significant part of pollutant emissions originate from the urban transport – suggests that using biodiesel as an alternative fuel, either pure or in different blends with diesel (the classical fuel) is an option to seriously think about.

IX. References

[1] M. Mateescu, *Combustibili auto si poluarea mediului. Prezent si viitor*. Craiova: Ed. Universitaria, 2010, ch. 4–7.

[2] H. Gaugitsch, "A differentiated assessment of the future of Biotechnology – a perspective from an EU member state," *Environ. Sci. Pollut. Res.*, vol. 11, no. 3, pp. 141–142, 2004.

[3] M. Johnston and T. Holloway, "A global comparison of national biodiesel production potentials," *Environ Sci. Technol.*, vol. 41, no. 23, pp. 7967–7973, 2007.

[4] H. Halleux, S. Lassaux, R. Renzoni and A. Germain "Comparative life cycle assessment of two biofuels, ethanol from sugar beet and rapeseed methyl ester," *Int. J. Life Cycle Assess.*, vol. 13, no.5, pp. 184–190, 2008.

[5] J. Kloverpris, H. Wenzel, M. Banse, I. Mila and A. Reenberg, "Modelling global land use implications in the environmental assessment of biofuels," *Int. J. Life Cycle Assess.*, vol. 13, no. 3, pp. 178–182, 2008.

Fault Detection and Isolation Using Recurrent Wavelet Neural Networks

Letitia Mirea

Abstract— **The paper proposes a new type of recurrent wavelet neural network to be applied to detect and isolate faulty behaviours of a dynamic process. Hybrid learning based on c-means fuzzy clustering algorithm and the steepest-descent method is used to train the proposed neural network. The experimental case study refers to the sensor and actuator fault diagnosis of a sub-system from the evaporation station of a sugar factory, namely the evaporator. A generalised neural network based observer scheme is used to generate the residuals (symptoms) in the form of one step-ahead prediction errors. These are then processed by a neural classifier in order to take the appropriate decision regarding the type of the behaviour of the process (normal or abnormal).**

I. Introduction

Fault diagnosis has become an important field of automatic control in the last decade, with a goal focused on the detection and isolation of process faults [1, 2]. Usually, a fault manifests as a deviation of at least one characteristic property or variable of the process with respect to the corresponding nominal value. A fault diagnosis system should satisfy some basic requirements such as: early detection of small faults with abrupt or incipient time behaviour, diagnosis of faults in different parts of the investigated (supervised) process, detection of faults in closed-loop and supervision of the process in the transient states. All these requirements should be satisfied in the face of the existing measurement uncertainty, disturbances and incomplete knowledge about the process. A fault diagnosis system is efficient if it is able to avoid interpreting the unknown inputs (disturbances, measurement noise, modelling errors) as faults, i.e. to produce a reduced number of false alarms. Of particular importance is the diagnosis of incipient faults that leads to the avoidance of dangerous operating conditions, maintenance as required, long distance diagnosis, increase of availability and productivity, and automatic quality assurance [3].

The current trends of research in fault diagnosis are focused on the development of *robust methods*. The robustness of a fault diagnosis system is achieved if it is able to maximise of the detectability and isolability of faults together with the minimisation of the effects of the uncertainties and disturbances on the fault diagnosis procedure. An *active robustness* is achieved by generating symptoms that are not influenced by the unknown inputs of the process. A *passive robustness* can be achieved in the stage of symptom evaluation by using robust decision-making procedures with respect to uncertainties and disturbances [3].

Artificial Neural Networks (ANNs) are widely used in Fault Detection and Isolation (FDI) as technique based on process data able to cope with the robustness problem [4]. They are currently used in FDI as both dynamic non-linear models to generate the symptoms and pattern classifiers to evaluate them. These approaches do not require an accurate mathematical model of the process, but need representative training data.

Wavelet decompositions provide a useful basis for localised approximation of functions with any degree of regularity at different scales and with a desired accuracy. Wavelets can therefore be seen as a new basis for approximating functions and can be used for non-linear system identification [5]. This approach is based on the concept of approximating an arbitrary non-linear function in terms of dilates and translates of a single function, namely the 'mother' wavelet function [6]. Wavelet Neural Networks (WNNs) were introduced [6], [7] as advanced computational schemes for signal representation and classification. They were designed to combine the results of the wavelet decomposition theory with the learning properties characterising the neural networks. WNNs have been successfully applied to the identification of non-linear systems and adaptive control, [8], [5] due to the time-frequency localisation properties of wavelets and the learning ability of the neural networks.

Only a few contributions have dealt with the application of wavelet neural networks to process monitoring and fault detection [9], [10]. In those investigations, different wavelet techniques (discrete wavelet transform, wavelet packet decomposition) were applied to extract significant information, i.e. features, characterising the signals of the monitored system for normal and faulty behaviours. The evaluation (classification) of the resulted features was then performed using threshold-based techniques or a neural classifier.

This paper refers to the development of a new type of WNN as a further development of the previously reported WNN structure in [11]. This new WNN, namely the Recurrent Wavelet Neural Network (RWNN), includes in

Manuscript received August 31, 2010.

L. Mirea is with the "Gh. Asachi" Technical University of Iasi, Department of Automatic Control and Industrial Informatics, Blvd. D. Mangeron 53A, RO-700050 Iasi, Romania (phone: +40-232-278680, fax: +40-232-214290, e-mail: lmirea@ac.tuiasi.ro).

This work was supported by The National Centre for Programs Management from Romania under the research grant SICONA - 12100/2008.

its structure internal Auto-Regressive Moving-Average (ARMA) filters instead of constant weights. Its application to sensor and actuator fault diagnosis of a sub-system from the evaporation station of a sugar factory is also considered. Process input-output data are used to train the RWNN, i.e. to determine the network parameters that would minimize a performance index. Firstly, the c-means fuzzy clustering algorithm is used to determine the number of wavelet nodes and the initial values for the network parameters. Following this, a gradient-based learning algorithm is applied in to refine the RWNN parameters.

The paper is organised in five sections as follows. Section II presents the architecture and the learning procedure for RWNN. Section III refers to the design of an FDI system based on RWNNs (residual generation) and Multi-Layer Perceptron (residual evaluation). The application of the developed RWNNs to the problem of detecting and isolation sensor and actuator faults in the evaporation station of a sugar factory is presented in Section IV. Conclusions are given in Section V.

II. WAVELET NEURAL NETWORKS

The main result of the theory of wavelet decomposition [12] refers to the fact that any function $f(x) \in L^2(\Re)$ can be approximated by:

$$f(x) \cong \sum_{k=1}^{K} c_k \cdot \psi\left(\frac{x - b_k}{a_k}\right) \tag{1}$$

where the function ψ is the "mother wavelet", c_k are the weighting coefficients, $b_k \in \Re$ are the translation parameters and $a_k \in \Re_+^*$ are the dilation parameters.

Relation (1) can be implemented via a feed-forward neural network structure. This wavelet neural network has one hidden layer of neurons with the transfer functions obtained as dilated and translated versions of the "mother wavelet". This function should have zero-mean and should be localised in both space and frequency domains. In the present approach, the following "Mexican hat" is considered as the "mother wavelet":

$$\psi(x) = \alpha \cdot (1 - \alpha \cdot x^2) \cdot e^{-\frac{\alpha \cdot x^2}{2}}. \tag{2}$$

The constant α in relation (2) is set to the value 0.5.

In the case of a Single-Input Single-Output (SISO) WNN, the following relation gives the output of the net:

$$y = c_0 + \sum_{j=1}^{M} c_j \cdot \psi\left(\frac{u - b_j}{a_j}\right), \tag{3}$$

where u is the WNN input and M is the number of wavelet nodes. The wavelet neuron itself has no capacity to implicitly identify offsets within measured data. Thus, the explicit estimation of a mean value possibly existing in the process-measured data becomes substantial, motivating the introduction of the constant term: c_0, in eq. (3).

For a problem with P inputs, multidimensional wavelets must be considered. The most frequent choice [7], [8], [9] is that of separable wavelets, i.e. the product of P mono-dimensional wavelets associated with each input:

$$\Psi_j(\mathbf{u}) = \prod_{i=1}^{P} \psi\left(\frac{u_i - b_{ij}}{a_{ij}}\right), \quad j = 1, \ldots, M. \tag{4}$$

where $\mathbf{u} = [u_i]_{i=1,\ldots,P}$ is the vector of WNN inputs, $\mathbf{b}_j = [b_{ij}]_{i=1,\ldots,P}$ and $\mathbf{a}_j = [a_{ij}]_{i=1,\ldots,P}$, j = 1, ..., M are the translation and dilation vectors, respectively. Fig. 1 presents the architecture of a multi-input single-output WNN. In this case, the WNN performs static mapping between its input space and its output space. The WNN output is given by:

$$\hat{y} = c_0 + \sum_{j=1}^{M} c_j \cdot \Psi_j(\mathbf{u}). \tag{5}$$

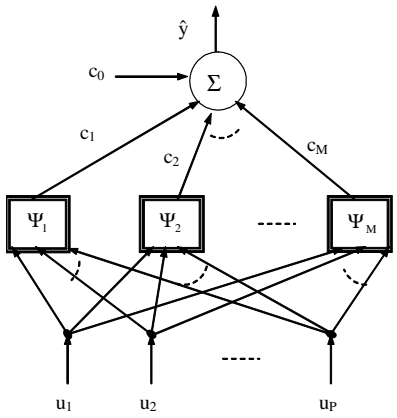


Fig. 1. Architecture of the static WNN

In the case of monitoring a dynamic system using sampled data, a discrete time process representation is required. The purpose is to identify WNN models for each system output, i.e. Multi-Input Single-Output (MISO) models. To be used for dynamic system identification, the wavelet neural networks require spatial time representation.

One way of achieving this is to use time delay units. In this case, the static WNN has to be fed with current and delayed values of the inputs and the outputs of the system [13]. This leads to an increase of the dimension of the neural network input space. Another possibility is to include into the WNN architecture dynamic elements or recurrent connections. In this case, the wavelet neural network with internal dynamic elements has to be fed only with current values of the process inputs and outputs. In this way, the dimension of the WNN input space is lower than in the previous approach. This further implies a reduction in the WNN's structure complexity.

The paper introduces a new type of WNN that includes local dynamic elements in the form of ARMA filters to substitute the constant weights, i.e. a recurrent wavelet neural network. The proposed RWNN has the architecture depicted in Fig. 2. Here q^{-1} stands for the delay operator. The output of the RWNN is given by:

$$\hat{y}[k] = c_0 + \sum_{i=1}^{M} \tilde{z}_i[k],$$

$$\tilde{z}_i[k] = \sum_{j=1}^{n_C} c_j \cdot z_i[k-j] - \sum_{\ell=1}^{n_D} d_\ell \cdot \tilde{z}_i[k-\ell], \quad z_i[k] = \Psi(\mathbf{u}[k])$$

where [k] is the sample time instant. Due to the use of general delay orders, n_C and n_D, the suggested RWNN allow for a flexible model design for non-linear systems of different orders. The parameters n_C and n_D are determined using a trial-and-error procedure in the training stage of the RWNN.

One considers N data pairs collected from the inputs and outputs of the process. It is desired that the output of the RWNN, $\hat{y}[k]$, would fit the system output, $y_P[k]$, where [k] is the sample time instant:

$$\hat{y}[k] = f(u_P[k], y_P[k-1]) \cong y_P[k], \tag{6}$$

where u_P denotes the process input, $y_P[k]$ represents the process output, $\hat{y}[k]$ denotes the approximating output given by the RWNN and *f* is the function that denotes the mapping performed by the neural net.

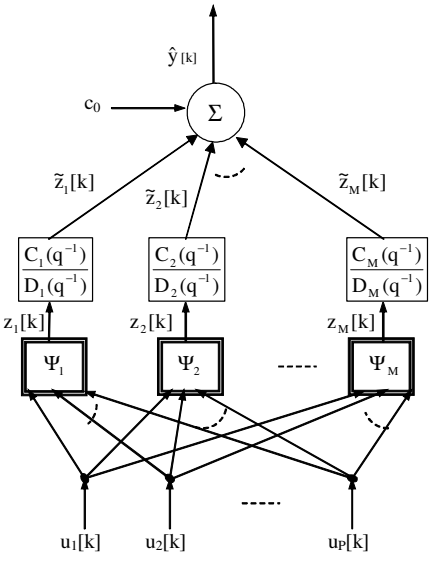


Fig. 2. Architecture of the RWNN

The RWNN network parameters $\{\mathbf{a}_m, \mathbf{b}_m, c_0, c_i, d_j\}$, m=1,..., M, i=1,…,$n_C$, j=1,…,$n_D$ are adapted in order to minimise a quadratic cost-function such as the sum-squared error between the network output $\hat{y}[k, \boldsymbol{\theta}]$, and the process output, $y_P[k]$:

$$SSE = \frac{1}{2} \cdot \sum_{k=1}^{N} (y_P[k] - \hat{y}[k, \boldsymbol{\theta}])^2 , \tag{7}$$

where $\boldsymbol{\theta}$ represents the vector of RWNN parameters.

As the network output is non-linear in its parameters, a non-linear optimisation technique must be applied. The network parameters are thus optimised iteratively based on the first-order steepest-descent method:

$$\xi^{new} = \xi^{old} - \eta \frac{\partial SSE}{\partial \xi} \tag{8}$$

where ξ is an arbitrary network parameter and η is the learning rate. The strategy of adaptive learning rate with momentum term is also used in order to allow for a stable learning.

The problem that arises here is how the RWNN parameters should be initialised in order to avoid the convergence of the gradient procedure to an undesirable local minimum and to insure the numerical stability of the learning algorithm. A random initialisation of all parameters to small values is not suitable, as the components of the gradient of the cost function may be very small in the area of interest. Thus, one wants to take advantage of the input space where wavelets are not zero [8]. Therefore, the RWNN parameters are initialised based on the c-means fuzzy clustering algorithm [14], [15].

The purpose of the C-means Fuzzy Clustering (CFC) algorithm is to distil natural groupings of the RWNN input data set, producing a concise representation of the system behaviour. Finally, a number of cluster centres are obtained to be used as translation parameters for the wavelet nodes, $\mathbf{b}_k = [b_{p,k}]_{p=\overline{1,P}}$, $k = \overline{1, M}$. For each data point from the set of N data points used in the CFC, a degree of membership μ_{ki} to each obtained cluster k, $k = \overline{1, M}$ is computed. Using the obtained membership degrees, the standard deviations of each cluster are obtained using the relation:

$$a_{p,k} = \frac{\sum_{i=1}^{N} (\mu_{ki})^2 \cdot (u_p(i) - c_{p,k})^2}{\sum_{i=1}^{N} (\mu_{ki})^2},$$

$$\mathbf{a}_k = [a_{p,k}]_{p=\overline{1,P}}, \quad k = \overline{1, M}.$$

These values are used as dilation parameters for the wavelet nodes. The initial values for the internal ARMA filters are set to small, positive, random values.

III. FDI System based on RWNN

A. Residual generation

For the generation of symptoms, the RWNN is used to model the considered process. For each output, a Multi-Input Single-Output (MISO) neural model is identified. Since the control system tends to hide the faults, both

process inputs and outputs are used as inputs to the RWNNs.

The developed RWNN models are further used in an observer scheme [16] to generate sets of residuals in the form of the one step-ahead prediction errors. The Extended Neural Generalised Observer Scheme (E-NGOS) is considered.

The extended Neural Generalised Observer Scheme (E-NGOS). For the present application, one considers a MISO process with I inputs $u_i[k], i = 1,...,I$ and the output $y_P[k]$, all known at sampling time k. The considered process is firstly identified by one RWNN. This is driven by all inputs and the output of the system. The network estimates the output of the process:

$$\hat{y}_0[k] = f_0(\mathbf{u}[k], y_P[k-1]), \mathbf{u}[k] := [u_i[k]]_{i=1,...,I} \quad (9)$$

Secondly, the Neural Generalised Observer Scheme (NGOS) is developed. It consists of as many RWNNs as process inputs are available. Each neural network of the NGOS is driven by the process output and all inputs except one input. The output of the MISO process is approximated by one RWNN model of the NGOS as:

$$\begin{aligned} &\hat{y}_j[k] = f_j(\mathbf{u}_j[k], y_P[k-1]), \\ &\mathbf{u}_j[k] := [u_i[k]]_{,i=1,...,I;i\neq j}\,;\ j = 1,...,I \end{aligned} \quad (10)$$

The resulting bank of neural models approximates all outputs of the process. The training of the RWNNs is based on the system data corresponding to its normal behaviour. The residuals are then generated by means of the *extended NGOS* given by eqs. (9), (10). One step-ahead prediction errors are obtained, i.e. the serial-parallel use of the RWNN models:

$$\varepsilon_j[k] = y_P[k] - \hat{y}_j[k]; \quad j = 0,...,I. \quad (11)$$

The residual ε_0 is affected by all process variables. Instead, each residual $\varepsilon_j, j = 1,...,I$ is sensitive to the process output and all inputs but the j^{-th} input. In case of a faulty input variable, the decoupled residual remains small, while the other are influenced. This is valid if all used process inputs are uncorrelated. If some of those variables are correlated, more residuals are not affected by a certain fault. In this case, the residual ε_0, based on Eq. (11), is important by reflecting all kinds of faults. These patterns that reflect all the changes in the process functioning are used to detect and locate the faults.

B. Residual evaluation

Residual evaluation means to match each pattern of the residual vector with one of the pre-assigned classes of faulty behaviour, if available, and the fault-free case, respectively.

The main problem in the residual evaluation stage is the uncertainty in classification of the patterns that may arise from the overlapping nature of various classes. Therefore, a robust decision can be achieved by using a neural network as pattern classifier [16]. The static Multi-Layer Perceptron with sigmoid neurons is considered in the present approach.

The neural classifier maps the patterns (11) from the residual space into a decision space, such as the patterns belonging to a class of process behaviour (normal or abnormal) to cluster around pre-selected points, optimally chosen in the decision space [15]. A class of process behaviour is detected and isolated if an unknown input pattern is mapped closest to one of the target vectors. That multi-dimensional point corresponds to a certain class that reflects one of the considered process behaviours (normal or faulty).

Usually, the target vectors of decision space $\{\mathbf{t}_c\}, c = 1,\ldots,C$ are chosen to be the set of vertices of C-unit (Euclidean) vectors. Here C stands for the number of the considered classes of process behaviour. An optimised choice of the target vectors is suggested in [13] as a subset of C vectors with K binary-valued components in the set $\{0.1, 0.9\}$, among the vertices of a hypercube. The number of the components is $K = 1 + [\log_2(C)]$, where [.] denotes the function of greatest integer.

A fault is only detected if the input pattern is mapped far from all learned classes. This situation corresponds to new (faulty) process behaviour and only the synthesis of the classifier must be reconsidered for further fault diagnosis. In the present approach, the decision logic is based on the Euclidean distance between the actual output and the target vectors of the classifier.

IV. APPLICATION

A. Process Description

The methodology presented is assessed by using real process data from the Lublin sugar factory in Poland. The example is based on a benchmark study initiated in the EC DAMADICS Research Training Network [17]. The Evaporation Station (ES) (Fig. 3) represents the subject of the investigation.

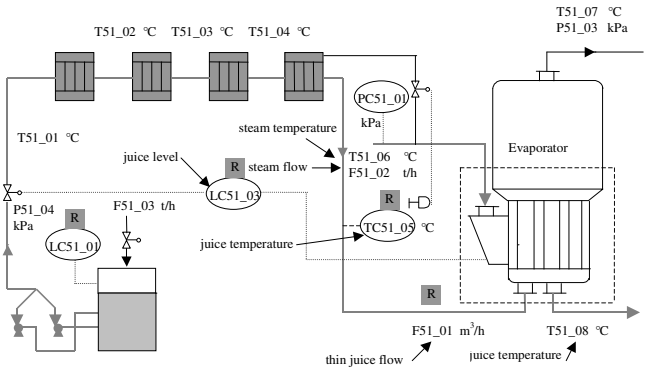


Fig. 3. Heater and first section of evaporation station of Lublin sugar factory.

The process is used to reduce the water content of sucrose juice. The liquid goes through a series of five sections. In each passage, the sucrose concentration increases. The juice steam recovered from one stage is used as a heating source for the next section. The sugar evaporation control should be performed in such a way that the used energy is optimised and the required quality of the final product is achieved [17]. There have been a number of studies using this benchmark system, mostly using neural network or neuron-fuzzy schemes, as one of the benchmark challenges has been to use model-free design methods [18].

The factory is equipped with a SCADA system that allows for the registration and storage of all set-point, control (CV) and process (PV) variables. The faults are rather exceptional, due to the technological improvement and careful inspection of installation before starting the three-month long production period. The fault-free data stored during one month, every 10 s, have been used to develop the residual generators. This data base contains over 200.000 rows of measurements, with over 100 variables in each row.

A number of 3000 rows of measurements from another month of plant exploitation have been used to test the quality of identified models. Another testing set has been created by introducing artificial single faults in the first testing data set. Faults are generated by adding or subtracting 5, 10 and 15 percent of the whole possible range of variation to each measurement, respectively, in different periods of time. The latter data set is used to both validate the relationships between process variables and design appropriate mechanisms of residual evaluation.

The investigated process is decomposed in five MISO sub-processes [16]. In this paper, only one sub-process has been considered, namely the evaporator. A diagnosing unit has been developed to detect and isolate actuator and sensor faults.

Evaporator. The inputs of the model (9) are the steam flow to the input of the ES, the steam temperature at the input of ES and the juice temperature (PV) after heater; the output is the juice temperature after section 1 of ES; artificial faults affect all variables; 9 classes of behaviour are monitored. Three other models, Eq. (10), have been developed to constitute the extended NGOS.

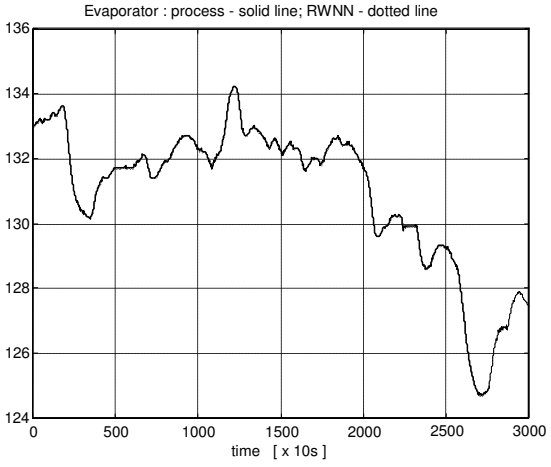
Data analysis and pre-processing. The selection of an appropriate learning data set has been done as a trade-off between the following criteria: - the selection of data from a day where the inputs have significant variation, i.e. maximum possible excitation of the process; - the number of missing values to be as small as possible; - the number of uncertain values to be as small as possible. The isolated missing and uncertain values have been replaced by means of polynomial interpolation. For each model, a training data set of 3000 rows has been selected. This corresponds to a period of time of about 8 hours, i.e. a production shift.

To develop a model, a spectral analysis of the experimental data has been performed. Based on this, a low-pass filtering has been applied to reduce the noise. This has been achieved by means of appropriate discrete-time Butterworth filters. This has also allowed for the reduction of the amount of data used in the RWNN learning. The data have been decimated using each 10-th sampled value. The obtained set was used only to train the RWNNs. Each identified model was tested using the complete training data set of 3000 rows and the testing data set from the previous month of plant exploitation.

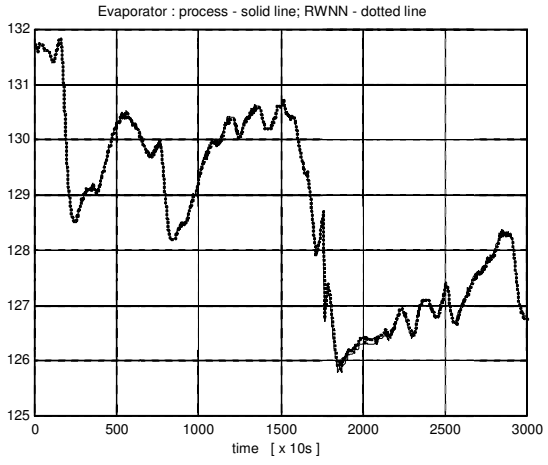
B. Experimental Results

The obtained results regarding system identification, residual generation and evaluation are presented in the following. For the detection and isolation of faults, a supplementary criterion of decision has been introduced. A fault is validated if the decision mechanism presented previously recognises a certain class at least 3 diagnosis cycles (sampling moments) of $T_s = 10$ s each, leading to elimination of false isolation and false alarms [18].

Fig. 4 presents some of the obtained identification results.



(a)



(b)

Fig. 4. Evaporator: identification results

Fig. 4a refers to the extended training data set, while Fig. 4b refers to the testing data set (different than the training data). In these figures, the process output has been represented with solid line and the output of the RWNN model has been represented with dotted line. The RWNN models are characterised by: $M \in \{3, 4\}$ and $n_C, n_D \in \{1, 2\}$.

The residuals (11) for the case of testing data with faults are generated. A number of four residuals resulted. All residuals are influenced by the faults on the output signal (samples 2100...2400). The second residual is almost not influenced by the faults on the first input (samples 600...900). The third residual is not influenced by the faults on the second input (samples 1200...1500), and the fourth residual is almost not influenced by faults on the third input (samples 2400…2700). For each faulty signal two classes are considered.

Altogether, 9 classes of sub-process behaviour are diagnosed: 1 - normal behaviour; 2 and 3 - positive and negative deviations of the output, respectively; 4 and 5 - positive and negative deviations of the first input, respectively; 6 and 7 - positive and negative deviations of the second input, respectively, and 8 and 9 - positive and negative deviations of the third input, respectively. The results of classification and final decision (diagnosis) are presented in Table I.

TABLE I
PERFORMANCE OF THE DIAGNOSIS SYSTEM

Recognition rates [%]	**Classification**	**Diagnosis**
Normal behaviour	98.85	99.30
Faulty behaviours	98.23	98.54
Global	98.57	99.23

V. CONCLUSIONS

The present paper investigates the development and application of a new type of recurrent wavelet neural network to robust fault diagnosis of an industrial plant. A neural approach to the generalised observer scheme is used. Residuals are generated by using the developed RWNN. The obtained symptoms are classified using a static Multi-Layer Perceptron. One remark the good results obtained with the FDI system based on the RWNN, reflected in recognition rate of around 99 %.

In the framework of fault diagnosis, further research will investigate the possibility to generate symptoms that reflect the behaviour of the process based on discrete wavelet decomposition. It is expected an improvement of the recognition rate due to the increase of the active robustness of the FDI system. This would be achieved because wavelet decomposition basically performs an analysis in both frequency and time domain.

REFERENCES

[1] Patton R.J., Clark R., Frank P.M., *Issues of Fault Diagnosis for Dynamic System*s, Springer-Verlag, New York, Inc., 2000

[2] Isermann R, *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*, Springer, 2006, ISBN 3-540-24112-4

[3] Chen J & Patton R J, *Robust Model-Based Fault Diagnosis for Dynamic Systems*, Kluwer Academic Publishers, 1999.

[4] Mirea L., Marcu T., "System identification using functional-link neural networks with dynamic structure", *IFAC Congress b'02*, Barcelona, Spain, CD-ROM Proceedings, 2002.

[5] Liu G P, Billings S A, Kadirkamanathan V., "Nonlinear identification via variable wavelet networks", *CD-ROM Proc. 14th IFAC World Congress*, Beijing, pp109-114, 1999.

[6] Coca D., Billings S. A., "Non-linear system identification using wavelet multi-resolution models", *Int. J. Control*, **74**, 18, p1718-1736, 2001.

[7] Zhang Q., Benveniste A., "Wavelet networks", *IEEE Trans. on Neural Networks*, **3**, 6, pp889-898, 1992.

[8] Oussar Y., Rivals I., Personnaz L., Dreyfus G., "Training wavelet networks for non-linear dynamic input-output modelling", *Neurocomputing*, **20**, pp. 173-188, 1998.

[9] Lesecq S., Gentil S., Taleb S., "Fault detection based on wavelet transform. Application to a roughing mill", *SAFEPROCESS 2006*, Beijing, Popular Republic of China, CD-ROM Proceedings, pp. 1177 – 1182, 2006.

[10] Odgaard P.F., Stoustrup J., Wickerhauser M.V., "Wavelet packet based detection of surface faults on compact discs", *SAFEPROCESS 2006*, Beijing, Popular Republic of China, CD-ROM Proceedings, pp. 1165 – 1170, 2006.

[11] Mirea L., Patton R. J., "Component Fault diagnosis using wavelet neural networks with local recurrent structure", *SAFEPROCESS 2006*, Beijing, Popular Republic of China, CD-ROM Proceedings, pp. 91 – 96, 2006.

[12] Daubechies I., *Ten Lectures on Wavelets*, SIAM Press, Philadelphia, 1992.

[13] Haykin S., (1994), *Neural Networks – A Comprehensive Foundation*, MacMillan, New York, 1994.

[14] Bezdek C J & Pal S K (Eds.),(1992), *Fuzzy models for pattern recognition. Methods that search for structures in data*, IEEE Press, 1992.

[15] Marcu T., "Pattern Recognition Techniques Using Fuzzily Labeled Data for Process Fault Detection", *Int. J. of Applied Mathematics & Computer Science*, **6**, 4, pp815-840, 1996.

[16] Marcu T, Mirea L, Frank P M & Kochs H D, "System identification and fault diagnosis using dynamic functional-link neural networks", European Control Conference *ECC'2001*, Porto, CD-ROM Proceedings, 2001.

[17] Bartys M., Patton R. J., Syfert M., Heras de las Sera, Quevedo J., "Introduction to the DAMADICS Actuator FDI Benchmark Study", *Control Engineering Practice*, (Invited Special Issue Paper), **14**: pp. 577-596, 2006.

[18] Marcu T., Mirea L., Ferariu L., Frank P.M., "Miscellaneous Neural Networks Applied to Fault Detection and Isolation of an Evaporation Station", *Prep. IFAC Symposium SAFEPROCESS2000*, Budapest, **1**, 352-357, 2000.

Intelligent Virus Spreading Simulator

A. Morariu, H. Valean, D.Bordencea

Abstract— **A pandemic is a worldwide epidemic of a disease. An influenza pandemic may occur when a new influenza virus appears against which the human population has no immunity. This paper focuses on simulation of the pandemic influenza and the graphic representation on the Romanian map, using three models: modified mathematical SIR model, in combination with Bayesian networks and Cellular Automata.**

I. INTRODUCTION

According to the World Health Organization (WHO), a pandemic can start when three conditions have been met:

- The emergence of a new disease to the population is increased.
- The agent infects humans, causing serious illness.
- The agent spreads easily and sustainable among humans**.**

The pandemic spread of some viruses depends on the specific characteristics of to the disease, the number and structure of the population, its dynamics, the existence of infection furnaces and/or risk factors (ex. animal farms or crowded people places with a high degree of insalubrities), the capacity of isolation for the infested area, the capacity to combat the disease's effects (the existence of well equipped hospitals in the contaminated area), the climate factors. It can be synthesized the fact that disease's spreading is achieved by three ways: proximity (a percentage of nearby population will get sick in time), by common transportation (primary and secondary service transportation on railway, public roads, with plane or ship) or, regarding diseases transmitted from animals, by their migration [1],[2].

Since the 1920's, stochastic models of epidemics have been used for viruses spreading modeling and simulation. Epidemic propagation models [3] have been applied on modeling the propagation of viruses [4].

This work is focused on the development of a general analytic model for avian influenza virus propagation. Based on this model, also a warning system for epidemic spreading is developed. Since the simulated and measured data are encapsulated and stored into databases, the implementation of a persistent connection to the database is a very important feature of the simulation software. Encapsulation is a design issue that deals with how functionality is compartmentalized within a system: you should not have to know how something is implemented to be able to use it [5],[6]. If encapsulation is used, the designer can build anything anyway he want, and then he can later change the implementation without affecting other components within the system, as long as the interface to that component did not change. By encapsulating the business logic of the application in domain/business classes and controller/process classes, the business logic can be used in more than one place. A persistence layer encapsulates the behavior needed to make objects persistent, in other words to read, write, and delete objects to/from permanent storage [6],[7].

II. THE MODEL

The model for avian influenza spreading proposed in this paper is based on three concepts: SIR model of epidemics [8], Bayesian networks and Cellular Automata. The SIR model is used for modeling the virus spreading into a populated area. The Bayesian networks are used for estimate the probability of virus spreading in the neighborhood of an infected area, and the Cellular Automata are used to predict the virus propagation.

A. The SIR model

This model is called SIR for Susceptible - Infected - Recovered. In this paper the authors propose an extension of the model (called Susceptible - Infected - Removed) by introducing a new parameter which allow during simulation to remove individuals (people that can die from the disease). The spread of avian influenza disease is the main topic during simulation. The individuals can be: *susceptible* to contact the disease, *infected*, *removed* (recovered or dead) and *immune* to the disease. Thus we have three groups or states in which we can place individuals. In addition we see that our data is a time series where we have a number of infected individuals at each point in time. Similarly we also have a number of susceptible and recovered individuals at each point in time.

The equations describing the virus spreading using the SIR model are presented in [9],[10].

As result, our data is a time series (Figure 1) where we have a number of infected individuals at each point in time. Similarly we also have a number of susceptible and recovered individuals at each point in time.

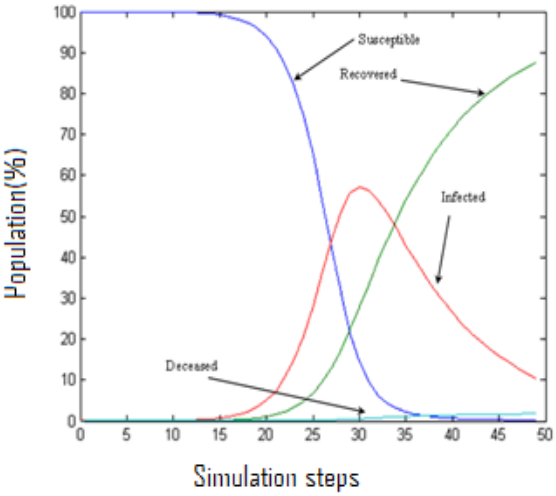


Figure 1: Virus spreading inside of area propagation

B. Bayesian network model

A Bayesian network is a probabilistic graphical model that represents a set of variables and their probabilistic independencies, and is used in the domain of artificial intelligence and statistics. Formally, Bayesian networks are directed acyclic graphs whose nodes represent variables and
whose arcs encode conditional independencies between the variables. If there is an arc from node *A* to another node *B*, *A* is called a *parent* of *B*, and *B* is a *child* of *A*. The set of parent nodes of a node *X*i is denoted by parents(*X*i). A directed acyclic graph is a Bayesian Network relative to a set of variables if the joint distribution of the node values can be written as the product of the local distributions of each node and its parents [11] f node *Xi* has no parents, its local probability distribution is said to be *unconditional*, otherwise it is *conditional*. If the value of a node is *bserved*, then the node is said to be an *evidence* node.

The equations describing the virus spreading between two localities are presented in [12],[13],[14]. In Figure 2 the Bayes network for virus spreading is presented.

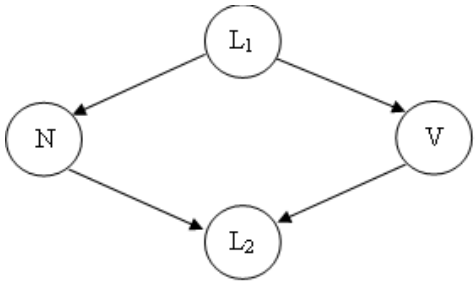


Figure 2: Bayesian network for virus spreading

L1 – populated area from where the virus is started
L2 – populated area where we which the virus can appear
N – Commuters from L1
V – Potential tourists from L1

C. Cellular automata model

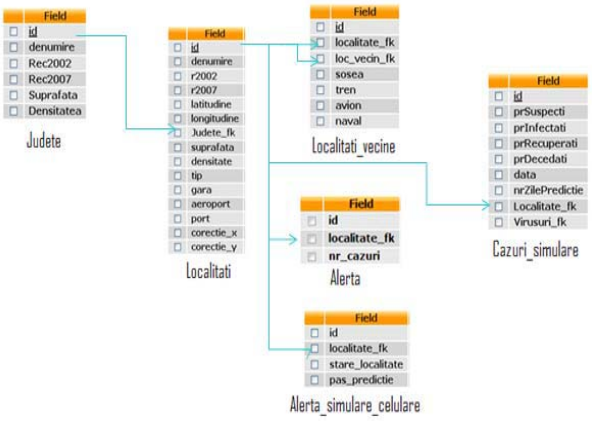
A cellular automata is a discrete model studied in computability theory, mathematics, theoretical biology and Microstructure Modeling [15],[16],[17],[18]. It consists of a regular grid of *cells*, each in one of a finite number of states. The grid can be in any finite number of dimensions. Time is also discrete, and the state of a cell at time *t* is a function of the states of a finite number of cells (called its *neighborhood*) at time $t - 1$. These neighbors are a selection of cells relative to the specified cell, and do not change. Every cell has the same rule for updating, based on the values in this neighborhood. Each time the rules are applied to the whole grid a new generation is created.

In this case the cells represent the Romania map areas.
Each cell can have 3 states:
0 – for the susceptible areas;
1 – for the infected areas;
2 – for the recovered areas;
The cells can change their states only from 0 to 1 or from 1 to 2.
A cell passes from 0 to 1 only if she has at least one neighbor in state 1.
A cell passes from 1 to 2 only after a time period assigned by the SIR model.
The model applied on cellular automata structure is presented in [19].

III. The System Architecture

The architecture of the system is based on the Model-View-Controller architectural pattern[20].

- The Model not only represents the data for the application, but it also represents persistent data, that is, data that persists beyond the life of the application. In other words, the model represents an application's persistent business domain objects. The model contains various Java classes. Some of the classes are entity classes – typical Java Persistence API entity objects. In the Java Persistence API, an entity instance, an instance of an entity object, represents a row of data in a database table. For example, "Judete" is an entity class that maps to the "Judete" table in the database. An instance of the "Judete" entity represents a row of data in the "Judete" table. Some of the main database tables and the relationships between them are:



Judete (Counties)

Field	Type	Description
Id	int	Primary key
Denumire	text	The county name
Rec2007	int	The county population in 2007
Suprafata	int	The area of the county
Densitate	double	The density of the population

Localitati (Localities)

Field	Type	Description
Id	int	Primary key
Denumire	text	The locality name
Rec2007	int	The population in 2007
Latitudine	float	Latitude
Longitudine	float	Longitude
Suprafata	double	The area of the locality
Densitate	decimal	The density of the locality
Tip	varchar	Type of locality (town or village)
Gara	tinyint	1 if railway station exists
Aeroport	tinyint	1 if airport exists
Port	tinyint	1 if seaport exists
Judete_fk	int	Foreign key that points to primary key from the table "Judete"

Alerta (Warning)

Field	Type	Description
Id	int	Primary key
Nr_cazuri	int	The number of infected people
Localitate_fk	int	Foreign key that points to primary key from the table "Localitati"

Alerta simulare celulare (Simulation warning)

Field	Type	Description
Id	int	Primary key
Stare_localitate	int	0 if the state of the locality is the initial state (uninfected), 1 if the state of the locality is infected and 2 if the state of the locality is recovered
Pas_predictie	int	The prediction step
Localitate_fk	int	Foreign key that points to primary key from the table "Localitati"

Localitati vecine (Neighbor localities)

Field	Type	Description
Id	int	Primary key
Sosea	tinyint	1 if in the locality with the id "Localitate_fk" is neighbor, on the road, with the locality with the id "Localitate_vecin_fk"
Tren	tinyint	1 if in the locality with the id "Localitate_fk" is neighbor, on the railway, with the locality with the id "Localitate_vecin_fk"
Avion	tinyint	1 if in the locality with the id "Localitate_fk" is neighbor, on the airport, with the locality with the id "Localitate_vecin_fk"
Naval	tinyint	1 if in the locality with the id "Localitate_fk" is neighbor, on the seaport, with the locality with the id "Localitate_vecin_fk"
Localitate_fk	int	Foreign key that points to primary key from the table "Localitati"
Loc_vecin_fk	int	Foreign key that points to primary key from the table "Localitati"

Cazuri simulare (Simulation cases)

Field	Type	Description
Id	int	Primary key
PrSuspecti	double	Susceptible people (in %)
PrInfectati	double	Infected people (in %)
PrRecuperati	double	Recovered people (in %)
PrDecedati	double	Dead people (in %)
NrZilePredictie	int	Prediction days horizon
Data	date	The time when simulation start
Localitate_fk	int	Foreign key that points to primary key from the table "Localitati"

- The view of this application is a GUI (Graphical User Interface) based on Romania territory maps. The maps are split in two levels: The first level represent the administrative map of Romania territory (Figure 3) and the second level represent the Romania counties (Figure 4).

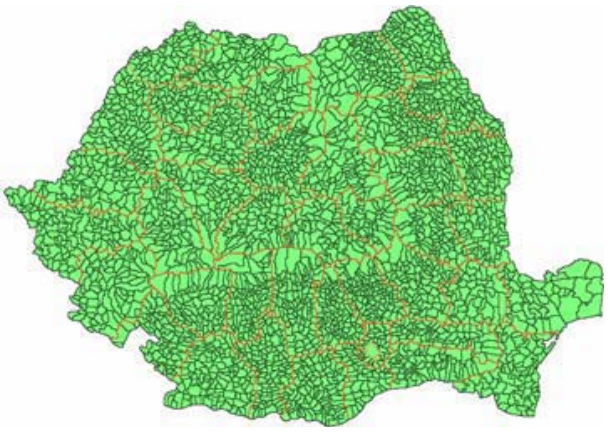


Figure 3: Romania territory map

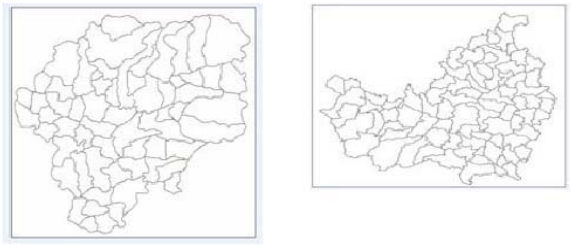


Figure 4: Two county maps (Bistrita and Cluj)

For representing the virus spreading on the maps, algorithms for encoding the colors, for determination of neighbor relations, for the approximation of the areas, for the determination of geographical centers and maps sticking are used. These algorithms are detailed presented in [21].

- The controller is the "brain" of the application. The controller decides what the user's input was, how the model needs to change as a result of that input, and which resulting view should be used. The controller has for parts: Controller 1 - reads the input given by the user (for example the locality from where the virus is starting to spread; which simulation model is selected by the user; Controller 2 – execute the SIR simulation model; Controller 3 – execute the Bayesian network simulation model; Controller 4 – execute the Cellular Automata simulation model. The most important controller classes are presented in Figure 5.

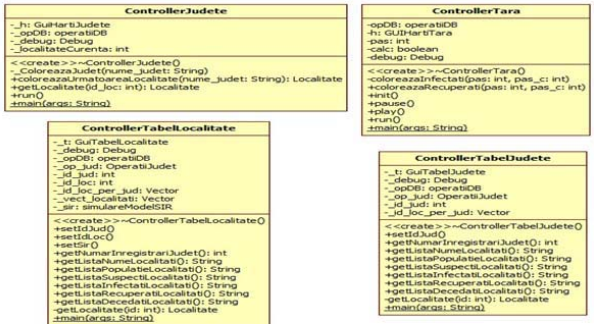


Figure 5: Controller class diagram

IV. TESTING AND SIMULATION

The virus spreading is split into three parts: first the SIR model is used to determine the evolution of the virus parameters inside of a populated area, taken into account several values to monitor the changes; second the Bayesian network model is used to simulate the virus spreading between two neighbor populated area; and third the Cellular Automata to simulate the virus spreading into a large area (like a county and then a country).The models are used to determine the evolution in time of the infected individuals in a populated area and between them. The simulation applies on the Romania territory map presented in Figure 6 and Figure 7.

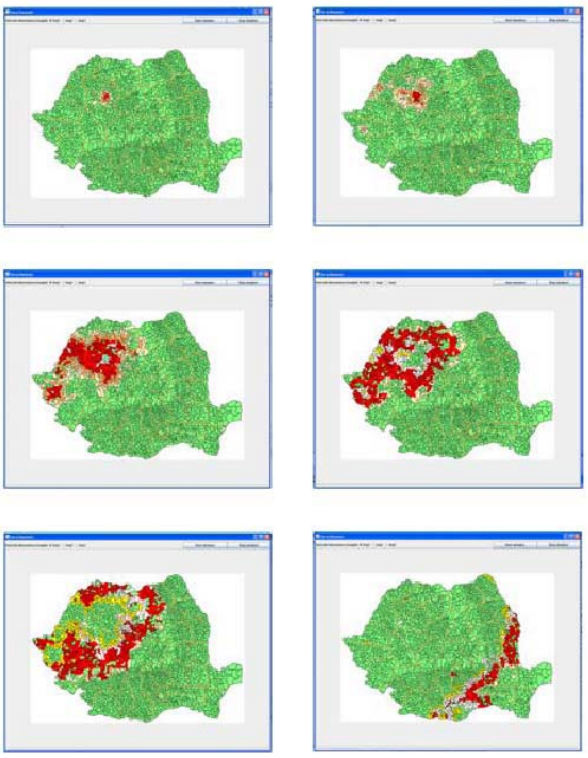


Figure 6: The evolution of epidemic

The code color used to represent the virus evolution in time is:

- Red - Maximum value of the infected population;
- Braun - Average value of the infected population;
- Beige - Minimum value of the infected population;
- White - First part of the recovering process;
- Yellow - Second part of the recovering process;
- Green - Uninfected or totally recovered population;

For simulation several input data were considered:
N - Total population;
I - Number of infected people;
B - Coefficient of infected persons during a time interval;
K - Recovering coefficient;

- Case 1:

For simulation the following input data are considered:
City: Cluj Napoca,
I = 5;
N = 315411;
$\beta = 0.6$
k=0.33

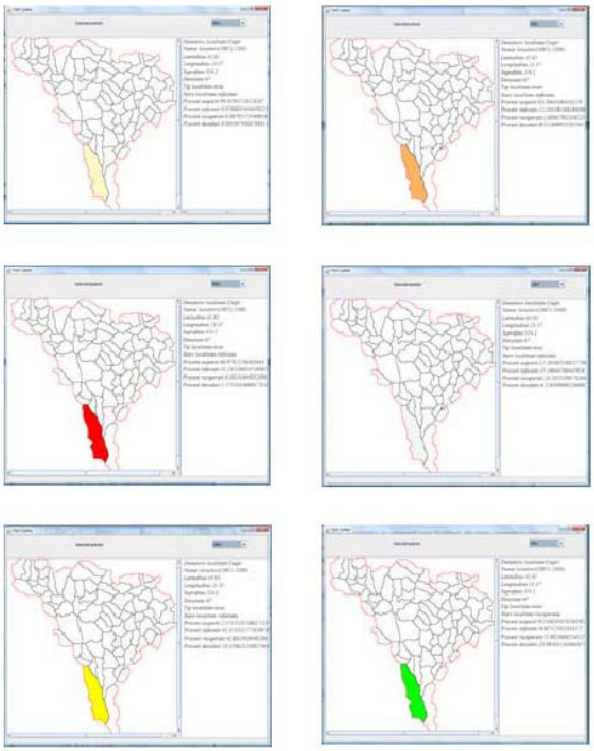


Figure 7: Different steps of simulation inside of a locality

In Figure 8 the simulation results are presented. Values are represented in percent.

It. step	Susceptible	Infected	Cured	Dead
1	99.9984	0.0015	0	0
2	99.9974	0.0019	0.0005	0
3	99.9962	0.0025	0.0011	0
4	99.9947	0.0032	0.0019	0
5	99.9928	0.0040	0.0030	0
…	…	…	…	
20	99.6753	0.1466	0.1743	0.0035
21	99.5876	0.1860	0.2218	0.0045
22	99.4765	0.2357	0.2819	0.0057
23	99.3357	0.2986	0.3582	0.0073
24	99.1577	0.37813	0.4548	0.0092
…				
40	73.7282	9.7658	16.1757	0.3301
41	69.4081	10.8632	19.3340	0.3945
42	64.8841	11.8023	22.8472	0.4662
43	60.2894	12.5022	26.6640	0.5441
44	55.7669	12.8990	30.7073	0.6266
45	51.4508	12.9584	34.8788	0.7118
46	47.4505	12.6824	39.0696	0.7973
47	43.8398	12.1079	43.1711	0.8810
48	40.6549	11.2972	47.0868	0.9609
49	37.8992	10.3248	50.7404	1.0355
50	35.5513	9.2654	54.0794	1.1036

Figure 8: Simulation results for Case 1

- Case 2:

For simulation the following input data are considered:
City: Cluj Napoca,

I = 5;
N = 315411;
β = 0.6
k=0.1

In Figure 9 the simulation results are presented. Values are represented in percent.

It. step	Susceptible	Infected	Cured	Dead
1	1	99.9984	0.0015	0
2	2	99.9974	0.0023	0.0001
3	3	99.9960	0.0035	0.0003
4	4	99.9939	0.0053	0.0007
5	5	99.9907	0.0079	0.0012
…	…	…	…	
20	20	95.9407	3.3717	0.6738
21	21	93.9998	4.9754	1.0042
22	22	91.1936	7.2840	1.4918
23	23	87.2081	10.5412	2.2056
24	24	81.6924	15.0027	3.2387
…				
40	0.7588	29.2905	68.5516	1.3990
41	0.6254	26.4948	71.4220	1.4575
42	0.5260	23.9447	74.0185	1.5105
43	0.4504	21.6258	76.3651	1.5584
44	0.3920	19.5217	78.4845	1.6017
45	0.3461	17.6154	80.3976	1.6407
46	0.3095	15.8905	82.1239	1.6759
47	0.2800	14.3309	83.6812	1.7077

Figure 9: Simulation results in Case 2

V. Conclusions and Further Works

In this paper, a map based simulation for spreading of avian influenza virus was presented, and solutions for some problems concerning the modelling of virus spreading and the processing of images that contain maps are given. Three different models were used in order to simulate the magnitude of an infection inside of a locality and the spread of the infection in the national territory. The models were tested and the results are presented above.

The spreading model can be used in order to generate early warnings if the condition of a pandemic disease apparition is satisfied. Also, this simulation and as result, the early warning generation can be extended to the EU territory.

The application is built for H5N1 virus. Because all the features of the virus and it spreading can be managed by users, the application can be easily extended to other viruses, such as H1N1.

REFERENCES

[1] Vittoria Colizza, Alain Barrat, Marc Barthelemy, Alain- Jacques Valleron, Alessandro Vespignani, "Modeling the World Spread of Pandemic Influenza: Baseline Case and Containment Interventions", www.plosmedicine.org,Volume4, 2007;

[2] A.Doyle, I.Bonmarin, D. Levy-Bruhl, Y. Le Strat, J.- C. Desenclos, Influenza pandemic preparedness in France: modelling the impact of interventions, J.Epidemiol Community Health, 2006, 60, 399-404.

[3] N.J.T. Bailey, "The Mathematical Theory of Infectious Diseases and Its Applications," *New York: Oxford University Press*, 1975.

[4] J.O. Kephart and S.R. White, "Measuring and Modeling Computer Virus Prevalence", *IEEE Computer Security Symposium on research in Security and Privacy.* Oakland, California, 1993.

[5] Scott W. Ambler – AmbySoft report,Encapsulating Database Access: An Agile "Best" Practice, 2007.

[6] [20] Scott W. Ambler – Agile Database techniques, John Wiley & Sons, ISBN: 0-471-20283-5.

[7] Scott W. Ambler – AmbySoft report, "The Design of a Robust Persistence Layer for Relational Databases", 2005

[8] T. Tassier. "SIR Model of Epidemics", *Anual report,* 2005.

[9] Honoriu Vălean, Adina Pop,Camelia Avram. Intelligent Model for virus spreading. 2007, Proc. of The International Symposium on System Theory, Automation, Robotics, Computers, Informatics, Electronics and Instrumentation SINTES 13, Craiova, pp :117-122; ISBN : 978-973-742-839-4; 978-973-742-841-7. IEEE-Romanian Section.

[10] Honoriu Vălean, Adina Pop ,Camelia Avram. Intelligent Model for Avian Influenza Virus. ISCA Proceedings 23rd International Conference on Computers and Their Applications .CATA-2008,9-11 aprilie, Cancun, Mexico, 2008,ISBN: 978-1-880843-66-6.

[11] *A Brief Introduction to Graphical Models and Bayesian Networks* - http://www.cs. berkeley.edu/~murphyk/ Bayes/bayes.html

[12] Adina Pop, Camelia Avram, Honoriu Valean. Hybrid Model for Spreading of Avian Influenza. 2008 IEEE International Conference on Automation, Quality and Testing, Robotics AQTR 2008, 22-24 May 2008, Tome 3, pp 66-72, IEEE Catalog Number CFP08AQT-PRT, ISBN 978-1-4244-2576-1, Library of Congress 2008904446.

[13] Adina Pop, Camelia Avram, Honoriu Vălean. Hybrid Model for Spreading of Avian Influenza. ACAM, 2008, ISSN 1221-437X, vol.17, nr. 4, pp. 613-621.

[14] Honoriu Vălean, Vasile Prejmerean, Adina Morariu, Ovidiu Ghiran. Map based simulation of panedmic influenza virus spreading. 21st International Conference on Computer Applications in Industry and Engineering CAINE-2008, 12-14 nov. 2008, Honoluluiu USA, ISBN 978-1-880849-69-7, pp. 66-71.

[15] R. Durrett, S. Levin, "The Importance of Being Discrete (and Spatial)", *Theoretical Population Biology 46:3 (1994) 363-394,* 1994.

[16] [18] D. S. Jones and F. Sleeman, "B. D. Ch. 14 in Differential Equations and Mathematical Biology", *London: Allen & Unwin,* 1983.

[17] [19] M.L. Martins, G. Ceotto, S.G. Alves, C.C.B. Bufon, J.M. Silva and F.F. Laranjeira, "A Cellular Automata Model for Citrus Variagated Chlorosis", *eprint arXiv: cond-mat/0008203*, 2000.

[18] [20] R. Willox, B. Grammaticos, A.S. Carstea and A. Ramani, "Epidemic Dynamics: Discrete-Time and Cellular Automaton Models", *Physica A 328*, 2003.

[19] Adina Morariu, Honoriu Vălean, Camelia Avram. Pandemic Virus Spreading Simulator. 22st International Conference on Computer Applications in Industry and Engineering CAINE-2009, 2-4 Nov. 2009, San Francisco, SUA. ISBN 978-1-880843-73-4, pp. 7-12.

[20] http://ootips.org/mvc-pattern.html.

[21] Adina Morariu, Silviu Folea, Honoriu Valean. RELIABLE AGENT BASED MONITORING SYSTEM. 24th International Conference on Computers and Their Applications, April 8-10, 2009 Holiday Inn Downtown-Superdome New Orleans, Louisiana, USA, ISBN: 978-1-880843-70-3, pp. 99-105

Hybrid modeling and optimal control of juggling systems

H.N. Nguyen and S. Olaru

Abstract— **The goal of this paper is to provide a modeling framework for juggling systems from a mixed logical dynamical point of view and to design a controller for the systems based on a model predictive control approach with soft constraints. In a first stage the basic juggling system, which consists of a ball and a juggling robot is detailed and subsequently the study is extended to the case of multiple balls juggling. The simulation results demonstrate the satisfactory performance of the presented approach.**

Keyword: Juggling systems, Mixed logical dynamical system, Model predictive control, Soft constraints.

I. Introduction

In the last decade, modeling and control of juggling systems has received significant attention in the control community, being motivated by wide applications in pedestrian flow, robotic manipulators, walking and jumping robot, mechanical system with impacts, etc.

However, modeling and control problems for this class of systems still represents an open topic with many unsolved problems. One of the complex phenomena is the abrupt change in the velocities which induces a discrete - continuous interplay (hybrid behavior).

Several modeling and control approaches have been proposed in the literature for these class of systems. A general framework for studying the modeling and control of juggling system in introduced in [1]. Based on the extended time domain for the system solutions (called hybrid time domain), the authors propose a hybrid control strategy for a 1 degree of freedom juggler. The main idea is to generate a virtual trajectory and subsequently solve the tracking problem for the platform.

Another framework for modeling and control of juggling system is studied in [2], [3], [4], upon a complementary - slackness juggling mechanical systems concept. In [5] a control law is derived for the force input applied to the platform in order to drive the system to a desired batting cycle.

An approach, which describes the motion in singular phase in enlarged spatial time scale is developed in [6]. Their idea comes directly from the methods of discontinuous time transformation developed originally for the optimal impulsive control problems.

In the present paper, we present a novel approach to the modeling and controller design problem for juggling system. This system is modeled as a hybrid system using the mixed logical dynamical (MLD) framework, the main idea being the translation of logic relations into mixed integer linear inequalities. Based on the resulting hybrid model, we formulate a finite time optimal control problem, for which nowadays there exists many effective solvers. Further improvement can be achieved for the MLD description by refining the classical model to a descriptor form (see [7] for further details).

This paper is organized as follows. Section II introduces a modeling framework for the standard juggling system while in Section III, the model predictive control (MPC) concepts are reviewed. The slack variables are introduced in Section III-C and the MPC concepts are reformulated in this context in Section III-D. We extend our control strategy to the case of juggling multiple balls in Section IV and evaluate the simulation results.

II. Hybrid modeling of juggling system

Consider a standard juggling system, which consists of a ball of mass m_b bouncing vertically on a juggling robot of mass m_p. The system has the state vector $x = (x_{11}, x_{12}, x_{21}, x_{22})^T$, with x_{11}, x_{12} the position and the velocity of the ball and x_{21}, x_{22} the position and velocity for the platform of the juggling robot, respectively.

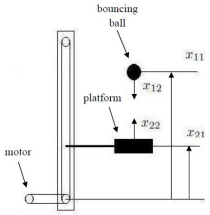


Fig. 1: Juggling system.

Before impact, the dynamics of the ball, in discrete time, are given by:

$$\begin{array}{rcl} x_{11}(t+1) & = & x_{11}(t)+x_{12}(t)T_s-\frac{gT_s^2}{2}; \\ x_{12}(t+1) & = & x_{12}(t)-gT_s. \end{array} \quad (1)$$

where: $t \in \mathbf{Z}_{\geq 0}$ is the current time, g is the gravity constant, T_s is the sampling time.

The juggling robot consists of a platform and a direct current motor, with the following dynamics:

$$\begin{array}{rcl} x_{21}(t+1) & = & x_{21}(t)+x_{22}(t)T_s+\frac{T_s^2}{2}u(t); \\ x_{22}(t+1) & = & x_{22}(t)+T_s u(t). \end{array} \quad (2)$$

The impact between the ball and the juggling robot occurs when $x_{11}(t)-x_{21}(t)=0$ and $x_{12}(t)-x_{22}(t)\leq 0$. The impact

H.N. Nguyen and S. Olaru are with SUPELEC Systems Sciences (E3S) - Automatic Control Department, Gif sur Yvette, France, Hoainam.Nguyen@supelec.fr, Sorin.Olaru@supelec.fr

is modeled by the following impact rule with conservation of momentum:

$$\begin{array}{lcl} x_{12}(t+1)-x_{22}(t+1) & = & -\lambda(x_{12}(t)-x_{22}(t)); \\ m_b x_{12}(t+1)+m_p x_{22}(t+1) & = & m_b x_{12}(t)+m_p x_{22}(t). \end{array} \tag{3}$$

where λ is the restitution coefficient, $\lambda \in (0,1]$.

In this paper it is assumed that $m_b \ll m_p$, the above equations, we can derive the simplified model:

$$\begin{array}{lcl} x_{12}(t+1) & = & -\lambda x_{12}(t)+(1+\lambda)x_{22}(t); \\ x_{22}(t+1) & = & x_{22}(t). \end{array} \tag{4}$$

The control problem of the standard juggling system has to consider one fundamental limitation: the position of the platform must be always lower (or equal) than the position of the bouncing ball. Starting from a valid initial condition and using a continuous-time description of the dynamics, this condition is implicitly verified all along the trajectories as long as the discontinuous change of the state at the impact prevents the system state to violate the constraint.

Then, the dynamics of the juggling system translated mutatis-mutandis from the continuous-time framework are described by:

- If $x_{11}(t)-x_{21}(t)=0$ and $x_{12}(t)-x_{22}(t)\leq 0$, then

$$\left\{ \begin{array}{lcl} x_{11}(t+1) & = & x_{11}(t); \\ x_{12}(t+1) & = & -\lambda x_{12}(t)+(1+\lambda)x_{22}(t); \\ x_{21}(t+1) & = & x_{21}(t); \\ x_{22}(t+1) & = & x_{22}(t). \end{array} \right.$$

- Else

$$\left\{ \begin{array}{lcl} x_{11}(t+1) & = & x_{11}(t)+x_{12}(t)T_s-\frac{gT_s^2}{2}; \\ x_{12}(t+1) & = & x_{12}(t)-gT_s; \\ x_{21}(t+1) & = & x_{21}(t)+x_{22}(t)T_s+\frac{T_s^2}{2}u(t); \\ x_{22}(t+1) & = & x_{22}(t)+T_s u(t). \end{array} \right. \tag{5}$$

However, specific precautions have to be taken for the constraints fulfillment in this discrete time framework due to the fact that the impermeability of the switching surface $x_{11}(t)=x_{21}(t)$ is not captured. In order to preserve the implicit constraints handling in the dynamics description, the impact conditions have to be adapted to $x_{11}(t)-x_{21}(t)\leq 0$ instead of $x_{11}(t)-x_{21}(t)=0$. This formulation assures the dynamics do not "miss" the impacts but introduce an uncertainty in the prediction model. The rigorous treatment of this inter sample uncertainty is beyond the scope of the present paper (see the literature on the modeling of the inter sample behavior [8], [9]). For the simulation purposes we use the following discrete time description of dynamics:

- If $x_{11}(t)-x_{21}(t)\leq 0$ and $x_{12}(t)-x_{22}(t)\leq 0$, then

$$\left\{ \begin{array}{lcl} x_{11}(t+1) & = & x_{21}(t); \\ x_{12}(t+1) & = & -\lambda x_{12}(t)+(1+\lambda)x_{22}(t); \\ x_{21}(t+1) & = & x_{21}(t); \\ x_{22}(t+1) & = & x_{22}(t). \end{array} \right.$$

- Else

$$\left\{ \begin{array}{lcl} x_{11}(t+1) & = & x_{11}(t)+x_{12}(t)T_s-\frac{gT_s^2}{2}; \\ x_{12}(t+1) & = & x_{12}(t)-gT_s; \\ x_{21}(t+1) & = & x_{21}(t)+x_{22}(t)T_s+\frac{T_s^2}{2}u(t); \\ x_{22}(t+1) & = & x_{22}(t)+T_s u(t). \end{array} \right. \tag{6}$$

Practically, the MLD framework will be used for modeling our juggling system [10], the main idea being to translate logic relations into mixed integer linear inequalities. These inequalities are combined with the continuous dynamical part, which is described by linear difference equations. A global MLD system is given by:

$$\begin{array}{l} x(t+1)=Ax(t)+B_1u(t)+B_2\delta(t)+B_3z(t); \\ y(t)=Cx(t)+D_1u(t)+D_2\delta(t)+D_3z(t); \\ E_2\delta(t)+E_3z(t)\leq E_1u(t)+E_4x(t)+E_5. \end{array} \tag{7}$$

where: $x=\left(\begin{array}{c} x_c \\ x_l \end{array}\right) \in \mathbf{R}^{n_c}\times(0,1)^{n_l}$ is the vector of continuous and logical state, $u=\left(\begin{array}{c} u_c \\ u_l \end{array}\right) \in \mathbf{R}^{m_c}\times(0,1)^{m_l}$ is the vector of continuous and logical control input, $y=\left(\begin{array}{c} y_c \\ y_l \end{array}\right) \in \mathbf{R}^{p_c}\times(0,1)^{p_l}$ is the vector of continuous and logical output, $\delta \in (0,1)^{r_l}$ is the vector of auxiliary logical variables, $z\in\mathbf{R}^{r_c}$ is the vector of auxiliary continuous variables and $A,B_1,B_2,B_3,D_1,D_2,D_3,E_1,\ldots,E_5$ are matrices of suitable dimensions.

The MLD model of the juggling system (6) is provided by **HYSDEL** (HYbrids System DEscriptions Language). The dimensions of corresponding variables are: $x(t)\in\mathbf{R}^4, u(t)\in\mathbf{R}^1, y(t)\in\mathbf{R}^1, z(t)\in\mathbf{R}^4$ and $\delta(t)\in(0,1)^3$. Matrices $A,B_1,B_2,B_3,D_1,D_2,D_3,E_1,\ldots,E_5$ have suitable dimensions.

III. Model predictive control

A. A classical formulation

Model predictive control (MPC) is an optimization based strategy, where a model of the plant is used to predict the future evolution of the system ([11], [12]). Based on this prediction and using the current state of the plant as the initial state, at each time instant t, the controller selects an optimal control sequence through an optimization procedure. Then the first control in the sequence is applied to the plant at time instant t, and at time instant $t+1$ the whole optimization procedure is repeated with new plant measurement.

For a fixed $N\in\mathbf{Z}_{\geq 1}$, let t be the current time, $X(t)=(x(1|t),\ldots,x(N|t))$ denote the state sequence predicted from the current state $x(t)$ and by applying the input sequence $U(t)=(u(0|t),\ldots,u(N-1|t))$. Consider the following optimal control problem([13]):

$$\begin{array}{l} \min_{U(t)} J(x(t),U(t))=||Q_f(x(N|t)-x_r(N|t))||_p+ \\ +\sum_{k=0}^{N-1}||Q_1(x(k|t)-x_r(k|t))||_p+||Q_u(u(k|t)-u_r(k|t))||_p+ \\ +||Q_\delta(\delta(k|t)-\delta_r(k|t))||_p+||Q_z(z(k|t)-z_r(k|t))||_p; \end{array} \tag{8}$$

s.t.

$$\begin{cases} E_2\delta(k|t)+E_3z(k|t)\leq E_1u(k|t)+E_4x(k|t)+E_5; \\ G_xx(k|t)\leq K_x; \\ G_uu(k|t)\leq K_u. \end{cases}$$

where $||\cdot||_p$ denote the p-norm of the vector, usually $p=1,2$ or ∞, the matrices $Q_f, Q_1, Q_u, Q_{delta}, Q_z$ are symmetric and positive semi-define for the case when $p=2$ and have full-column rank when $p=1,\infty$. x_r, u_r, δ_r, z_r are the reference signals for the system state, input, auxiliary logical variables and auxiliary continuous variables, respectively.

Using the optimal sequence of controls $U(t)^* = (u(0|t)^*,\ldots,u(N-1|t))^*$ for the problem (8), the MPC control law is defined as

$$u(x(t)) = u(0|t)^*. \tag{9}$$

B. Reference signal

The control of juggling system depends on the choice of the reference signal.

- If $r_1(t)\geq h$

$$\begin{cases} r_1(t+1) &= r_1(t)+r_2(t)T_s-\frac{gT_s^2}{2}; \\ r_2(t+1) &= r_2(t)-gT_s. \end{cases}$$

- Else, if $r_1(t)=h$ and $r_2(t)\leq 0$ then

$$\begin{cases} r_1(t+1) &= r_1(t); \\ r_2(t+1) &= -r_2(t); \end{cases}$$

Indeed, this reference signal is a trajectory of the ideal bouncing ball with unitary restitution coefficient.

The main drawback of this reference signal is the computation burden. Between impacts, the bouncing ball can be considered as an autonomous dynamical object. Based on the above reference, the MPC controller have to compute the control action in between the impacts, which are the only instants when the control action can influence the positions of the ball.

Imposing the meeting point of the platform with the bouncing ball at the fixed time-instances, one can generate the virtual trajectory for the platform. Our control problem is then reduced to a tracking problem for the platform, similar with open loop dynamic evolution of the ball, with unpredicted effected in case the meeting points are missed.

In this paper, the future evolution of the juggling system is predicted in order to evaluate the closest meeting point for the ball and platform positions. After finding viable conditions for impact occurrence, the velocity of the ball and the desired velocity are minimized aver the prediction window. This means that the impact times are not fixed, the references providing desired levels and desired velocities.

C. Introduction of slack variables

Even though the incorporation of the constraints in the problem formulation is the great advantage of MPC, it can lead to feasibility problems in presence of equality constraints on the predicted state. External disturbances, measurement noise or modeling errors can drive the system to a region where MPC problem is infeasible and no control action can be computed. Hence, it is important to have a way of dealing with these situations.

In practical applications, constraints are normally divided into two different classes: hard constraints and soft constraints. The input constraints can always be regarded as hard. A hard constraint is absolute, in that it cannot be violated. The state constraints is often seen as soft, their consideration being related to performance considerations. As a consequence, the violation of soft constraints can be allowed in case of infeasibility.

A straightforward way for softening constraints is to introduce slack variables ([14]). The slack variables are zero if no constraints are violated. By penalizing the non-zero values of the slack variables in the cost function, the constraint violations are kept to a minimum. The standard MPC formulation with slack variables can be written:

$$\begin{aligned} &\min_{U(t)} J(x(t),U(t)) = ||Q_f(x(N|t)-x_r(N|t))||_p + \\ &+\sum_{k=0}^{N-1}||Q_1(x(k|t)-x_r(k|t))||_p+||Q_u(u(k|t)-u_r(k|t))||_p + \\ &+||Q_\delta(\delta(k|t)-\delta_r(k|t))||_p+||Q_z(z(k|t)-z_r(k|t))||_p + \\ &+||Q_\varepsilon\varepsilon(k|t)||_p; \end{aligned} \tag{10}$$

s.t.

$$\begin{cases} E_2\delta(k|t)+E_3z(k|t)\leq E_1u(k|t)+E_4x(k|t)+E_5; \\ G_xx(k|t)\leq K_x+\varepsilon(k|t); \\ G_uu(k|t)\leq K_u; \\ -\varepsilon(k|t)\leq 0. \end{cases}$$

where Q_5 is the symmetric and positive semi-define matrix if $p=2$ and the full column rank matrix if $p=1,\infty$.

We observe that, in the case of feasible solution, $\varepsilon\to 0$ and the objective function (10) is equivalent to the objective function (8).

D. Reformulation of the model predictive control

In this section, we consider the control problem of the standard juggling system, using MPC.

For constraints handling, first of all, the position of the platform has to be lower than the position of the ball, leading to the hard constraints:

$$-x_{11}(k|t)+x_{21}(k|t)\leq 0; \tag{11}$$

Preferably the position of the platform will be kept at less than, for example, 90% of the magnitudes of the positions of the ball. Naturally this is impossible at impact. In order to avoid infeasibility we consider the following soft constraints

$$\begin{aligned} &-0.9x_{11}(k|t)+x_{21}(k|t)-\varepsilon_1(k|t)\leq 0; \\ &-\varepsilon_1(k|t)\leq 0. \end{aligned} \tag{12}$$

The control strategy for the juggling system is to catch an impact during the prediction window at the desired level. Subsequently the objective function changes in order to ensure that, by receding the prediction window, the existence of an impact at predicted step k at time t, leads at time $t+1$ to an impact at predicted step $k-1$. Precisely at the impact, the optimization problem minimizes the difference between the velocities of the balls and the reference velocities.

Explicitly, if $x_{11}(N|t)-h \geq \varepsilon$, then our objective function is given by:

$$J(x(t),U(t)) = |Q_1(x_{11}(N|t)-h)| + \sum_{k=0}^{N-1} |Q_u u(k|t)| + \\ + |Q_\varepsilon \varepsilon_1(k|t)|; \tag{13}$$

s.t.

$$\begin{cases} x(k+1|t) = Ax(k|t) + B_1 u(k|t) + B_2 \delta(k|t) + B_3 z(k|t); \\ -E_1 u(k|t) + E_2 \delta(k|t) + E_3 z(k|t) \leq E_4 x(k|t) + E_5; \\ -x_{11}(k|t) + x_{21}(k|t) \leq 0; \\ -0.9 x_{11}(k|t) + x_{21}(k|t) - \varepsilon_1(k|t) \leq 0; \\ -\varepsilon_1(k|t) \leq 0. \end{cases}$$

else, for r from $N-1$ to 1

$$J(x(t),U(t)) = |Q_2(x_{12}(k|t)-v)| + \sum_{k=0}^{N-1} |Q_u u(k|t)|; \tag{14}$$

s.t.

$$\begin{cases} x(k+1|t) = Ax(k|t) + B_1 u(k|t) + B_2 \delta(k|t) + B_3 z(k|t); \\ -E_1 u(k|t) + E_2 \delta(k|t) + E_3 z(k|t) \leq E_4 x(k|t) + E_5; \\ -x_{11}(k|t) + x_{21}(k|t) \leq 0; \\ x_{11}(r|t) - x_{21}(r|t) \leq 0. \end{cases}$$

where: ε is an accuracy level; $Q_1, Q_u, R, Q_\varepsilon$ are the full column rank weighting matrices; h, v is desired profile and desired velocity of the ball, respectively.

Algorithms for solving the optimization problems (13) and (14) are similar, so we will only detail here the transposition of the problem (13) into a mixed integer linear program(MILP) problem.

Define:

$$\begin{array}{lcl} \varepsilon_{x1} & \geq & Q_1(x_{11}(N|t)-h); \\ \varepsilon_{x1} & \geq & -Q_1(x_{11}(N|t)-h); \\ \varepsilon_u(k+1|t) & \geq & Q_u u(k|t); \\ \varepsilon_u(k+1|t) & \geq & -Q_u u(k|t). \end{array} \tag{15}$$

A sum $J(\varepsilon_{x1}, \varepsilon_u, \varepsilon_1) = \varepsilon_{x1} + \sum_{k=1}^{N} \varepsilon_u(k|t) + Q_\varepsilon \varepsilon_1(k|t)$ represents an upper bound on $J(x(t),U(t))$.

It is straightforward to prove that the vector:

$$s = \{u(0|k), \ldots, u(N-1|k), \delta(0|k), \ldots, \delta(N-1|k), z(0|k), \\ \ldots, z(N-1|k), \varepsilon_{x1}, \varepsilon_u(1|t), \ldots, \varepsilon_u(N|t), \varepsilon_1(1|t), \ldots, \varepsilon_1(N|t)\}$$

that satisfies equation (15) and simultaneously minimizes $J(\varepsilon_{x1}, \varepsilon_u, \varepsilon_1)$ also solves the original problem (13), i.e. the same optimum $J^*(x(t),U(t))$ is achieved. Therefore problem (13) can be reformulated as a MILP problem:

$$J(\varepsilon_{x1}, \varepsilon_u, \varepsilon_1) = \min_s \{\varepsilon_{x1} + \sum_{k=1}^{N} \varepsilon_u(k|t) + Q_\varepsilon \varepsilon_1(k|t)\}; \tag{16}$$

s.t.

$$\begin{cases} x(k+1|t) = Ax(k|t) + B_1 u(k|t) + B_2 \delta(k|t) + B_3 z(k|t); \\ -E_1 u(k|t) + E_2 \delta(k|t) + E_3 z(k|t) \leq E_4 x(k|t) + E_5; \\ -x_{11}(k|t) + x_{21}(k|t) \leq 0; \\ -0.9 x_{11}(k|t) + x_{21}(k|t) - \varepsilon_1(k|t) \leq 0; \\ -\varepsilon_1(k|t) \leq 0; \\ Q_{1f} x_{11}(N|t) - \varepsilon_{x1} \leq Q_{1f} h_1; \\ -Q_{1f} x_{11}(N|t) - \varepsilon_{x1} \leq -Q_{1f} h_1; \\ Q_u u(k|t) - \varepsilon_u(k+1|t) \leq 0; \\ -Q_u u(k|t) - \varepsilon_u(k+1|t) \leq 0. \end{cases}$$

Problem (16) can be rewritten in the more compact form:

$$\begin{array}{l} J = \min_s (f^T s); \\ \text{s.t. } Gs \leq Ex_t + W. \end{array} \tag{17}$$

where $f = [\overbrace{0,\ldots,0}^{N}, \overbrace{0,\ldots,0}^{Nr_c} \overbrace{0,\ldots,0}^{Nr_l}, \overbrace{1,\ldots,1}^{1+2N}]$ and G, E, W are the matrices of suitable dimensions obtained from (16).

By solving a MILP problem, at each time instant t, the optimal control sequence $U^*(t) = \{u^*(0|t), \ldots, u^*(N-1|t)\}$ is obtained and the first component $u^*(0|t)$ is applied to the plant (the rest is discarded). At the next time $t+1$ the whole procedure is repeated, based on the new measured state $x(t+1)$.

IV. Juggling system. Two balls case

A. Modeling

In this section, we consider a case of juggling system, with two balls bouncing on a juggling robot. This system has the state vector $x = (x_{11}, x_{12}, x_{21}, x_{22}, x_{31}, x_{32})^T$, where $x_{11}, x_{12}, x_{21}, x_{22}, x_{31}, x_{32}$ are the positions and the velocities of the first ball, the second ball and the platform, respectively.

The dynamic of this juggling system is given by:

- If $x_{11}(t) - x_{31}(t) \leq 0$ and $x_{12}(t) - x_{32}(t) \leq 0$ then

$$\begin{cases} x_{11}(t+1) &=& x_{31}(t); \\ x_{12}(t+1) &=& -\lambda x_{12}(t) + (1+\lambda) x_{32}(t); \\ x_{21}(t+1) &=& x_{21}(t); \\ x_{22}(t+1) &=& x_{22}(t); \\ x_{31}(t+1) &=& x_{31}(t); \\ x_{32}(t+1) &=& x_{32}(t). \end{cases}$$

- Else, if $x_{21}(t) - x_{31}(t) \leq 0$, $x_{22}(t) - x_{32}(t) \leq 0$ then

$$\begin{cases} x_{11}(t+1) &=& x_{11}(t); \\ x_{12}(t+1) &=& x_{12}(t); \\ x_{21}(t+1) &=& x_{31}(t); \\ x_{22}(t+1) &=& -\lambda x_{22}(t) + (1+\lambda) x_{32}(t); \\ x_{31}(t+1) &=& x_{31}(t); \\ x_{32}(t+1) &=& x_{32}(t). \end{cases}$$

- Else

$$\begin{cases} x_{11}(t+1) &=& x_{11}(t) + x_{12}(t) T_s - \frac{g T_s^2}{2}; \\ x_{12}(t+1) &=& x_{12}(t) - g T_s; \\ x_{21}(t+1) &=& x_{21}(t) + x_{22}(t) T_s - \frac{g T_s^2}{2}; \\ x_{22}(t+1) &=& x_{22}(t) - g T_s; \\ x_{31}(t+1) &=& x_{31}(t) + x_{32}(t) T_s + \frac{T_s^2}{2} u(t); \\ x_{32}(t+1) &=& x_{32}(t) + T_s u(t). \end{cases}$$

(18)

The control problem is characterized by an increased number of state constraints with respect to the standard juggling system. First of all, the position of the platform must be lower than the position of the juggling balls. Suppose that, our bouncing balls have to track some reference signals with desired profiles and desired velocities after impact. It is obvious from Figure 2 that one cannot impose the third impact for second ball, because at that specific time instant,

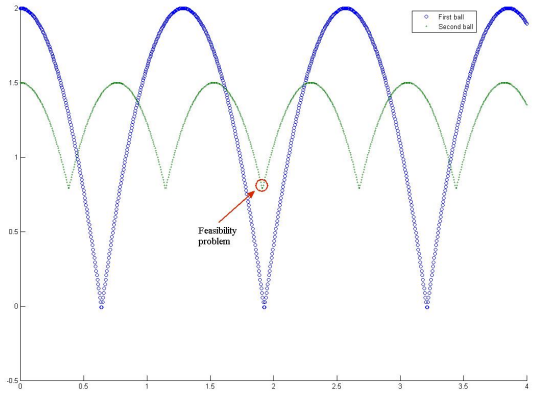


Fig. 2: Feasibility problem.

the first ball position is lower than the second ball position. This can lead to a different type of infeasibility problems.

Though the HYSDEL compiler, we obtain the equivalent MLD form for the two balls juggling system with the respective adaptation for taking into account the discrete time particularity. The dimensions of corresponding variables are: $x(t) \in \mathbf{R}^6, u(t) \in \mathbf{R}^1, y(t) \in \mathbf{R}^2, z(t) \in \mathbf{R}^6$ and $\delta(t) \in (0,1)^7$. Matrices $A, B_1, B_2, B_3, D_1, D_2, D_3,\ E_1, \ldots, E_5$ have suitable dimensions.

B. Optimization problem

First of all, the position of the platform must be always lower than the position of any one of two balls leading to the following hard constraints:

$$\begin{array}{l} -x_{11}(k|t) + x_{31}(k|t) \leq 0; \\ -x_{11}(k|t) + x_{32}(k|t) \leq 0; \end{array} \tag{19}$$

Practically the position of the platform is imposed to be less than, for example, 90% of the magnitudes of the positions of any one of two balls. This is in contradiction with the existence of impacts and in order to avoid infesibility we consider the following constraints softening mechanism:

$$\begin{array}{l} -0.9x_{11}(k|t) + x_{31}(k|t) - \varepsilon_1(k|t) \leq 0; \\ -0.9x_{21}(k|t) + x_{31}(k|t) - \varepsilon_2(k|t) \leq 0; \\ -\varepsilon_1(k|t) \leq 0; \\ -\varepsilon_2(k|t) \leq 0. \end{array} \tag{20}$$

Using the same technique, described in the section (3.3), if $x_{11}(N|t) - h_1 \geq \varepsilon$ and $x_{21}(N|t) - h_2 \geq \varepsilon$, the objective function is given by:

$$J(x(t),U(t)) = |Q_{11}(x_{11}(N|t) - h_1)| + |Q_{21}(x_{21}(N|t) - h_2)| + \sum_{k=0}^{N-1} |Q_u u(k|t)| + |Q_{\varepsilon_1}\varepsilon_1(k|t)| + |Q_{\varepsilon_2}\varepsilon_2(k|t)|; \tag{21}$$

with a set of constraints

$$\begin{cases} x(k+1|t) = Ax(k|t) + B_1u(k|t) + B_2\delta(k|t) + B_3z(k|t); \\ -E_1u(k|t) + E_2\delta(k|t) + E_3z(k|t) \leq E_4x(k|t) + E_5; \\ -x_{11}(k|t) + x_{31}(k|t) \leq 0; \\ -x_{21}(k|t) + x_{31}(k|t) \leq 0; \\ -0.9x_{11}(k|t) + x_{31}(k|t) - \varepsilon_1(k|t) \leq 0; \\ -0.9x_{21}(k|t) + x_{31}(k|t) - \varepsilon_2(k|t) \leq 0; \\ -\varepsilon_1(k|t) \leq 0; \\ -\varepsilon_2(k|t) \leq 0. \end{cases}$$

Else, if $x_{11}(N|t) - h_1 \leq \varepsilon$, for r from $N-1$ to 1

$$J(x(t),U(t)) = |Q_{12}(x_{12}(k|t) - v_1)| + \sum_{k=0}^{N-1} |Q_u u(k|t)|; \tag{22}$$

with the constraints

$$\begin{cases} x(k+1|t) = Ax(k|t) + B_1u(k|t) + B_2\delta(k|t) + B_3z(k|t); \\ -E_1u(k|t) + E_2\delta(k|t) + E_3z(k|t) \leq E_4x(k|t) + E_5; \\ -x_{11}(k|t) + x_{31}(k|t) \leq 0; \\ -x_{21}(k|t) + x_{31}(k|t) \leq 0; \\ -x_{11}(r|t) + x_{31}(r|t) = 0. \end{cases}$$

Finally, if none of the above then for r from $N-1$ to 1

$$J(x(t),U(t)) = |Q_{22}(x_{22}(k|t) - v_2)| + \sum_{k=0}^{N-1} |Q_u u(k|t)|; \tag{23}$$

with the constraints

$$\begin{cases} x(k+1|t) = Ax(k|t) + B_1u(k|t) + B_2\delta(k|t) + B_3z(k|t); \\ -E_1u(k|t) + E_2\delta(k|t) + E_3z(k|t) \leq E_4x(k|t) + E_5; \\ -x_{11}(k|t) + x_{31}(k|t) \leq 0; \\ -x_{21}(k|t) + x_{31}(k|t) \leq 0; \\ -x_{21}(r|t) + x_{31}(r|t) = 0. \end{cases}$$

where: ε is an accuracy level; $Q_{11}, Q_{12}, Q_{21}, Q_{22}, Q_u, Q_{\varepsilon_1}, Q_{\varepsilon_2}$ are the full column rank weighting matrices; h_1, v_1, h_2, v_2 is desired profiles and desired velocities of the first and second ball, respectively.

V. Simulation

Figure (3) shows a simulation results of a closed loop system, one ball case. Initial condition $x_0 = (1,0,0,0)^T$.

The prediction horizon is $N = 8$ and the weighting matrices:

$$Q_1 = 1; Q_2 = 1; Q_\varepsilon = 10^3; Q_u = 10^{-6}$$

The impact profile and desired velocity are:

$$\begin{array}{l} h(t) = 0.5, \text{ if } t \leq 2; \\ h(t) = 1, \text{ if } t \geq 2; \\ v(t) = 4. \end{array}$$

The simulation results demonstrate that the presented approach has a very good performance.

Figure (4) shows simulation results of the closed loop system in the two ball case. Initial condition: $x_0 = (1,0,0.9,0,0,0)^T$. Prediction horizon: $N = 6$. Weighting matrices:

$$\begin{array}{l} Q_{11} = 1, Q_{12} = 1, Q_{21} = 1, Q_{22} = 1, Q_u = 10^{-6}, \\ Q_{\varepsilon_1} = 10^3, Q_{\varepsilon_2} = 10^3 \end{array}$$

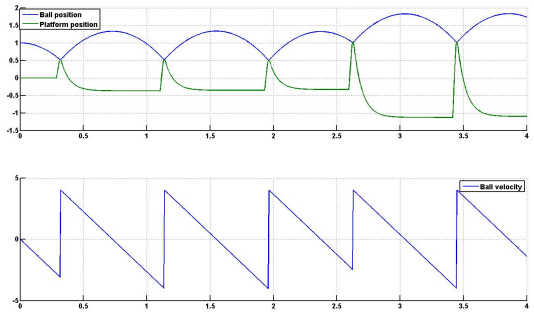


Fig. 3: One ball case. Position and velocity of the ball and the platform. System parameters: $g = 9.8$, $T_s = 0.005$, $\lambda = 0.8$.

The desired profile and desired velocity of first ball and the second ball:

$$h_1(t) = 0, v_1(t) = 6;$$
$$h_2(t) = 0.8, v_2(t) = 2.5.$$

It is interesting to note that, we cannot have a second impact for the second ball at level h_2, because at this specific time the position of the first ball is lower than the position of the second. The MPC controller avoids infeasibility by waiting until the position of the second ball is lower than the position of the first ball. We have the same situation for the fourth and seventh impact with the second ball.

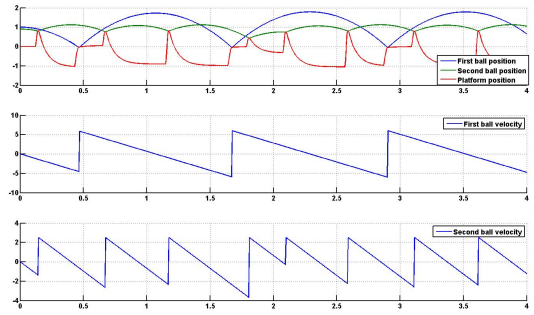


Fig. 4: Two balls case. Positions and velocities of the balls and the platform. System parameters: $g = 9.8$, $T_s = 0.005$, $\lambda = 0.8$.

Note that, for $x_{11} = x_{21}$ and $x_{12} = x_{22}$, the two balls trajectory coincides. This is demonstrated by result of the figure (5) with initial conditions $x_0 = (1.5, 0, 1.5, 0, 1.5, 0)^T$.

The impact profile and desired velocity of the first ball and second ball are:

$$h_1(t) = 0, v_1(t) = 6;$$
$$h_2(t) = 0.8, v_2(t) = 2.5.$$

providing strong constraints on the time of the impacts, and showing that we cannot de-synchronize the balls. Practically, this is not realistic, as long as m_b/m_p is small but non-zero and the velocity depends on the masses of the ball and the platform.

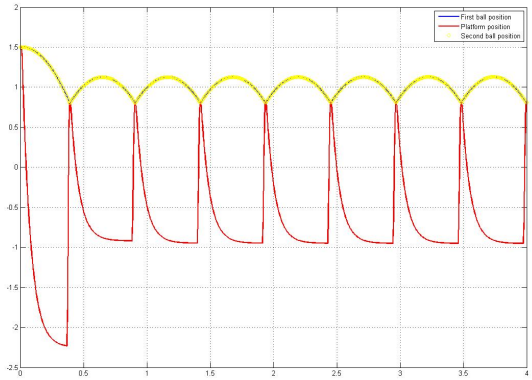


Fig. 5: Two balls case. Positions of the balls and the platform. System parameters: $g = 9.8$, $T_s = 0.005$, $\lambda = 0.8$.

VI. Conclusion

In this paper, a modeling framework and the corresponding model-based control was studied for a juggling systems. The approach makes use of a MLD description and synthesize the control action based on a model predictive control strategy with soft constrains. The efficiency of the proposed approach was demonstrated by a set of simulation results.

References

[1] R. Sanfelice, A. Teel, and R. Sepulchre, "A hybrid systems approach to trajectory tracking control for juggling systems," in *Proc. 46th IEEE Conference on Decision and Control*, pp. 5282–5287, 2007.

[2] B. Brogliato, *Nonsmooth mechanics: models, dynamics, and control*. Springer Verlag, 1999.

[3] B. Brogliato and A. Rio, "On the control of complementary-slackness juggling mechanical systems," *IEEE Transactions on Automatic Control*, vol. 45, no. 2, pp. 235–246, 2000.

[4] A. Zavala-Rio and B. Brogliato, "Direct adaptive control design for one-degree-of-freedom complementary-slackness jugglers," *Automatica*, vol. 37, no. 7, pp. 1117–1123, 2001.

[5] A. Zavala-Rio and B. Brogliato, "On the control of a one degree-of-freedom juggling robot," *Dynamics and control*, vol. 9, no. 1, pp. 67–90, 1999.

[6] B. Miller and J. Bentsman, "Optimal control problems in hybrid systems with active singularities," *Nonlinear Analysis*, vol. 65, no. 5, pp. 999–1017, 2006.

[7] K. Kobayashi and J. Imura, "System Representation for Logical Dynamics via a Descriptor Form," 2005.

[8] J. Skaf and S. Boyd, "Analysis and synthesis of state-feedback controllers with timing jitter," *IEEE Transactions on Automatic Control*, vol. 54, no. 3, pp. 652–657, 2009.

[9] R. Gielen, S. Olaru, M. Lazar, W. Heemels, N. van de Wouw, and S. Niculescu, "On polytopic inclusions as a modeling framework for systems with time-varying delays," *Automatica*, 2010.

[10] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *AUTOMATICA-OXFORD-*, vol. 35, pp. 407–428, 1999.

[11] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *AUTOMATICA-OXFORD-*, vol. 36, pp. 789–814, 2000.

[12] G. Goodwin, M. Seron, and J. De Doná, *Constrained control and estimation: an optimisation approach*. Springer Verlag, 2005.

[13] A. Bemporad, F. Borrelli, and M. Morari, "Model predictive control based on linear programming˜ the explicit solution," *IEEE Transactions on Automatic Control*, vol. 47, no. 12, pp. 1974–1985, 2002.

[14] E. Kerrigan and J. Maciejowski, "Soft constraints and exact penalty functions in model predictive control," in *Control 2000 Conference, Cambridge*, 2000.

On a hybrid control structure in automotive applications

Virginia Ecaterina Oltean, Radu Dobrescu and Dan Popescu

Abstract— **Hybrid systems, combing time-driven and event-driven dynamics, have emerged in past decade as an important modelling and control framework for automotive applications. An important class of problems related to performance improvement of modern cars is the cut-off control. In the hybrid dynamical model of the engine and power train, called in the sequel, the Balluchi model, the subsystem representing the power train is continuous, but the objectives of the cut-off control drive to switching control laws. This paper studies a simplified version of a cut-off control solution, earlier reported in the literature, as a hybrid supervision structure (HSS) with interface. Adequate techniques for MATLAB simulation of the closed loop hybrid system, ensuring Zeno path avoidance, are proposed.**

I. INTRODUCTION AND MOTIVATION

Hybrid systems, combing time-driven and event-driven dynamics [1], have emerged in past decade as an important modelling and control framework for automotive applications. This was motivated by the intrinsic hybrid nature of plants - like in the case of vehicles with four-stroke engines -, on one side, and by the increasing computational power of micro-controllers, capable to extend performance and functionality of electronic sub-systems controlling the car, on the other side.

An important class of problems related to performance improvement of modern cars is the control of oscillations damping in the process of deceleration, known in the literature as *cut-off control*. There are two basic design stages: firstly a state space partition of the second order oscillating subsystem of the state equations is requested, with an area centred in the origin, such that inside this area the system evolves free - the control drops to zero -, because the oscillations amplitude is decreased below a sensitivity threshold; secondly, a bang-bang control has to be built, in order to drive the state trajectory to the border of this area. A very refined solution, formulated as a hybrid optimal control problem, was reported in [2], [3].

In the hybrid dynamical model of the engine and power train called, in the sequel, the Balluchi model, the subsystem representing the power train is continuous, but the objectives of the cut-off control drive to switching control laws. This paper studies a simplified version of the cut-off control solution presented in [3], as a *hybrid supervision system* (HSS) with interface [7] [6]. Adequate MATLAB simulation techniques of the closed loop hybrid system, ensuring Zeno path avoidance are proposed. The paper is structured as follows. The Balluchi model, an analysis of the oscillating subsystem and the continuous time control problem are briefly presented in section II. The structure of the HSS is sketched in section III. The simplified version of the switching control law together with the state space partition of the associated HSS are deduced in section IV and the logical version of the HSS model is introduced in section V. Finally, a discussion concerning a set of MATLAB simulation experiments and concluding remarks are presented in the last two sections. The considered parameter values of the simulation model are the same as the ones specified in [2] and they are characteristic for a commercial car from the Magneti-Marelli Engine Control Division.

This work was supported in part by the National Center for Programs Management, under Grant CNMP-12-100/2008

V.E. Oltean, R. Dobrescu and D. Popescu are with Faculty of Automatic Control and Computers, Politehnica University of Bucharest, 77206 Bucharest, Romania ecaterina.oltean@aii.pub.ro

II. THE BALLUCHI POWER TRAIN MODEL AND THE CONTROL PROBLEM

A. The model structure

The model of the power train and engine with four cylinders considered in [3] has the structure depicted in Fig.1. The corresponding global hybrid dynamic model for a single cylinder, detailed in [2], comprises three interacting subsystems: a *finite state machine* (FSM), describing the piston dynamics, a *discrete event system* (DES), modelling the torque generator and a *continuous-time* subsystem, represented by the state equations of the powertrain. The four states of the FSM are: exhaust run, intake run, compression run and expansion. The power train model is given by the continuous-time equations

$$\begin{aligned} \dot{\zeta}(t) &= A^p\zeta + b^p u(t) - b_0 \\ \dot{\phi}_c(t) &= [\ 0 \quad 1 \quad 0\ ]\zeta(t) \\ a(t) &= (c^p)^T\zeta(t) \end{aligned} \tag{1}$$

with $\zeta = \left[\ \alpha_e \quad \omega_c \quad \omega_p\ \right]^T$ the state vector. α_e is the axle torsion angle, ω_c is the crankshaft revolution speed, ω_p is the wheel revolution speed and ϕ_c represents the crankshaft angle. The control signal u is the torque produced by the engine acting on the crankshaft and the vector b_0 models the resistant actions on the power train (denoted T and T_l in Fig.1, respectively). The system (1) is asymptotically stable with a real dominant pole $\lambda_1 < 0$ and a pair of complex poles $\lambda \pm j\mu$, $Re(\lambda) < 0$, responsible for the oscillations of the vehicle acceleration a, which is also the power train output of interest for drivability comfort.

Given the system (1), with admissible control signals $u : [0, \infty) \to \mathbf{R}$, $u(t) \in [0, M]$, $\forall t \geq 0$, together with a region in the state plane of the oscillating second-order subsystem of (1), where the amplitude of the acceleration drops below

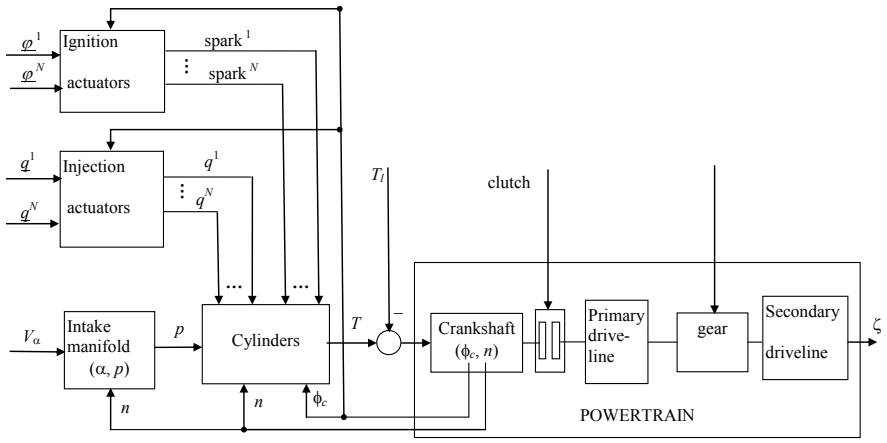


Fig. 1. The engine blocks and their communication topology - adapted from [3].

a comfort threshold, the first instance of *the control problem* asks for a control law which drives any initial state to the border of the specified region, in an arbitrary time $T < \infty$.

B. The oscillating subsystem and a relaxed version of the cut-off control problem

In order to isolate oscillations from the monotonic behaviour, a decoupling mode decomposition is applied. The transformed state equations are

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} u, \quad (2)$$

where

$$A_2 = \begin{bmatrix} \lambda & -\mu \\ \mu & \lambda \end{bmatrix}. \quad (3)$$

The new state components $x_1 \in \mathbf{R}$, $x_2 \in \mathbf{R}^2$ are given by

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} P_1 \\ P_2 \end{bmatrix} (\zeta - (A^p)^{-1} b_0), \quad (4)$$

with

$$P = \begin{bmatrix} P_1 \\ P_2 \end{bmatrix}, P^{-1} = \begin{bmatrix} \hat{P}_1 & \hat{P}_2 \end{bmatrix}, P_1 \in \mathbf{R}^{1\times 3}, P_2 \in \mathbf{R}^{2\times 3}.$$

The oscillating component $\tilde{a}$ of the acceleration can be expressed as the output of the subsystem

$$\dot{x}_2 = A_2 x_2 + b_2 u, \quad (5)$$

$$\tilde{a} = c_2^T x_2, c_2^T = (c^p)^T \hat{P}_2, \quad (6)$$

which has a stable behaviour, with damped oscillations. A MATLAB simulation experiment with the numerical values

$$\begin{aligned} \lambda \pm j\mu &= -2.671 \pm j21.54, \quad (7) \\ b_2 &= \begin{bmatrix} 1.923 \\ -14.32 \end{bmatrix}, \\ c_2^T &= \begin{bmatrix} 3.726 \cdot 10^{-2} & -2.523 \cdot 10^{-3} \end{bmatrix}, \end{aligned}$$

specified in [2] for a testing vehicle from the Magneti-Marelli Engine Control is illustrated in Fig.2.

The objective of the cut-off control is to minimize the peaks of the acceleration $\tilde{a}$, until they are less than a threshold of acceleration perception $a_{th} > 0$, i.e. $|\tilde{a}(t)| = |c_2^T x_2(t)| \leq a_{th}$, $\forall t \geq 0$. Consider the Euclidean norm $\|\cdot\|$ and the disk

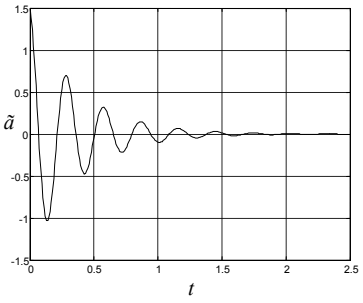


Fig. 2. Simulation of the free evolution of the oscillating component of the vehicle acceleration.

$$B_\rho = \{x_2 \in \mathbf{R}^2 : \|x_2\| \leq \rho, \rho = ath/\left\|c_2^T\right\|\}, \quad (8)$$

with the boundary

$$\partial B_\rho = \{x_2 \in \mathbf{R}^2 : \|x_2\| = \rho, \rho = ath/\left\|c_2^T\right\|\}. \quad (9)$$

In the free evolution of the system (5), the norm of the state vector decreases with time, as $d(x_2^T x_2)/dt = 2\lambda x_2^T x_2$, with $\lambda < 0$. Consequently, if $\exists t^* > 0$ so that $x_2(t^*) \in \partial B_\rho$, then $x_2(t) \in B_\rho$, $\forall t \geq t^*$ and also $|\tilde{a}(t)| \leq a_{th}$, $\forall t \geq t^*$.

The relaxed cut-off control problem is now: given the power train dynamic equations (1) with bounded admissible control inputs from the set $U = \{u : [0, \infty) \to \mathbf{R} : u(t) \in [0, M], \forall t \geq 0\}$, find $\hat{u} \in U$ so that for any $x_{02} \notin B_\rho$, the oscillating component of the acceleration $\tilde{a}$ (6) satisfies $\sup_{0\leq t\leq T} |\tilde{a}(t)|_{u=\hat{u}} \leq \min_{u\in U} \sup_{0\leq t\leq T} |\tilde{a}(t)|$ with the constraint $\dot{x}_2(t) = A_2 x_2(t) + b_2 u(t)$, $x_2(0) = x_0$, $x_2(T) \in \partial B_\rho$, where the arrival moment $T < \infty$ is arbitrary.

It can be shown that the requested control law, if it exists, is a bang-bang control [2], so the closed loop system can be abstracted to a HSS, as presented below.

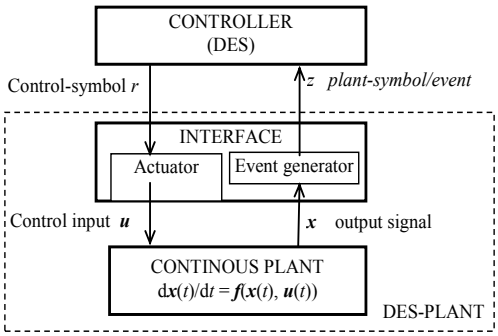


Fig. 3. The structure of the HSS.

III. THE STRUCTURE OF A HYBRID SUPERVISION SYSTEM

The structure of the HSS (Fig.3) considered next is a variant of the hybrid systems formalism with interface firstly proposed by P.J. Antsaklis and his co-workers [7] and detailed in [6]. A continuous time system is controlled, through an interface by a discrete event system (DES), represented by a deterministic Moore machine. The continuous plant evolves in a partitioned state space, defined by hypersurfaces, and it accepts piecewise constant control signals. The interface permits signal conversion between the two systems: when the state trajectory of the plant crosses a partition boundary, a plant-symbol is sent to the controller which, in response, sends a control-symbol through the interface, forcing the control signal to switch to a new value. For the DES controller, the plant coupled to the interface behaves like a nondeterministic DES, called the DES-plant.

Starting from a desired path of adjacent cells in the partitioned state space, the DES-plant is modelled as a nondeterministic automaton and the DES controller can be synthesized within the DES control theory of Ramadge and Wonham [8]. In the variant proposed in [6], it is shown that the HSS has two instances: a pure logical one and a pure continuous one, in which the DES controller coupled to the interface is remodelled as a switching control law (Fig.4).

IV. THE HSS MODEL OF THE CONTROLLED OSCILLATING SUBSYSTEM

The goal is to deduce a simplified version of the switching control law proposed in [2]. The first step is to build the partition of the state plane based on the specifications of the relaxed cut-off control problem. The first functional of the partition,

$$h_1 : \mathbf{R}^2 \to \mathbf{R}, h_1(x_2) = \sqrt{x_{21}^2 + x_{22}^2}, \tag{10}$$

describes the boundary (9). In order to define the second functional of tht state space partition, consider, as in [2], the equilibrium point of the oscillating system (5) for the maximal torque value $u = M$, given by

$$x_M = -A_2^{-1} b_2 M. \tag{11}$$

Define $z_\perp = R(\frac{\pi}{2})z$ the vector obtained by rotating in counterclockwise with $\frac{\pi}{2}$ the vector z and $v = -\overrightarrow{vers}$

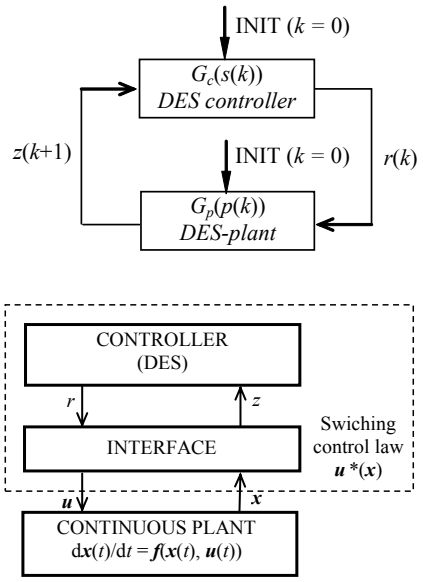


Fig. 4. Two approaches for the HSS in Fig.3: a pure logical one (up) and a nonlinear switching control system.

$(x_M)_\perp$ (Fig.5). The second functional of the state space partition is

$$h_2 : \mathbf{R}^2 \to \mathbf{R}, h_2(x_2) = v^T x_2. \tag{12}$$

Consider the following simplified switching control law:

$$u^*(x_2) = \begin{cases} 0, & \text{if } x_2 \in B_\rho \\ \begin{cases} 0, & \text{if } v^T x_2 \geq 0 \\ M, & \text{if } v^T x_2 < 0 \end{cases} & \text{if } x_2 \notin B_\rho \end{cases} \tag{13}$$

which differs, for $x_2 \notin B_\rho$, from the control law defined in proposed in equations (10) and (11) in [2]. Denote $u^* = u_1$ and let u_2 be the value of the control u_1 when the state trajectory is outside the disk B_ρ, as illustrated in Fig.6. The switching control law (13) can be rewritten in the form

$$u_1(x_2) = u_2(x_2) \cdot 0.5(1 + \text{sgn}(h_1(x_2))), \tag{14}$$

with

$$u_2(x_2) = 0.5\text{sgn}(1 - \text{sgn}(h_2(x_2))). \tag{15}$$

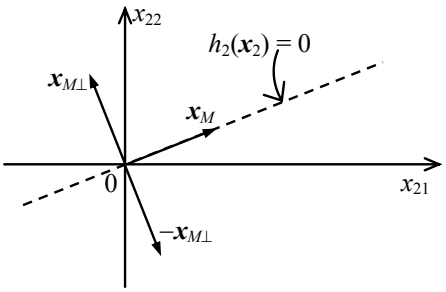


Fig. 5. The geometric significance of the functional h_2 .

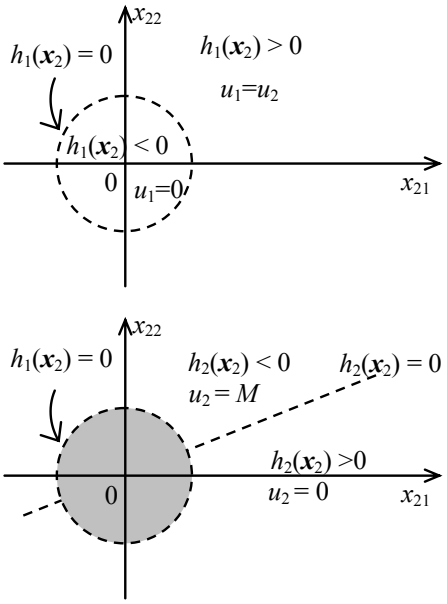


Fig. 6. The behaviour of the switching control law 13) with respect to the functionals h_1 (up) and h_2, respectively.

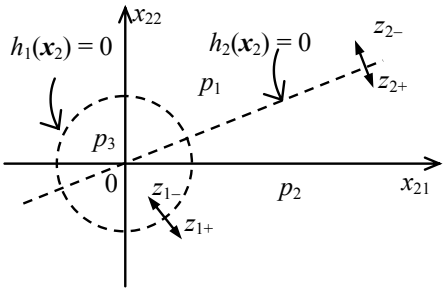


Fig. 7. The state space partition of the HSS associated to the simplified deceleration control law

V. THE LOGICAL MODEL OF THE HSS ASSOCIATED TO THE SIMPLIFIED CONTROL LAW

Starting from the state space partition defined by the functionals h_1 (10) and h_2 (12), the alphabet of the discrete states of the DES-plant model is defined by $P = \{p_1, p_2, p_3\}$, and the alphabet of the plant-symbols is $Z = \{z_{1+}, z_{1-}, z_{2+}, z_{2-}\}$, with the significance illustrated in Fig.7. The control takes one of the two values 0 or M, so the alphabet of control-symbols is $R = \{r_0, r_M\}$, with the significance: $r_0 \Rightarrow u_1 \downarrow 0$ and $r_M \Rightarrow u_1 \uparrow M$.

The DES-plant model is the automaton G_p and the DES controller is the Moore machine G_c (Fig.8). The DES-plant automaton G_p connected in close loop, from the initial state p_1, to the machine G_c (as depicted in Fig.4) can behave like the automaton in Fig.8, right. The automaton G_p is nondeterministic either starting from p_1, with control-symbol r_M, or starting from p_2, with control-symbol r_0. This can be solved by a refinement of the state space partition, which, unfortunately, is realizable only based on the phase portraits.

There are two possible evolutions of the closed-loop automaton in Fig.8, right: one reflects an unstable behaviour, according to the sequence of discrete states

$$\omega_1 = (p_1 p_2)^*, \tag{16}$$

while the other one corresponds to a stable trajectory, with damped oscillations until it enters the disk B_ρ (8), where it goes asymptotically to the origin. This is modelled by the discrete evolution

$$\omega_2 = (p_1 p_2)^n p_1 p_3, n \text{ integer}. \tag{17}$$

This last situation is illustrated in the simulation experiments. The particular value taken, in each experiment, by n may vary with the initial continuous state value and reflects the number of switchings between p_1 and p_2, generated when the continuous trajectory crosses the partition line $h_2(x_2) = 0$.

VI. SIMULATION MODELS

The experiments performed in MATLAB are based on the numerical data from the literature [2].

A. Continuous time simulation with lower upper torque value

Simulations refer the oscillating subsystem (5) controlled with the switching control law (13).

The first data set comprises:

- the parameters values (7),
- the initial state $x_{02} = [\ 40 \quad 2\ ]^T$,
- the maximal torque value $M = 12.41$ Nm,
- the threshold level for the module of the oscillating component of the acceleration $a_{th} = 0.294$.

With these values the resulting radius of the disk (8) is $\rho = 0.78723$ and the equilibrium point for maximal torque, i.e. outside the disk B_ρ, is $x_M = [\ 8.2607 \quad 0.00836\ ]^T$, with $\|x_M\| = 8.2611 > \rho$. The simulated continuous-time evolution corresponds to the sequence (17). The simulated phase plot is illustrated in Fig.9.

B. Continuous time simulation with higher upper torque value

For the same system as in previous case, *the second data set* consists of the first data set with upper torque value replaced by $M = 24$ Nm. The resulting equilibrium point is now, $x_M = [\ 15.797 \quad 0.1616\ ]^T$, with $\|x_M\| = 15.9763 > \rho$, so the difference is bigger than in the previous case. The simulated continuous-time evolution still corresponds to the sequence (17) and the simulated phase plot is illustrated in Fig.10.

C. Discrete-time simulation and Zeno path avoidance

If, for higher upper torque values, the initial state x_{20} is not far enough from the equilibrium point x_M, then the frequency of the oscillations outside the disk B_ρ increases and the simulation may become untractable with the classic variable step integration methods, due to Zeno behaviour. An

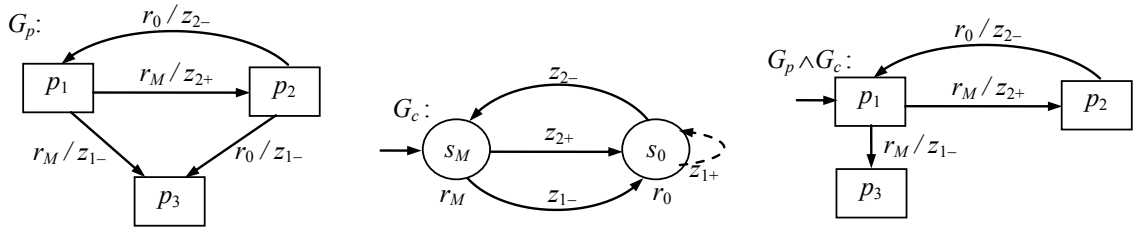


Fig. 8. The DES-plant automaton (left), the DES controller (centre) and a possible model of the closed loop discrete model (right).

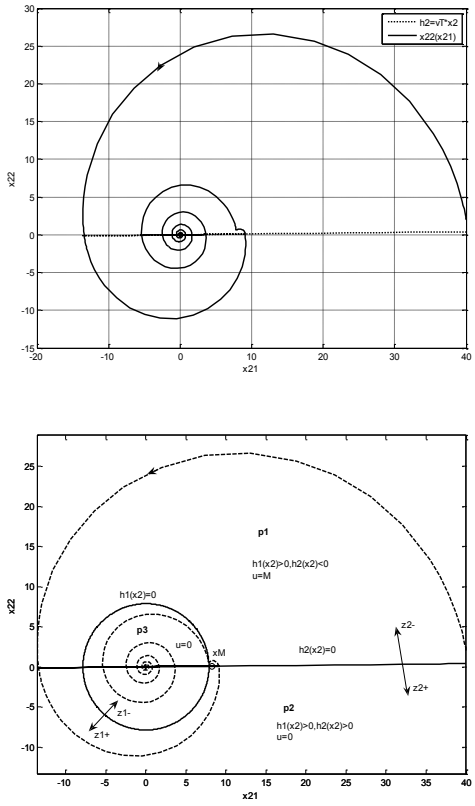


Fig. 9. Simulation of the controlled oscillating subsystem with the first data set: the phase portrait with switching line $h_2(x_2) = 0$ (up) and the behaviour with respect to the state space partition (down).

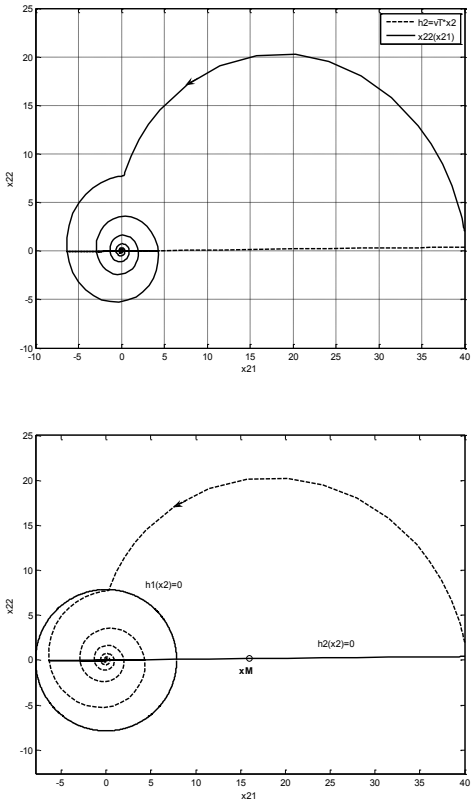


Fig. 10. Simulation of the controlled oscillating subsystem with the second data set: the phase portrait with switching line $h_2(x_2) = 0$ (up) and the behaviour with respect to the state space partition (down).

approach to this problem consists of simulating the sampled dynamical system corresponding to (5), given by

$$x_{2d}(k+1) = A_{2d}x_{2d}(k) + b_{2d}u_{2d}(k), \qquad (18)$$

with $x_{2d}(k) = x_2(kh)$, $u_d(k) = u(kh)$ and $h > 0$ is the sampling step. Theoretically, Zeno path avoidance is ensured, in this case, for any value of h, because for discrete time dynamical equations, the computation of the current state value is obtained by summation over past values, so the simulation process doesn't get stuck, as it happens when using variable step techniques beyond the Zeno time [5]. The matrices in (18) are deduced from the corresponding matrices in (5), with the well-known formulas

$$A_{2d} = \exp(hA_2), b_{2d} = \int_0^h \exp\theta A_2 d\theta \cdot b_2, c_{2d} = c_2. \quad (19)$$

The third data set consists of the first data set with following replacements:

- the initial state $x_{02} = [\ 19 \quad 1\ ]^T$,
- the maximal torque value $M = 24$ Nm.

The simulated phase plot of the system (18), with sampling step $h = 0.005$, is illustrated in Fig.11.

VII. CONCLUSIONS

The problem of the cut-off control is to ensure an admissible upper bound to the amplitude of the oscillating

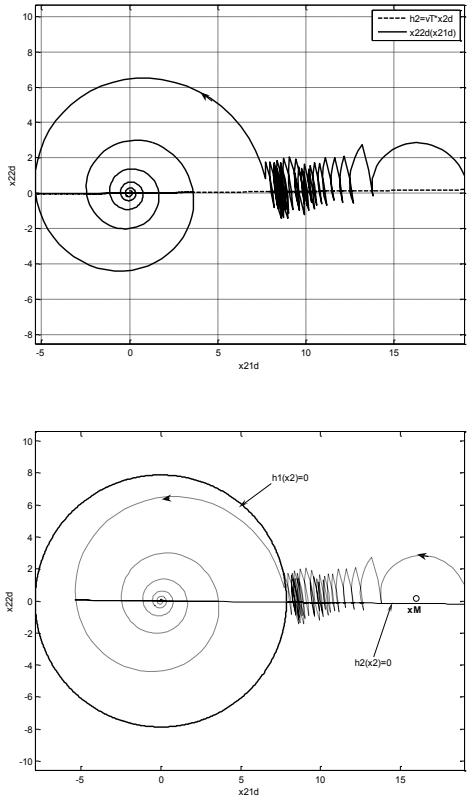


Fig. 11. Simulation of the controlled oscillating subsystem (18) with the third data set: the phase portrait with switching line $h_2(x_{2d}) = 0$ (up) and the behaviour with respect to the state space partition (down).

component of the acceleration, quantified by a threshold value.

Starting from the solution presented in [2], an approximation of the switching control-law is proposed, which is simpler and still valid, if some generic constraints are satisfied. These constraints concern the upper value of the generated torque and the distance between the initial state and the equilibrium point corresponding to maximal torque value. Informally, the generated torque has not to exceed a certain limit, so that the norm of the corresponding equilibrium point be lower than the radius of the disk centred in the origin, resulted from the comfort threshold value imposed to the acceleration. Within this disk, the trajectory evolves free towards the origin.

The proposed simplified control-law is a bang-bang control, which drops to zero when the trajectory of the oscillating subsystem enters the disk centred in the origin.

For the purely continuous oscillating subsystem associated to the aceleration, the local closed loop system can be considered as a HSS with interface. In order to build this HSS model, a formal description of the state space partition was proposed, defined by two functionals. Then, a compact form of the simplified control-law was deduced, which was successfully implemented also within the simulation experiments. The discrete evolutions of the automata associated to the logical version of the HSS provide a qualitative evaluation of the simulated trajectories, related to the state space partition. Also, a simple technique for Zeno path avoidance of the controlled subsystem was introduced, based on fixed-step sampling of the continuous-time subsystem.

Summing up, this study proposes: (*i*) the simplified switching control law (13), with the compact formulas (14)-(15), used for MATLAB numerical simulations, (*ii*) the logical interpretation of the controlled system, in section V, providing a qualitative interpretation of the continuous-time controlled evolution, as illustrated by (*iii*) the simulation models in subsections VI-A, VI-B and, finally, (*iv*) a discrete-time simulation model, in subsection VI-C, motivated by the necessity to avoid a potential Zeno behaviour during the simulation process.

It has to be noted, also, that the proposed hybrid systems approach for the cut-off control is different from the mixed logical dynamical (MLD) approach discussed in [4] in at least two main aspects: firstly, in the HSS, the process is purely continuous and the controller is a DES, while in MLD systems, continuous and logical variables appear in a single mixed state vector, so the process itself can be hybrid; secondly, the HSS can be simulated in MATLAB as a classic nonlinear system with switching control law (see Fig.4), while the MLD formalism needs a dedicated library. Also, the MLD model uses, for the time-driven part of the state equations, directly a discrete time model. As a final remark, the HSS approach is advantageous at least in cases of lower order systems, because it is quite simple and intuitive and also because it can be simulated in a general purpose MATLAB environment. A more detailed discussion in this direction is beyond the scope of this paper.

Future work implies the extension and implementation of these considerations for real-time control of a car emulator, and, in parallel, a theoretical comparative study of main recent hybrid modelling approaches for automotive systems.

REFERENCES

[1] C.G. Cassandras and S. Lafortune, "Discrete Event Systems: The State of the art and New Directions", *in Applied and Comput. Control, Signals and Circuits*, vol. 1, 1999, pp 83-148.

[2] A. Balluchi, M.D. Di Benedetto, C. Pinello, C. Rossi, A. Sangiovanni-Vincentelli, Hybrid Control in automotive applications : the cut-off control, *Automatica*, vol. 35, 1999, pp 519-535.

[3] A. Balluchi, L. Benvenutti, M.D. di Benedetto, C. Pinello and A. Sangiovanni-Vincentelli, Automotive engine control and hybrid systems: Challenges and opportunities, *in Proceedings of the IEEE*, vol. 88, 2000, pp 888-912.

[4] A. Bemporad and M. Morari, Control of systems integrating logic, dynamics, and constraints, *Automatica*, vol. 35, 1999, pp 407-427.

[5] Virginia Ecaterina Oltean, On simulation of Zeno hybrid systems, *Revue Roumaine des Sciences Techniques, Srie lectrotechnique et nergtique*, vol. 52, no. 2, 2007, pp 229-239.

[6] Virginia Ecaterina Oltean, On Qualitative Behaviours of a Class of Piecewise-Linear Control Systems (Part I: Basic Models), *Rev. Roum. Sci. Techn. - lectrotechn. et nerg.*, vol. 54, no. 1, 2009. pp 95-104.

[7] J.A. Stiver , P.J. Antsaklis and M.D. Lemmon, A Logical DES Approach to the Design of Hybrid Control Systems, *Technical Report of the ISIS Group at the University of Notre Dame*, ISIS-94-011, 1994.

[8] W.M. Wonham, Supervisory Control of Discrete Event Systems, (updated 2008.07.01), http://www.control.toronto.edu/people/profs/wonham.

Robust Analysis Approach for Prediction of Pilot Induced Oscillations

Valentin Pană and Adrian-Mihail Stoica

***Abstract*— The paper presents a method to detect possible Pilot in the Loop Oscillations (PIO) based on robust stability analysis of a system subject to parametric uncertainty. The nonlinear elements are substituted by linear uncertain parameters. The proposed method addresses the robustness versus both time-invariant and time-varying uncertainty when multiple nonlinearities are present in the pilot-vehicle system.**

I. Introduction

An important problem in the design of flight-control systems of aircrafts is the determination of handling qualities and pilot induced oscillations (PIO) tendencies due to nonlinearities effects.

A pilot induced oscillation (PIO) is a complex interaction between the human pilot and the aircraft that leads to sustained and sometimes very large amplitude oscillations of the aircraft. It is characterized by a loss of stability margin in the pilot-aircraft closed loop system. These oscillations can occur about any of the aircraft's axes of symmetry.

Many flight test accidents and incidents have been attributed to PIO problems. Most recently, both the F-22 and JAS-39 prototypes have crashed as a result of PIO incidents. Commercial aircraft are also not immune to PIO problems (A-320, Boeing 777). The potential occurrence of PIO is amplified by the use of modern control technology including fly-by-wire systems that determine important modification of the airplane response characteristics. For example in heavy aircrafts, the problems result in a faster roll rate than normally expected. This combined with delays introduced by the fly-by-wire system cause PIOs.

The origin of these oscillations is a missadaptation between the pilot and the aircraft during some tasks in which tight closed loop control is required (such as aerial refueling or aircraft-carrier landing), when the aircraft is not responding to pilot commands as expected. The result can trigger a pilot action capable of driving the aircraft out of pilot control.

Detailed analytical studies of PIO incidents are based on pilot behavioral models and closed loop analysis procedures designed to understand and rationalize the phenomena and their associations. The classification of PIO [15] takes into account some possible different behaviors of the closed loop pilot vehicle system during the PIO. Recently a new category (*IV)* has been added to account for another type of interaction in the pilot vehicle system.

PIO Category I – Essentially Linear Pilot-Vehicle System Oscillations: The effective controlled element characteristics are essentially linear, and the pilot behavior is also quasi-linear and time stationary.

PIO Category II – Quasi-Linear Pilot Vehicle System Oscillations with Rate Limiting or Position Limiting: The closed loop pilot vehicle system has a nonlinear behavior, mainly characterized by the saturation of position or rate limited elements.

PIO Category III – Essentially Nonlinear Pilot Vehicle System Oscillations with Transitions: These PIO depend on nonlinear transitions in either the effective controlled element or in the pilot's behavioral dynamics.

PIO Category IV – Refers to coupling effects between pilot inputs and the aircraft structural modes.

In the present paper an analysis method to predict Category II PIO is considered. These oscillations are induced by nonlinearities determined by rate or position saturations of control surface actuators. This kind of nonlinearity is present in any aircraft, because of the physical constrains of elements such as stick deflections, actuator position and rate limiters, limiter in the controller software. Actuator rate limiting occurs when the input rate to the control surface exceeds the hydraulic and/or mechanical capability of the control surface actuator. Rate limiting has been identified with PIO for two main reasons.

First, it introduces additional phase lag, or delay, between commanded control surface position and actuator control surface. The time delay caused by the additional phase lag can drive the pilot to compensate with faster inputs, worsening the situation. This can ultimately lead to a PIO or unstable situation.

The second reason rate limiting has been identified in PIO is the reduction in gain. The pilot sees this as a reduction in control effectiveness, so he may compensate with larger inputs making the problem worse. These effects often mislead the pilot into thinking the aircraft is not responding to his inputs. These two rate limiting concepts are illustrated in figure 1.

This work was supported by the CNCSIS Grant no. 1721.

Valentin Pană is with the University Politehnica of Bucharest, Faculty of Aerospace Engineering, Str. Polizu, No. 1, Bucharest, Romania, (e-mail: valentin_pana@yahoo.com).

Adrian-Mihail Stoica is with the University Politehnica of Bucharest, Faculty of Aerospace Engineering, Str. Polizu, No. 1, Bucharest, Romania, (e-mail: amstoica@rdslink.ro).

Some methods for the analysis of Category II PIO are currently available. *Describing Function* (DF) method, which is the traditional method to analyze the amplitude and frequency of a limit cycle by linearizing the nonlinear elements (see for instance [15]). New methods have been investigated over the last years: *Open Loop Onset Point* (OLOP) method [10] with a modified/enhanced version [9], derived from the describing function method, Robust Stability Analysis Methods [3], [4], [20] also considered in this paper, Time Domain *Neal Smith Criterion* [5], μ-*analysis* based method. Also, solutions to alleviate the effects of PIO have been proposed: phase compensation [17], anti-windup synthesis for PIO avoidance [19], nonlinear pre-filters for PIO prevention [14].

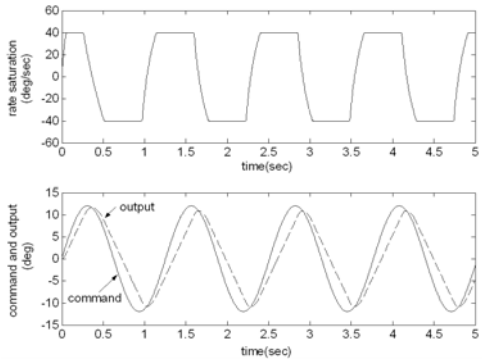


Fig. 1. Example time history of rate limiting

The aim of the present paper is to compare two analysis methods used to predict the PIO occurrence. The first one is based on a robustness analysis with respect to parametric uncertainties and the second one is based on the linear matrix inequalities version of the Popov criterion for absolute stability. Such methods have been used previously for instance in [4] for a single nonlinearity. In the present paper multiple nonlinearities generated by position and rate saturation are considered. The paper is organized as follows: the next section presents the transformation needed in order to obtain a suitable system for robustness analysis. In section III the analysis method based on the Edge theorem approach is presented. In section IV the condition for absolute stability of the system with sector type nonlinearity is described. The paper ends with some concluding remarks.

II. PIO Detection Using Robust Stability Analysis With Respect to Parametric Uncertainties

A possible method to handle saturations is presented in [3], [4] where the nonlinear elements are replaced by linear elements with uncertain gain. Thus an equivalent robustness problem with respect to parametric uncertainty is obtained.

Two different criteria for PIO analysis will be presented in the following sections. Both are based on robust stability analysis of a linear system, obtained by substituting the nonlinear elements with linear uncertain gain. In the first criterion the uncertain parameter is assumed time-invariant. An approach based on the Edge Theorem is used for the analysis. As it will be shown in section IV more realistic analysis takes into account that the uncertain parameters are *time-varying*.

In order to illustrate the proposed methods a classical closed loop scheme for the study of Category II PIO occurrence in the pitch axis is considered, namely the X-15 research aircraft model (figure 2). In this model a nonlinearity is added to account for stick limit deflections.

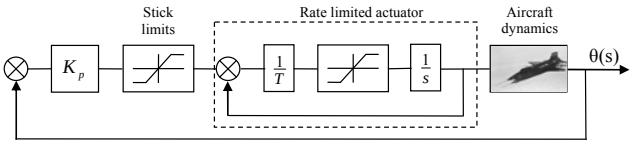


Fig. 2. Closed loop diagram for Category II PIO analysis

The blocks in figure 2 are: K_p the human pilot gain, a saturation block representing the stick limits, the block diagram of a rate limited actuator, T is the time constant of the first order actuator dynamics and the aircraft dynamics transfer function $\theta(s)/\delta(s)$ from the controlled surface position to the variable controlled by the pilot.

In this scheme the rate limited actuator was replaced with an equivalent saturation model that is suitable for our analysis [15]. This type of modeling can be checked using the following SIMULINK model. As shown in figure 3 for a sinusoidal input with amplitude 12 and frequency 5 rad/sec the rate saturated outputs for the two structures are identical.

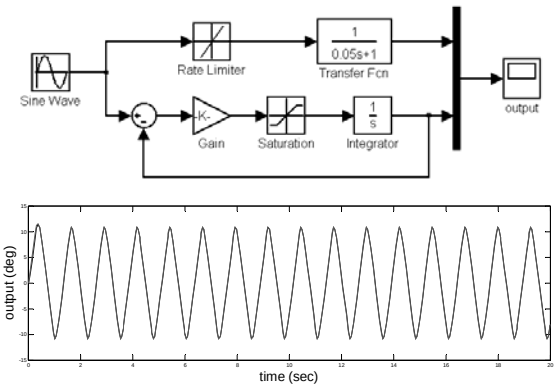


Fig. 3. Equivalent representation of a rate limited actuator; time response for the saturated case

As mentioned before, Category II PIO are mainly determined by rate and position saturations. Considering for example a position saturation, the behavior of this nonlinearity is equivalent with a linear unknown gain L, as illustrated in figure 4 (for details see [3]). In this figure $\psi_{\max}$ is the maximum output amplitude; $u_{\max}$ denotes the maximum input amplitude and $u_T = \psi_{\max}$ is the linear threshold in input. A large value of the predicted maximal command $u_{\max}$ corresponds to a small value of $L_{\min}$,

therefore $L \in [L_{\min}, 1]$.

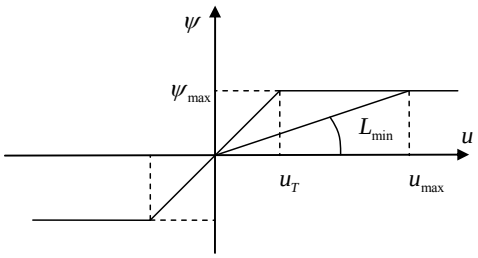


Fig. 4. Saturation non linear characteristic

With this representation we can define the model suited for robust analysis. In figure 5 three uncertain parameters will be considered: L_1 corresponding to the rate limited actuator, L_2 corresponding to the stick limits and K_p the pilot gain. The PIO detection implies to determine the triplets (L_1, L_2, K_p) separating the stability and instability regions.

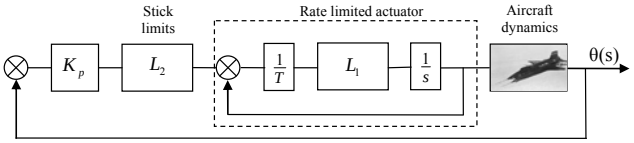


Fig. 5. Robust stability analysis block diagram

III. Robust Stability Method with Time Invariant Uncertainties

In this section a method to perform the robust stability analysis of a linear time-invariant system subject to parametric time-invariant uncertainties is presented. This method to determine the maximal domain (L_1, L_2, K_p) for which the resulting system is stable is based on the Edge Theorem [6]. The notations and definitions used to state this result are briefly presented bellow. Consider the family of n-degree polynomials:

$$P(s,\delta) = a_0(\delta) + a_1(\delta)s + \cdots + a_n(\delta)s^n \qquad (1)$$

where δ is an m-dimensional vector of uncertain parameters. Assuming that δ_i are independent and $\delta_i \in [\underline{\delta}_i, \overline{\delta}_i]$ where $\underline{\delta}_i, \overline{\delta}_i, i = 1,\ldots,m$ are given, it follows that δ_i lies in an m-dimensional box D.

$$D = \left\{ \delta \in \mathbb{R}^m \mid \delta \in \left[\underline{\delta}_1, \overline{\delta}_1\right] \times \ldots \times \left[\underline{\delta}_m, \overline{\delta}_m\right] \right\} \qquad (2)$$

If the polynomial coefficients $a_k, k = 0,\ldots,n$ are affine functions of $\delta_i, i = 1,\ldots,m$ then:

$$P(D) = \left\{ p(s) = \sum_{i=1}^{2^m} \lambda_i P_i(s),\ \lambda_i \geq 0,\ \sum_{i=1}^{2^m} \lambda_i = 1 \right\} \qquad (3)$$

is the *polytope of polynomials.* The *vertices of the polytope* $p_i(s), i = 1 \ldots 2^m$ are obtained by replacing in (1) the parameters $\delta_i, i = 1,\ldots,m$ with their extreme values, in all 2^m possible modes. By $E_{ij}(s,\lambda), i, j = 1,\ldots,2^m, i \neq j$ are denoted the *edges of the polytope P(D)*, defined as:

$$E_{ij}(s,\lambda) = \left\{ p(s) = \lambda p_i(s) + (1-\lambda) p_j(s),\ \lambda \in [0,1] \right\} \qquad (4)$$

Definition 1. If $\mathcal{D}$ is a domain in the complex plane, the family of polynomials are called $\mathcal{D}$-*stable* if their roots lie within $\mathcal{D}$.

Theorem 1. *(Edge Theorem [6]). The polynomial family (1) with affine coefficient functions $a_i(\delta)$ is $\mathcal{D}$-stable if and only if the edges of the polytope $P(D)$ are $\mathcal{D}$-stable,* w*here $\mathcal{D}$ is a simply connected domain in the complex plane.*□

The above result provides necessary and sufficient $\mathcal{D}$-stability conditions and therefore the results obtained using it are not conservative. The Edge Theorem can be also used in the case when the polynomial coefficients are multilinear functions of the uncertain parameters but in this case conditions in the statement are sufficient. A possibility to reduce the conservativeness arrising in the multilinear case can be found in [11] where an incremental polytope expanding technique is presented.

IV. PIO Detection Based on Popov Criterion with Multiple Nonlinearities

A drawback of the above method is that it assumes that the uncertainties are time-constant which is not the case in the above representation of L. Indeed, one can directly check that the uncertain gain L does not accomplish the condition for time invariance $S_T L = L S_T \ \forall T > 0$, where S_T is the shift operator defined as $S_T f(t) = 0, \forall t < T$ and $S_T f(t) = f(t-T), \forall t \geq T$. Therefore the parameters L_1 and L_2 are time-varying, which in fact implies that the stability domain can be smaller than the one obtained in the previous section. An alternative approach to overcome the limitations of the previous analysis method is based on the Popov criterion, considering the nonlinear saturation elements as in figure 2.

In [8] a linear matrix inequalities (LMI) version of the Popov criterion for multiple nonlinearities can be found. It is interesting to note that according with [8], in the case with a single nonlinearity the LMI feasibility provides necessary and sufficient absolute stability conditions. In the case of multiple nonlinearities as in our application these conditions are just sufficient.

In the following, the LMI version of the Popov criterion with multiple nonlinearities will be recalled. Consider the Lur'e system:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \qquad (5)$$

with $x \in \mathbb{R}^n, u, y \in \mathbb{R}^m$, where $u_i(t) = \psi_i(y_i(t))$, $i = 1,\ldots,m, \psi_i$ denoting $[0,1]$, sector-type nonlinearities

satisfying the conditions $0 \le \sigma\psi_i(\sigma) \le \sigma^2, i = 1,...,m$. Then using the Lyapunov function of the form

$$V(x) = x^T Px + 2\sum_{i=1}^{m} \lambda_i \int_0^{C_i x} \psi_i(s)ds \qquad (6)$$

where C_i stands for the i-th row of C, together with S-procedure based arguments, in [8] it is proved that (5) is absolutely stable if there exists $P > 0$, $\Lambda = diag(\lambda_1,\ldots,\lambda_m) \ge 0$ and $\Upsilon = diag(\tau_1,\ldots\tau_m) \ge 0$ such that

$$\begin{bmatrix} A^T P + PA & PB + A^T C^T \Lambda + C^T \Upsilon \\ B^T P + \Lambda CA + \Upsilon C & \Lambda CB + B^T C^T \Lambda - 2\Upsilon \end{bmatrix} < 0. \qquad (7)$$

In the next section a case study with two nonlinearities will be presented. The results obtained with the method described in Section III and with the above LMI-version of the Popov criterion will be comparatively analyzed.

V. A CASE STUDY

This section considers the PIO of X-15 occurred during a landing flare with the pitch SAS off. For this flight the control surface rate was limited to 15 deg/sec. Besides the nonlinear element of the rate limited actuator a saturation block representing the stick limits is added. Figure 5 shows the block diagram of this system.

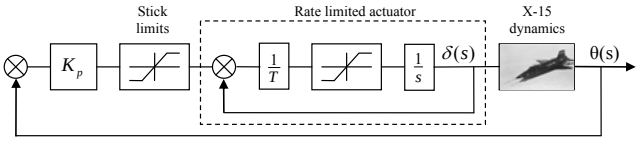


Fig. 6. X-15 block diagram

The numerical values of the elements in the block diagram are:

$$\frac{\theta(s)}{\delta(s)} = \frac{3.476(s+0.0292)(s+0.883)}{(s^2+0.019s+0.01)(s^2+0.841s+5.29)}$$

$$T = 0.04\,\text{sec}$$

The characteristic polynomial of the closed loop system in figure 5 is:

$$\begin{aligned} P(s) = {} & s^5 + (25L_1 + 0.86)s^4 + (21.52L_1 + 5.31)s^3 + \\ & + (132.9L_1 + 86.9K_p L_1 L_2 + 0.1)s^2 + \\ & + (2.72L_1 + 79.27K_p L_1 L_2 + 0.05)s + \\ & + 1.32L_1 + 2.24K_p L_1 L_2. \end{aligned}$$

In Figure 7 the result of robust stability analysis is presented for the case when the saturation block corresponding to the stick limits has no effect i.e. $L_2 = 1$.

The robustness versus time-invariant uncertain parameters is characterized by the curve (S) and the analysis versus time-varying parameters obtained with Popov stability by (P).

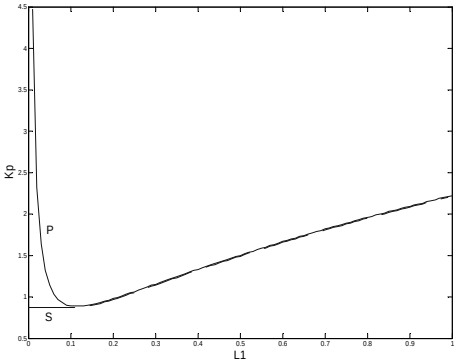


Fig. 7. Stability boundary for constant and time-varying case $L_2 = 1$

As noted in [16] and [21] the above result is not the same as that in [3]. The method used in [3] to perform robust stability analysis of the system with time invariant uncertainty is called ROBAN.

Considering all three uncertain parameters, using the approach proposed in section III the surface from figure 8 is obtained. This surface delimits the stability region (below the surface) from the unstable region (above).

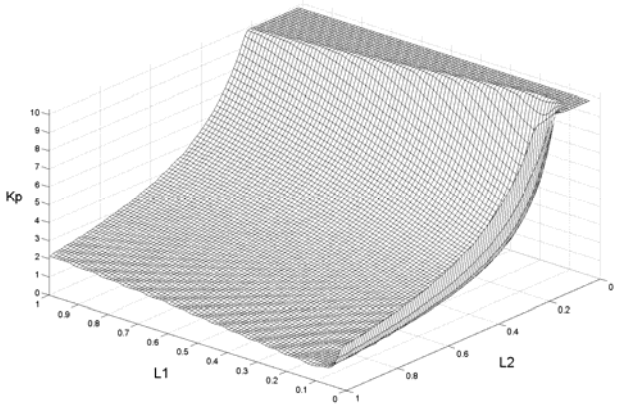


Fig. 8. Stability boundary for the time invariant case

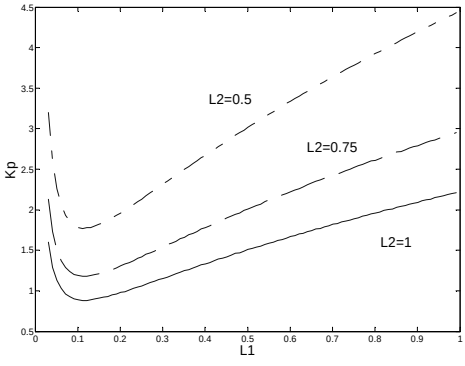


Fig. 9. Stability boundary with L_2 constant

In figure 9 the stability boundary is plotted for fixed values of L_2. From figure 8 and 9 we can establish that a tight stick limit has a stabilizing effect on the pilot vehicle system.

The time simulation results at four test points are presented in figure 10. In figure 10(d) a PIO occurrence

can be observed.

Case (a): $K_p = 2, L_2 = 0.75, L_1 = 0.7;$

Case (b): $K_p = 3, L_2 = 0.75, L_1 = 0.7;$

Case (c): $K_p = 2, L_2 = 0.5, L_1 = 0.4;$

Case (d): $K_p = 1.81, L_2 = 0.75, L_1 = 0.41.$

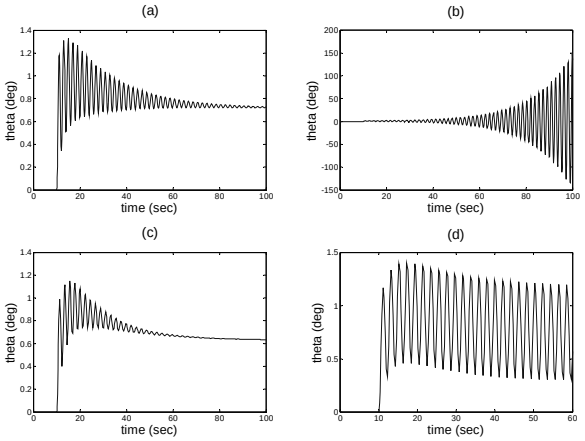


Fig. 10. Time response of the pitch angle

For the time varying case the stability boundary is presented in figure 11. This analysis is made considering that the uncertain parameters at each point can have any variation in the following domains:

$L_1 \in [L_{1\min}, 1]; L_2 \in [L_{2\min}, 1]; K_p \in [K_p, K_p + 0.01].$

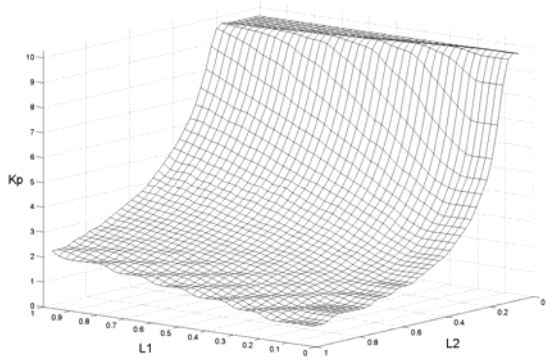


Fig. 11. Stability boundary for the time varying case

VI. CONCLUSIONS

Two methods to analyze the PIO occurrence determined by position and rate saturation were presented in this paper. The first one is based on robust stability method in which the nonlinearities are regarded as parametric uncertainties. The stability domain is determined using the Edge theorem in multilinear form. The second approach is based on Popov criterion with multiple nonlinearities in LMI form. Numerical results show that the stability domain determined using the second method is smaller than the one obtained by the Edge theorem. This is due to the fact that in the first case the time-varying character of the uncertain parameters replacing the saturations was ignored. Based on Popov criterion version for multiple nonlinearities, the case when the pilot delay is taken into account will be investigated in the forthcoming papers.

REFERENCES

[1] J. Ackermann, "Robust Control Systems with Uncertain Physical Parameters", Springer-Verlag 1993.

[2] F. Amato, "Robust Control of Linear Systems Subject to Uncertain Time-Varying Parameters", Lecture Notes in Control and Information Sciences; Springer 2006.

[3] F. Amato, R. Iervolino, S. Scala and L. Verde, "Actuator Design for Aircraft Robustness Versus Category II PIO", *Proceedings of the 7th Mediterranean Conference on Control and Automation, Haifa, Israel* (1999), pp.1804-1820.

[4] F. Amato, R. Iervolino, S. Scala and L. Verde, "Category II Pilot-in-the-Loop Oscillations Analysis from Robust Stability Methods", *Journal of Guidance, Control and Dynamics*, Vol. 24, No 3, May-June 2001.

[5] R. E. Bailey, T. J. Bidlack, "A Quantitative Criterion for Pilot-Induced Oscillations: Time Domain Neal-Smith Criterion", AIAA-96-3434-CP, 1995.

[6] B. Barmish, "Stabilization of uncertain systems via linear control", *IEEE Transactions on Automatic Control*, Volume 28, Issue 8, 1983.

[7] A. C. Bartlett, C. V. Hollot and Huang Lin, "Root Locations of an Entire Polytope of Polynomials: It Suffices to Check the Edges", *Math. Control Signals Systems (1988)*.

[8] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, "Linear Matrix Inequalities in System and Control Theory", Society for Industrial and Applied Mathematics (SIAM), 1994.

[9] D. Ossmann, M. Heller, O. Brieger, "Enhancement of the Nonlinear OLOP-PIO-Criterion Regarding Phase-Compensated Rate Limiters", *AIAA Atmospheric Flight Mechanics Conference and Exhibit, Honolulu, Hawaii*, Aug. 18-21, 2008.

[10] H. Duda, "Prediction of Pilot-in-the-Loop Oscillations due to Rate Saturation", *Journal of Guidance, Navigation, and Control,* Vol. 20, No. 3, May-June 1997.

[11] O. Ekdal, B-C Chang "Robust Stability Analysis of Real Structured Uncertain Systems" *Proceedings of the 11th IFAC World Congress*, Aug. 1990.

[12] P. Gahinet, A. Nemirovski, A. Laub, and M. Chilali, "LMI Control toolbox user's guide". The MathWorks, inc, 1995.

[13] K. H. Khalil, "Nonlinear Systems", Mac Millan, New York 1992.

[14] B. S. Liebst, M. J. Chapa, D.B. Leggett, "Nonlinear Prefilter to Prevent Pilot-Induced Oscillations due to Actuator Rate Limiting", *Journal of Guidance, Control and Dynamics*, vol.25, no.4, July-August 2002.

[15] D. T. McRuer, D. H. Klyde, and T. T. Myers, "Development of a Comprehensive PIO Theory", *AIAA paper 96-3433*, (1996), pp. 581-597.

[16] V. Pana, "Prediction of Pilot Induced Oscillations", *The 31-th Internationally Attended Scientific Conference of the Military Technical Academy*, 2005, Bucharest.

[17] L. Rundqwist, R. Hillgren "Phase compensation of rate limiters in JAS 39 Gripen",*AIAA Atmospheric Flight Mechanics Conference, San Diego, CA*, July 29-31, 1996.

[18] R. E. Skelton, , T. Iwasaki and K. Grigoriades "A Unified Algebraic Aproach to Linear Control Design", Taylor and Francis 1998.

[19] J. Sofrony, M. C. Turner, I. Postlethwaite, O. Brieger, D. Leissling, "Anti-windup synthesis for PIO avoidance in an experimental aircraft", 45th *IEEE Conference on Decision and Control*, 2006.

[20] A. M. Stoica "State Feedback Q-Stabilization with Robust H^∞ Performance for Systems with LFT Based Parametric Uncertainty", *Journal of Control Engineering and Applied Informatics*, CEAI, Vol. 7, No. 2, pp. 3-9, 2005.

[21] B. F. Wu and J. W. Perng "Gain–Phase Margin Analysis of Pilot-Induced Oscillations for Limit-Cycle Prediction", *Journal of Guidance, Control and Dynamics*, Vol. 27, No. 1, January–February 2004.

A multiagent based solution for mobile robots path planning

D. Panescu, M. Kloetzer, A. Burlacu, and C. Pascal

***Abstract*—The purpose of this paper is to propose a planning and coordination scheme for a system with two mobile robots. The task to be solved by the robots regards their movement in a partially known environment so that they meet in the shortest time. As such an application has a distributed nature, a multiagent architecture can be an appropriate solution. Moreover, an agent based approach creates good premises to obtain the needed coordination and reactivity. The designed scheme contains two agents dedicated to the robots, which apply a heuristic search to find the plan for the robots' movement. A suitable coordination protocol and an agent proper structure allow the path generation even when the environment is changing when the robots are moving. The correct system operation has been proved by simulation experiments.**

I. Introduction.

THE multiagent systems (MASs) represent a growing research area and their applications include more and more fields. Robotics can benefit from the new agent based control schemes. If a multi-robot system is involved then the need of coordination becomes obvious and the connection with an MAS turns out to be a practical solution [1], [2]. This contribution takes into account the case of a system with two mobile robots. The task to be solved regards the robots' movement in a partially known environment so that they should meet as soon as possible. Such a scenario may appear both in an industrial environment (two mobile robots that must transfer a part or a tool) and in other types of situations (for example, in exploring or rescue robots' activities) [3], [4]. The paper shows that by considering the two robots as components of an MAS an efficient and optimal solution can be obtained, even in a dynamic environment.

The problem specifications are as follows. The map of the robot environment is known and the two robots initial positions, too. The plane area where the two robots can move contains locations interconnected by several ways. The robots have to find the optimal route to traverse so that they meet in the shortest time, which obviously conducts to determining the shortest path. The difficult aspect is that though the map is known in advance, some changes can appear, namely it may happen that some new obstacles block certain ways. Thus the robots start with a planned path and then they have to adapt it when a way is obstructed by an object that was not considered in the initial map. Our hypothesis is that such an obstacle will be detected by the robots' sensorial system (e.g., by using ultrasonic sensors in an approach as the one explained in [5]) from the beginning of a way (namely, from the moment when the robot is in an intersection), so that it will be able to find an alternative path.

As about the agent based approach, because the on-board computing resources of a mobile robot are often limited, the proposed solution is to have an external computer to run the agents dedicated to the two robots. This means each robot has as its high level decision system an agent. The two agents can communicate one with the other and with the robot under control. If the task to be solved becomes more complicated and further reasoning abilities are needed, then a solution with the two agents running on two interconnected computers may be also taken into account, as shown in Fig. 1. The paper is organized as follows. First the devised coordination protocol is explained, then some details regarding the searching mechanism are being provided, followed by a brief description on the agents' design and implementation; some comments on the performed experiments and a few conclusions end this contribution.

II. The Coordination Protocol

The robots' operation comprises two main parts: planning and execution. These must be interleaved in a specific manner, so that the imposed performance is obtained. The planning part is solved by the MAS, applying a distributed search; then,

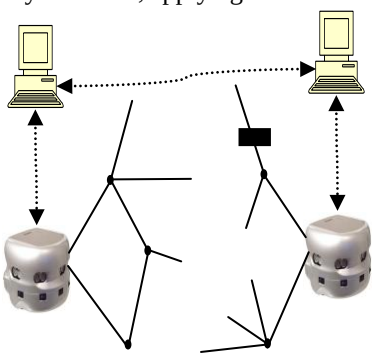


Fig. 1. A multi-agent architecture to solve the navigation for two mobile robots

Manuscript received on May 15, 2010. Part of this work was supported by the research project SOFHICOR, granted by the Romanian Ministry of Education, the National Council of Research, Contract no. 11-042/2007.

D. Panescu is with the Department of Automatic Control and Applied Informatics, "Gheorghe Asachi" Technical University of Iasi, Romania, Str. Prof. dr. doc. D. Mangeron 27, IS 700050 RO (corresponding author, phone: +40 232 230751, e-mail: dorup@tuiasi.ro).

M. Kloetzer is with the Department of Automatic Control and Applied Informatics, "Gheorghe Asachi" Technical University of Iasi, Romania, Str. Prof. dr. doc. D. Mangeron 27, IS 700050 RO (e-mail: kmarius@tuiasi.ro).

A. Burlacu is with the Department of Automatic Control and Applied Informatics, "Gheorghe Asachi" Technical University of Iasi, Romania, Str. Prof. dr. doc. D. Mangeron 27, IS 700050 RO (e-mail: aburlacu@tuiasi.ro).

C. Pascal is with the Department of Automatic Control and Applied Informatics, "Gheorghe Asachi" Technical University of Iasi, Romania, Str. Prof. dr. doc. D. Mangeron 27, IS 700050 RO (e-mail: cpascal@tuiasi.ro).

each agent knows the solution and can send to the corresponding mobile robot commands regarding the execution of the necessary movements; they come back to planning when a change in their environment is detected. At a first decomposition level the agents' activity is conducted according to the following scheme:

Phase 1. Receive the initial robots' position.
Phase 2. Plan a whole path to obtain the robots' meeting.
Phase 3. Launch the execution of the planned path.
Phase 4. Receive information from the corresponding robot; if the goal position is to be reached in the next step then an approaching command is sent to the robot, then the mission is ended; if the sensorial information regards a blocked way, then go to Phase 1.

The above cycle is explained as follows. Based on the initial information on the robots' positions and the environment map an appropriate searching algorithm is carried out to find the solution for the robots' movement. Our proposal is for using the A* heuristic search in a distributed manner, because it offers certain advantages. The method provides both completeness and optimality; moreover an optimally efficient behavior is obtained [6], [7].

After the Phase 2 the two agents will know the entire optimal succession of ways connecting their initial positions (of course, one may exclude the case when no solution is possible). In the above description Phases 3 and 4 appear in sequence, but in fact they will be interleaved. This means an agent sends towards its robot a specific command depending on the robot acquired sensorial information. Namely, when the planned way is obstructed the mobile robot control system sends this information to its agent and the entire cycle is re-started through the agents' coordination mechanism. Meanwhile, the two agents can detect the moment when the two robots are supposed to meet and thus they will inform the robots to apply a specific approaching procedure, so that they should come nearby without collision. To better clarify how the Phases 3 and 4 operate, further details are being provided, explaining the agent coordination protocol. Let us note P_i the current position of the first robot and respectively P_j the present position for the second robot, while P_{i+1} and P_{j+1} are the next positions of the two robots. The proposed protocol contains the following steps:

Step 1. If the agent received a message for re-planning from the other agent then it replies by sending to its partner the current position of its controlled robot (P_i) and then go to Phase 2.

Step 2. If $P_i = P_{j+1}$ then send to the robot the command to apply the approaching procedure and the cycle is ended.

Step 3. Send a message to the robot to check the way from P_i to P_{i+1} together with the other ways connected to P_i and wait for the robot's answer.

Step 4. If the way P_i to P_{i+1} is blocked then send to the partner agent the message asking a re-planning phase. Then go to Phase 2.

Step 5. If the way P_i to P_{i+1} is not blocked then test for $P_{i+1} = P_{j+1}$. If this condition holds
then:
- send to the robot the command to go to P_{i+1}, to apply the approaching procedure and the cycle is ended;
- send P_{i+1} to the partner agent, as the next position.

else:
- send to the robot the command to go to P_{i+1}
- send P_{i+1} to the partner agent, as the next position.

Step 6. The agent waits for the information from its robot regarding the reach of the commanded position. When this happens, P_i becomes P_{i+1} and go to Step 1.

These steps regard the phases 3 and 4 in the above presented scheme and the connection with the phase 2, when needed. It is to remark that an agent based implementation overcomes a purely sequential operation. Namely, it is considered a BDI agent architecture (as it can be obtained by making use of the Jack agent development environment [8], [9]). In an agent based implementation the execution is an event driven one [10]. This supposes that an agent is waiting for events, mainly these meaning messages received from its environment. As already told, in our case an agent may receive messages from the other agent and from the controlled robot. These messages can asynchronously arrive and they will be kept in a queue. So, even if a message is not received during the waiting state of the Step 6, it will be kept and accordingly used, either in the Step 1 or in the Steps 2 and 5, so that the agent can decide on the necessity of a re-planning stage, a present or next approaching procedure. Regarding the Steps 2 and 5, they are devoted to allow a smooth robots' meeting without the need to a priori establish this moment. Thus, the robots can have different speeds and the coordination procedure allows a correct robots' approaching.

It is important to understand how the re-planning phase is launched and operates. It is started as soon as a robot detects an obstructed way. In that moment the corresponding agent sends to its partner both the present robot location (P_i) and the way that is blocked ($P_i - P_{i+1}$). According to the Step 1 (in fact according to the plan being attached to the agent re-planning event), the agent has to further receive the present position of the other robot (P_j). Thus all the information for applying a new A* based search is available (the two robots initial positions and the up-dated map with the previously open way being deleted). As soon as a new entire path is found, this is sent for execution to the two agents, coming back to the Phase 3. It is to remark that when the re-planning phase is asked, the agents exchange information on all the environment changes produced since the last updating.

Even during an execution without any obstacle appearance the agents send messages to each other, in order to be informed about the next positions to be reached, according to

the Step 5. Thus, there is no need for other feedback so that the robot should have a meeting without collision. One can easily show that this holds in the proposed procedure for all the paths containing at least three segments.

III. The Algorithm to Find the Optimal Path

As already mentioned, a heuristic search was used to get the plan of the robots' movement, namely the A* algorithm. As the considered application contains two agents an obvious construction would be to apply a distributed approach, like the one offered by the bidirectional search [6], [7]. The problem is that when applying A* in a bidirectional search, the performance of the method is highly dependent on the heuristic function depression. Namely, for a path planning problem (a case for which the difference between the values of the heuristic function for the successors of a node can be high) when trying to get an efficient bidirectional search, the performance in the combined problem space is worse than when using the initial problem space [7]. That is why the proposed approach is to avoid the drawbacks of making the search in the combined problem space, but to further benefit by the use of a distributed bidirectional algorithm. Thus, the two agents know the initial and the goal states (according to the initial positions of the two robots) and apply A* with the corresponding initial data: the initial state for an agent is the goal state for the other one. The goal in each of the two searches is fixed and thus the initial problem space is kept, while the ending condition regards the moment when one agent finds a position that is already within the solution of the other agent. Some details of the constructed search procedure are further presented.

The devised bidirectional A* search relies on the typical A* routine for finding successor nodes and it is implemented by each of the two agents by following the next protocol:

Agent 1:

(start node = initial position of Agent 1;
goal node = initial position of Agent 2)
1. Use the typical A* strategy to choose a successor node (denoted by P_1)
2. Update the current path (best path to P_1)
3. Receive from Agent 2 its current path (denoted by $Path_2$)
4. IF P_1 belongs to $Path_2$
   - construct the whole path by combining the current path to P_1 with $Path_2$
   - send the whole path to Agent 2
   - exit the search algorithm and begin movement
   ELSE
   - go to 1
   END IF

Agent 2:

(start node = initial position of Agent 2;
goal node = initial position of Agent 1)
1. Use the typical A* strategy to choose a successor node (denoted by P_2)
2. Update the current path (best path to P_2) and send it to Agent 1
3. IF Agent 1 sends the whole path
   - exit the search algorithm and begin movement
   ELSE
   - go to 1
   END IF

Such a searching approach benefits from the complexity reduction determined by the bidirectional search [6]. In our case this is coupled with the fact that the two agents can work simultaneously and exchange messages to detect the moment when the solution was reached, which additional reduces the period for the solution reaching; of course in this case there is a time spent with the communication phase.

IV. On the Agents' Design and Implementation

The two agents are carried out in the JACK programming environment and Fig. 2 represents their planning and execution design diagram [10], [11]. The common agent structure is represented with continuous lines, while the entities specific for the Agent 1 are marked with discontinuous line, and the elements that appear only in the case of Agent 2 are represented with dotted line. As already mentioned, an agent has an operation driven by events. The reception of a message represents an external event (*ExternalMsg*) that is treated by a corresponding plan, found in accordance with the message contents and the BDI mechanism. There will be internal events too, the ones that an agent is using to drive its actvities, like the *Planning* and *Execution BDI Events* in Fig. 2.

Three types of messages are used by agents and these determine the launching of corresponding plans:

- messages regarding the need of a re-planning phase, containing as parameters the present robot position and the blocked way or ways (the corresponding plan is labeled *Re_PlanningMsg* in Fig. 2).
- messages concerning the present robot position, that are used when the partner agent has asked re-planning (these activate the plan called *PresentPositionMsg* in Fig. 2).
- the third type of messages regards the path planning phase and is different for the agent that is supposed to assemble the plan of the path – Agent 1, and respectively the other agent, called Agent 2. The Agent 1 sends a message after a plan was found for the entire path and the message contains this path (the plan *PathMsg* handles the message); Agent 2 sends messages containing the positions that it has already planned (such messages are handled by the plan *NextPositionMsg*)

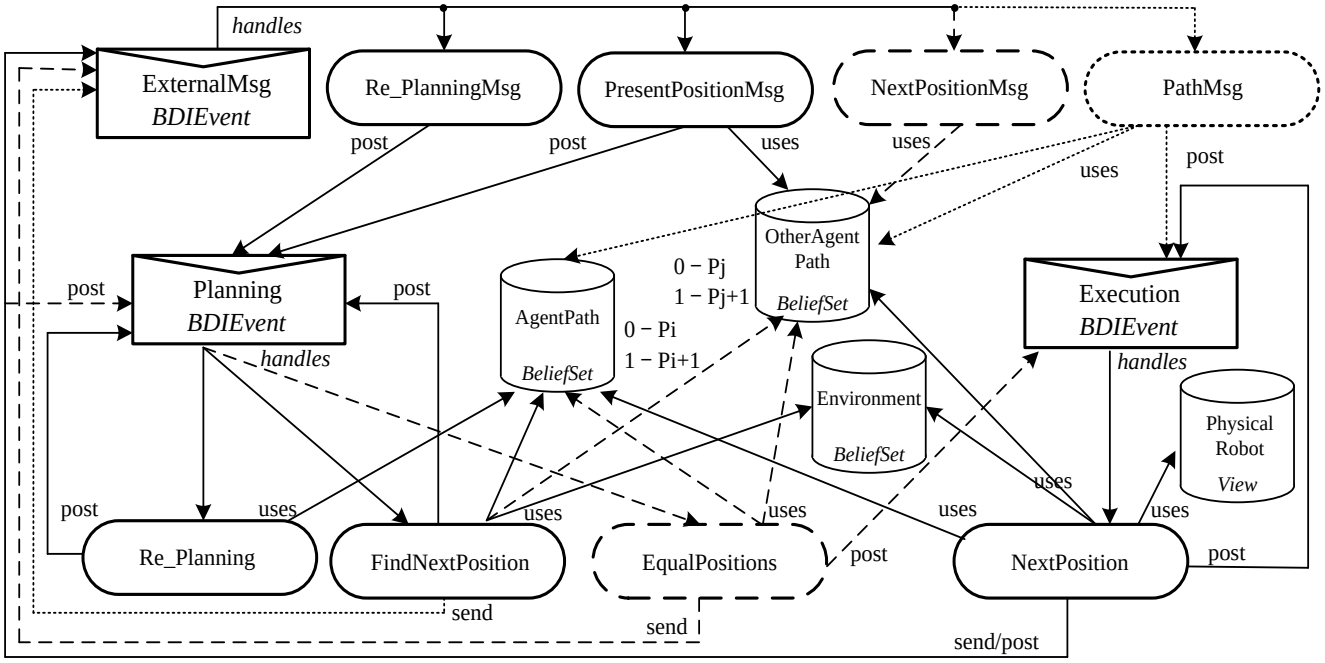


Fig. 2. The agents' planning and execution design diagram

At the initial time, the planning process is started after each agent received the other agent's robot position. Through the *PresentPositionMsg* plan the *Planning BDI Event* is posted as an internal message. This activates the *FindNextPosition* Plan that materializes the corresponding searching algorithm, as described in the previous section. The agent's knowledge is kept in three entities, named *BeliefSets* for JACK agents. Thus, the *AgentPath* beliefset stores the information on the path for the commanded robot. In the planning phase this knowledge base is filled in with the positions found through the A* algorithm, while during execution it keeps the track of the robot's movements. The *OtherAgentPath* beliefset contains the current position and the path planned by the other agent. The third knowledge base, the *Environment* beliefset holds the information about the map of the robots' environment.

All these belief sets are used by the *FindNextPosition* plan to determine the next robot planned position. This plan contains an additional action for the Agent 2, namely the one for sending a message regarding the next planned position towards the Agent 1 (this appears with dotted line in Fig. 2). The last action of the *FindNextPosition* plan posts a new *PlanningBDIEvent* message, thus it results a loop (each iteration corresponds with an application of the algorithm presented in the previous section). This loop is ended when the Agent 1 determines a coincidence between two positions within the belief sets *AgentPath* and *OtherAgentPath*. At that moment the *EqualPositions* plan is activated; this can compose the whole path to be traversed by the two robots. After that, the Agent 1 up-dates its belief sets and sends the entire path to the Agent 2 (as an external message).

The *Re_Planning* plan is used after an external message regarding the need of a re-planning phase is received and has as consequence the ceasing of the execution phase when the corresponding robot reaches the next position. After that both agents enter the planning cycle.

The execution phase is started by different events for the two agents. Agent 1 starts its execution phase by launching the *ExecutionBDIEvent* from the *EqualPositions* plan. This event is handled by the *NextPosition* plan, which uses the agent's belief sets to determine each movement. This plan includes the tests within the Steps 2 and 5 of the algorithm presented in Section II. The plan supposes a link with the physical robot by the means of a JACK view entity, which materializes the communication with the robot controller. As with the *FindNextPosition* plan, the *NextPosition* plan, which controls the execution phase, establishes a loop (see the algorithm in Section II). This is ended by one of the following three situations. One case regards the moment when the test for the two robots' positions shows that they met. The second ending condition appears when the test regarding an obstructed way has a positive result (in this case the agent sends an external message to inform the partner agent and to restart its planning phase). The third situation is when according to the *AgentPath* beliefset the next robot position is not available, as a consequence of receiving an external re-planning message from the other agent. The execution phase is similar for the Agent 2, except for its starting condition that is determined by the *PathMsg* plan.

The structure of Fig. 2 conducted the JACK agents' development and these can run as a single process on the same computer, or they can be deployed on distinct networked computers.

V. Experimental Results and Conclusions

This contribution regards an on-going research. The plan is to couple the two agents with two Khepera type mobile robots. Until now only some simulation experiments have been conducted to prove the efficiency and adequacy of the proposed scheme, the performance of the agent based path searching mechanism and of the agents' coordination mechanism.

A sequence of situations obtained within an illustrative scenario is presented in Figs. 3, 4 and 5. One can see in Fig. 3 the initial robots' positions (1 and 38) and the map of their environment. The experiment considers the case when the nodes in the searching graph have one or two successors. The edges' weight is proportional with the ways' length. Thus, in our case the horizontal edges' weight is 1, while the oblique edges have the cost of $\sqrt{2}$, except for the edges that correspond to the ways between positions 5 and 8, 24 and 32 which have the weight of $\sqrt{5}$. The optimal path for this map consists in the following succession of positions: 1 – 2 – 4 – 11 – 17 – 21 – 27 – 33 – 36 – 38. In fact the Agent 1 detected the path from position 1 to 21, and the Agent 2 the path from 38 to 11, and then the node corresponding to the position labeled 21 is detected by the Agent 1 in both agents' paths and the searching processes were ended, the result being the optimal solution.

The image in Fig. 4 shows that in our experiment five ways were successively detected as being obstructed. This is a difficult case, as the two agents have had to re-plan their path three times. Thus, first the robots begin with the path having the minimal cost and when the first robot is in the position 4, both possibilities to continue are being obstructed (see Fig. 4). The optimal path in these new conditions is the one through the positions: 2 – 5 – 9 – 16 – 20 for the first robot (as discovered by the Agent 1), and 36 – 32 – 25 – 20 for the second one. A further blocked way appears (16 – 20) and the agents carry out another planning phase. The reconfigured path is: 9 – 5 – 2 – 1 – 3 – 7 – 13 – 18 – 22 for the first robot, and 25 – 32 – 36 – 38 – 37 – 34 – 28 – 22 for the second one. The two paths are traversed until the first robot is in the position 7 and the second robot is in the position 34. By that time, the second robot detects the two ways blocked and asks for the re-planning phase. The agents are able to find a new solution, this being: 7 – 3 – 6 – 14 – 19 – 23 for the first robot, and 34 – 37 – 35 – 30 – 23 for the second one, which is again the optimal one for the current environment. By making use of this plan the robots successfully meet in position 23, as shown in Fig. 5 obtained in our simulation.

Even the distributed solution has visible advantages in the common cases, it may appear that when a large number of blocked ways occurs the multi-robot solution is not better than a single robot one. Such a case is presented in Fig. 6. Positions 1 and 5 are the robots' initial places. The scenario involves several movement attempts. First, the optimal routes are 1 – 3 – 4 for the first robot and 5 – 4 for the second one. The robots start the movement on these paths and when reaching the positions 3 and respectively 4, the ways 3 – 4, 3 – 6 and 4 – 6 become obstructed. The new re-planned path is: 3 – 2 – 8 – 6 for the first robot, and 4 – 5 – 6 for the second robot. After carrying out a single movement, when the robots are in the positions 2 and 5, the way between 2 and 8 becomes obstructed. A new reconfigured path is: 2 – 3 – 1 – 8, coupled with 5 – 6 – 8. After two movement steps for the two robots, the first of them detects the blocked way between the positions 1 and 8, while the second one observes the obstructed way between the positions 8 and 9. After a further re-planning, the path 1 – 9 – 7 – 5 is used by the first robot, while the second robot traverses the route 8 – 6 – 5. Thus, the meeting position is the initial place of the second robot. Though in this scenario the distributed solution seems to provide no advantage, it is to

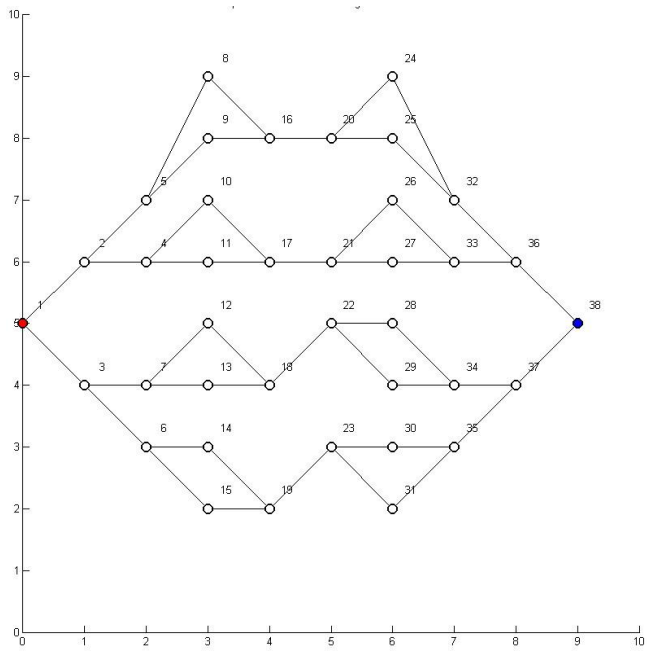


Fig. 3. The environment map and initial robots positions

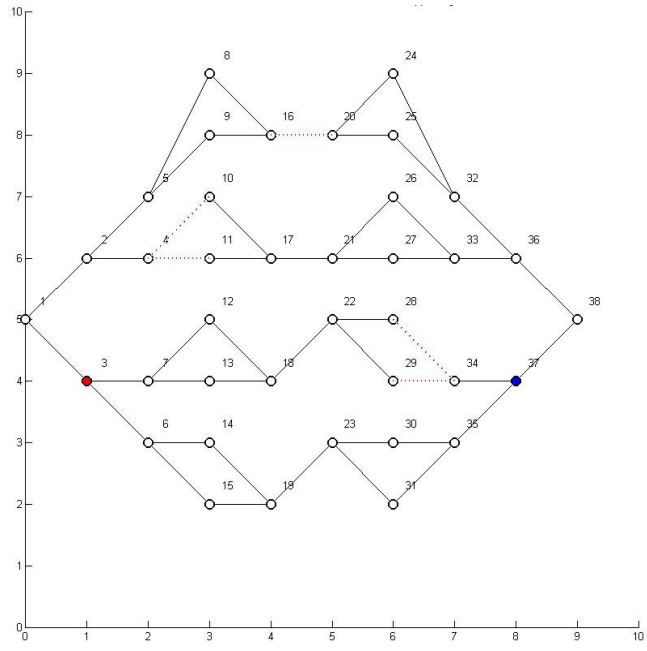


Fig. 4. The environment with several obstructed ways

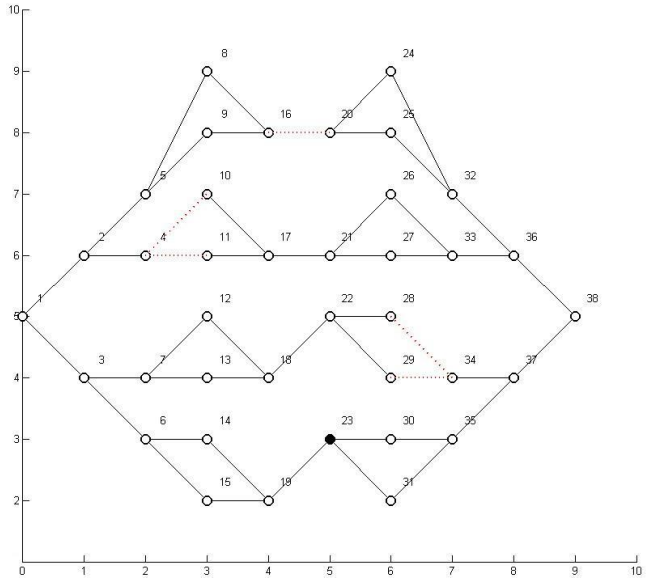


Fig. 5. The robots in the meeting position

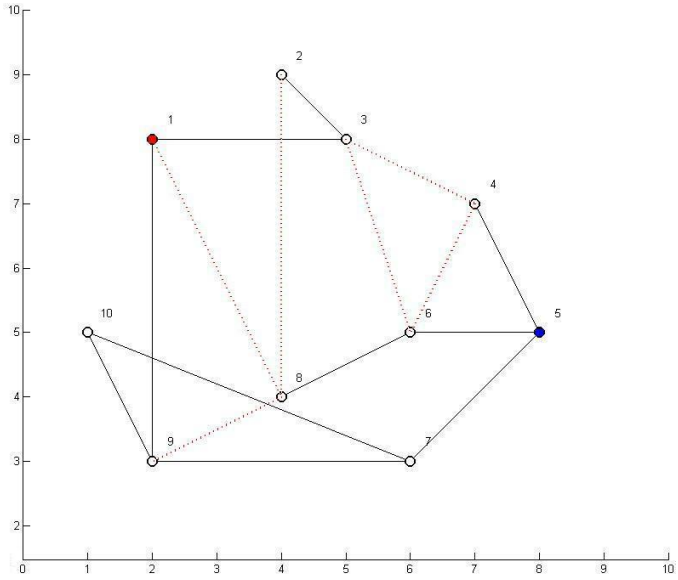


Fig. 6. A case with a large number of blocked ways

remark that the second robot influences a faster reaching of the solution through its sensorial acquisition. Thus, the second robot is the one that detects the obstructed way between the positions 8 and 9, and this information being provided to the first robot' agent allows it to reduce the number of re-planning phases.

In conclusion, though the proposed solution was only partially carried out (the connection with the mobile robots was not achieved) the results obtained by simulation show certain benefits. The way planning and execution are interleaved and the robots are coordinated by means of an MAS allow the system to face a dynamic environment. Even when several changes appear during the robots' movement, their decisional system is able to reconfigure the path in an optimal way. It is to mention that the sensorial system considered for the mobile robots can be a simple one, as only the presence of the obstacles has to be detected. The proposed architecture is a further example on how the Artificial Intelligence techniques can determine an improved solution for Robotics.

The agent based solution has distinct advantages regarding the way a dynamic environment can be handled. An environment modification is producing an event that can be appropriately treated by the attached plans. The BDI agents' reasoning mechanism determines a robust planning scheme because more plans can be devised for the same event; these will be chosen by a suitable filtering scheme, and used one after the other when failures appear. Thus the proposed application can be enhanced, so that the planning phase should be conducted by more decision criteria: if there is no dangerous area find the shortest path, when such an area exists find a path that avoids the robots' approaching the prohibited places.

As future work, two Khepera type robots are to be used, the ways being detected by the infrared sensors mounted under the robots or by the visual sensors endowing such robots. The proposed planning technique must be coupled with an appropriate control algorithm to provide an accurate robot positioning. As about the real-time computation aspects, these are lessened by the fact that execution is launched only after the end of the planning phase and then it is stopped when the re-planning is needed. Because of the way the coordination protocol was conceived no failure due to a delay of a message receiving is possible: a robot starts to traverse a way after checking that the respective way is open and a message received from its partner anytime during this execution period is considered when arriving at the next intersection. A possible improvement of the method is to take into account a further sensorial feedback during the robot movement and in this case the real-time aspects will be harder.

REFERENCES

[1] J. Liu, and J. Wu, *Multi-Agent Robotic Systems*, CRC Press, Boca Raton, 2001, pp. 4 – 17.

[2] H. C.-H. Hsu, and A. Liu, Multiagent-Based Multi-team Formation Control for Mobile Robots, *Journal of Intelligent and Robotic Systems*, **42**, 2005, pp. 337–360.

[3] M. B. Dias, M. Zinck, R. M. Zlot, and A. Stentz, Robust Multirobot Coordination in Dynamic Environments, *IEEE International Conference on Robotics and Automation*, April, 2004, pp. 3435 - 3442.

[4] R. Fiero, et al, A Framework and Architecture for Multi-Robot Coordination, the International Journal of Robotics Research, Oct.-Nov. 2002, pp. 977 – 995.

[5] I. Nagy, Behaviour Study of a Multi-Agent Mobile Robot System during Potential Field Building, *Acta Polytechnica Hungarica*, Vol. 6, No. 4, 2009, pp. 111 – 136.

[6] S. Russell, and P. Norvig, *Artificial Intelligence. A Modern Approach*, Prentice Hall, Upper Saddle, 2003, pp. 96–106.

[7] M. Yokoo, and T. Ishida, "Search Algorithms for Agents", in: *Multiagent Systems. A Modern Approach to Distributed Artificial Intelligence*, G. Weiss, Edit., The MIT Press, Cambridge, 2001, pp. 179-191.

[8] M. Wooldridge, "Intelligent agents", in: *Multiagent Systems. A Modern Approach to Distributed Artificial Intelligence*, G. Weiss, Edit., The MIT Press, Cambridge, 2001, pp. 54-61.

[9] R. Evertsz, *et. al*., "Implementing Industrial Multiagent Systems using Jack™", *PROMAS 2003*, Australia, Springer, 2004, pp.18-48.

[10] L. Padgham, and M. Winikoff, *Developing Intelligent Agent Systems. A Practical Guide*, John Wiley & Sons, Chichester, pp. 8-31, 2004.

[11] JACK™ Intelligent Agents, *Agent Manual*, Agent Oriented Software, Carlton South, Victoria, Australia, 2005.

A Study on the Holons' Interaction for Manufacturing Flexibility

C. Pascal and D. Panescu

***Abstract*—The purpose of this paper is to make a study on the way the interaction between the holons within a Holonic Manufacturing System can influence the manufacturing flexibility and to propose a strategy in order to increase this feature. From different issues on flexibility, the adaptive resource utilization was chosen and more specific the problem of finding the most appropriate part to be used in an operation. This characteristic was called object flexibility and four strategies regarding it are analyzed. They consider an agent based implementation and the experiments developed on some cases involving a product holon and several resource holons allowed us to conclude on the best strategy to use in order to attain a high degree of flexibility.**

I. Introduction. Flexibility and Holonic Manufacturing Systems

ONE of the main characteristics required for the present manufacturing systems regards flexibility. A high degree of flexibility is obtained when the production process can adapt to different types of changes in the manufacturing environment, like the variations of the customers' orders and the production resources' alterations. The Holonic Manufacturing Systems (HMSs) propose a control architecture that creates the means to satisfy all the flexibility criteria. Namely three different types of flexibility can be generally considered in manufacturing [1], and an HMS is able to carry out all of them. Operation flexibility refers to the possibility of performing the same operation on different devices. By adopting a plan-space planning approach [2], which can use partially specified plans, it is possible to obtain operation flexibility. Thus, if one considers the PROSA holonic reference architecture [3], [4] (this was used in the present contribution, too), then at an order or product holon level that uses a partial plan the operation flexibility can be materialized according to the bids the respective holon receives from the resource holons. Sequencing flexibility regards the opportunity to change the sequence of production operations; when such a decision is possible, the HMS is able to carry it out at the product holons' level, again with a suitable planning capability of these holons. At last, processing flexibility is obtained when a manufacturing feature can be attained with alternative operations. By the coordination between the order and product holons, according with the current manufacturing context revealed by the resource holons, this type of flexibility can be reached by an HMS, too. The aim of obtaining the highest possible degree of flexibility is important, as it will be directly reflected into a better manufacturing system performance (just in time operation, balanced use of resources, fault tolerance). Meanwhile when flexibility is the result of a proper control architecture it is expected that the optimal operation can be also gained, resulting in a cost reduction.

It is to notice that all the above mentioned flexibility types are influenced by the way the manufacturing system is able to allocate and use its resources. One of the performance criteria for the HMS should be the choosing of the most suitable object (let it be a raw or semi-processed part) so that the manufacturing cost is minimized. Such cases appear when several parts are available, but their transfer and use in a specified sequence of operations are made with different costs, as they have distinct locations, can be handled by different robots, etc. In the remainder, this issue that influences both the flexibility and optimality of the HMS is analyzed, being named *object flexibility*. Consequently, the paper investigates which is the object flexibility attained when some different interaction strategies are used within the holonic scheme. These strategies differ regarding the efficient HMS operation (as it is reflected in the communication load, computational complexity). Because the distribution of resource operations is decided between the product and resource holons, their interaction is studied; nevertheless, the obtained results can be generalized to the case when the order holons are considered too, as the principles guiding their decisional process are the same as for the product holons. Thus, the purpose of this contribution is to establish an adaptive strategy for the interaction between the product and resource holons, so that a fast operation and flexible use of resources should be gained. The paper is organized as follows. First, the four strategies possible to use for guiding the holons' interaction are being defined. A concise description of the holons' behavior and the cases used for analysis precedes the study on the strategies' performance. This is based on several real life and simulated experiments and the obtained conclusions constitute the last section.

II. Strategies for Obtaining Object Flexibility in an HMS

From the various resources involved in manufacturing an important degree of flexibility is determined by the way the parts and/or semi-products that pass through different processing

Manuscript received on May 15, 2010. This work was supported by the research project SOFHICOR, granted by the Romanian Ministry of Education, the National Council of Research, Contract no. 11-042/2007.

Carlos Pascal is with the Department of Automatic Control and Applied Informatics, "Gheorghe Asachi" Technical University of Iasi, Romania, Str. Prof. dr. doc. D. Mangeron 27, IS 700050 RO (e-mail: cpascal@tuiasi.ro).

Doru Panescu is with the Department of Automatic Control and Applied Informatics, "Gheorghe Asachi" Technical University of Iasi, Romania, Str. Prof. dr. doc. D. Mangeron 27, IS 700050 RO (corresponding author, phone: +40 232 230751, e-mail: dorup@tuiasi.ro).

stages are chosen and used. Focusing our study on this type of resources, it is to determine how the respective objects will be involved in the planning process and in the holons' coordination.

According to the way the production knowledge is shared between different types of holons in the PROSA based HMSs, the product holons are supposed to specify the needed objects' types and their attached operations. Through the relation established between the product and resource holons, the holonic scheme should decide which objects lead to the optimal solution. In this relation each type of holon plays a distinct role. The resource holons may undertake the goal operations defined by the product holons, and certain resource holons must possess the knowledge and capabilities to determine the objects available for each goal (e.g., the sensors on a storage device can determine the number and type of stored parts). Though this holon knowledge specialization is clear, the interaction between holons can change the bias of the decisional process towards the product or resource holons, with consequences on the HMS performance, as it will be further shown.

In agreement with the semi-heterarchic holonic organization [5], the resource holons must be able to perceive three categories of goals, as issued by the other holons (these may be order/product holons or even other resource holons asking for cooperation); these goals' types can be labeled as *reserving*, *operation* and *information acquiring goals*. The first kind refers to the act of reserving (locking) an entity, which can be a location needed for a forthcoming operation (for example, a position in a storage, assembly table where a part is to be placed), or an object needed for a future operation. The second type aims at undertaking a predefined operation such as transport, assembly, processing or product inspection, and the third type appears when a piece of information is only required (e.g. an order/product holon needs to know the number of available parts). Certain relations exist between these categories. For example, before a product assembling it is necessary to know whether the necessary parts exist, then to ensure the location where the operation will take place. Thus, a manager holon – our assumption is that the Contract Net coordination Protocol (CNP) is involved, this being the most used method in HMSs [6], [7] – is supposed to firstly determine the existence of the possible parts and locations, to choose and lock the most appropriate ones and then to manage the proper operations; in such a context all the previously mentioned goal types appear.

Four methods have been considered and are further described with respect to the object flexibility. These are named as: *centralized strategy*, *operation strategy*, *hybrid strategy* and *goal set strategy*. These strategies are being sketched in Fig. 1 and the details on their operation are presented below.

The first strategy is entitled the centralized one because the holons placed higher in the holarchy – the product holons in our case – undertake a central role in collecting the information necessary before deciding, while the other holons placed in lower positions have less decisional influence. Thus, first the product holon sends the reserving goals and after that it proposes operation goals for each of the solutions (reserved objects) obtained for the reserving goals. In this sequence, the bids obtained for an operation goal allow the product holon to choose an available solution; in this way the optimal solution can be reached. This strategy, which brings the holonic approach closer to a classical manufacturing control scheme, is expected to diminish the flexibility on resource utilization.

Taking into account this disadvantage, one may come out with the second strategy by moving the task of the reserving goals' handling at the resource holons' level. In this case the product holons send only the operation goals which explains the strategy name. Each of the two strategies has strong and weak points and therefore it is normal to try to combine them; in this way, a hybrid strategy has been introduced. When the HMS has one or no object that could be used for a manufacturing goal, the centralized strategy is to be used, and for the other cases the operation strategy is preferred. This switching procedure (applied by the hybrid strategy) is determined by the product holons based on the answers received for the information acquiring goals. Thus, this method limits the negative effects of a case with no solution (especially the consequences on the planning process), but it increases the number of goals to be handled with supplementary information acquiring goals.

In each of the previous strategies the optimal solution is the same, but the number of goals changed between holons until finding the result is different. This aspect, important for the HMS efficiency, will be further analyzed. The cause for the increased goal number in the centralized strategy is the fact that for each reserving bid a distinct operation goal is sent by the product holon. An alternative solution is to group the separate operation goals that refer to the same operation but make use of different objects into a composite operation goal. The resource holons will be required to reply with only the best bid for the entire set of goals contained within the composite goal. This is the fourth strategy, named the goal set strategy. It entails new holon capabilities, namely to be able to handle the new goal format and select the appropriate goal entity, the one that can lead to the best solution. The performance of the enumerated strategies is discussed in the following sections.

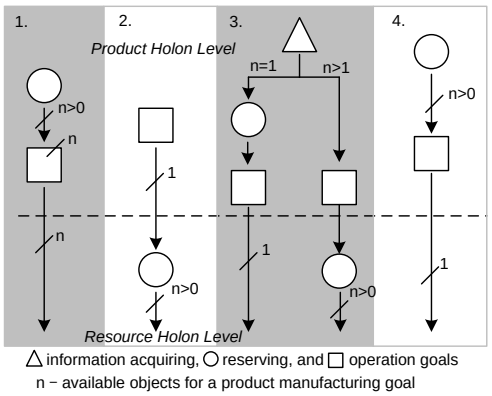


Fig. 1. Goals' succession in the four coordination strategies: 1. centralized strategy, 2. operation strategy, 3. hybrid strategy, 4. goal set strategy.

III. The HMS Organization and the Cases Considered for Analysis

To make a comparison between the strategies described in the previous section, three cases were selected, ranging from the situation when the HMS is able to provide for a manufacturing goal a single solution, two alternative solutions or none. These three cases are determined by the number of bids provided to the product holon for the initiating goal (the operation goal) by the resource holons. The considered testbed is an experimental manufacturing system (see [4] for its detailed description) that includes two industrial robots able to carry out part transfer and assembling operations. Three storage devices are present around the robots: each robot has a storage in its working area, and the third device is placed so that it can be reached by both robots. Thus, some real experiments have been conducted on this manufacturing layout. In order to further analyze the proposed strategies' performance, specific simulation experiments were considered with an increased number of storage devices (complying to the condition that the working area of a robot contains at least one storage). In this way there can be tested the HMS capacity to face the environment changes, which in our experiments meant the presence/absence of the parts and the existence of less or more storage devices. In an HMS the significant physical entities are managed by corresponding resource holons; thus, in our case robot and storage device holons exist. In all the considered scenarios a simple, initial goal is being used, launched by a product holon: a part must be placed in a specified position; this will be named *part transfer goal*.

Besides analyzing the way a strategy is able to find a solution and checking the solution's optimality, the total number of goals (sub-goals) interchanged between holons until finding the solution is measured. This measurement let us have a good overview on the system's performance when the proposed methods are used, as this number gives a measure of the communication load and the complexity of each strategy. To understand how the different strategies work, one has to know the holons' principle of operation. Without going into a detailed description (see also [8]), each holon is composed of a decisional part, carried out as an intelligent agent (the Belief Desire Intention - BDI architecture was used in JACK programming environment implementation [9], [10]) and a physical part in the case of the resource holons (e.g., a robot controller, a PLC based storage device control scheme), or a holarchy for the order/product holons.

Regarding the behavior of the robot resource holons, when receiving a part transfer goal they act according to the CNP. By making use of an initial filtering condition, the holon's agent responds with a negative bid if the position to place the part is not within the robot working area; else, the agent tries to find a solution and compose a bid. In this situation, consistent with the BDI mechanism the section of one of four plans can be done. If the part transfer goal is formulated so that it already contains information about the initial position of the involved part, then two cases are possible, conducting the agent to be a simple contractor or to also become a manager, because it is not able to compose the whole solution by itself. The agent is only contractor if the initial position of the part is within the working area of the corresponding robot. If this condition cannot be satisfied, then the agent takes into consideration the second plan; this implies that the agent launches another part transfer goal so that the part should be moved in its working area by another robot. Depending on the received bids, the robot holon is able or not to compose a solution for the initial goal. When the initiating part transfer goal does not contain the information about the initial part's position, other two plans are involved. First, the agent tries to find the needed part in its robot working area (it issues a *part finding goal* for which it should receive answers from the storage device holons), trying first to compose a solution acting only as a contractor (without applying for the other robot cooperation). If this intention fails, then the agent issues a new part transfer goal trying to compose a solution based on cooperation. In this case it makes use of the fourth plan.

A difference appears in the goal set strategy, namely it needs an additional plan that treats the set of goals' appearance. The plan compares the goals within the set and chooses the one conducting to the best bid that the respective robot can carry out. It implies two phases: first the agent tries to find a solution as single contractor, and then, if this is not possible the agent acts as a manager and resends the entire set of goals to the other robot agent (modifying only the parameter on the final object location), waiting for its decision and bid.

The behavior of a storage device holon that reacts when receiving a part finding goal is simple. If the storage contains the needed part, it proposes a bid including the information about object's location; else, a negative bid constitutes the answer. It is to mention that the object indicated in a bid must be locked. After the contract awarding phase it is unlocked, if the bid was rejected. This mechanism can introduce limitations on the solution space, but this discussion is without the scope of this paper [11].

IV. A Comparison between the Strategies to Obtain Object Flexibility

The analysis starts with the first case (the one with a single solution), and the use of the cooperation plan (the robot holon that composes the solution is a manager, too). The case is illustrated in Fig. 2, which has to be understood as follows. A black filled rectangle represents the demanded part, an empty rectangle with continuous lines indicates the part's specified final location, and a rectangle with dotted lines represents an intermediate position. The goals passed between holons are depicted by arrows, which are oriented from the goal manager to the contractors. A dotted line arrow marks a goal that received a negative bid. The reserving goals for the intermediate and final locations are not represented and also not considered in the analysis because they introduce a constant value. For this experiment, the reserving, operation, and information acquiring type goals are represented by the goals: *Part Finding* (PF), *Part Transfer* (PT), and *Part*

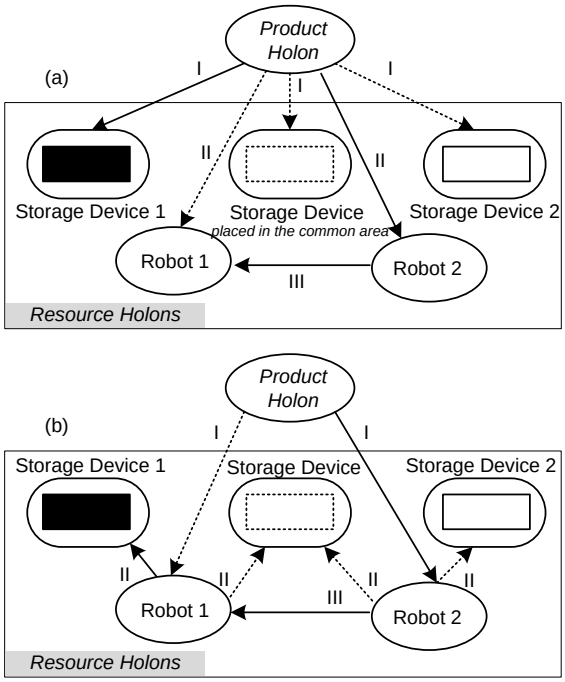


Fig. 2. Holons' interaction scheme using a) centralized strategy and b) operation strategy for the case one (a single solution).

TABLE I
NUMBER OF GOALS REQUIRED FOR THE CASE ONE

	Goal Type	Total Goals	Bids	Negative Bids	New SD RH A	B	C	D
	I – PF	3	1	2	1	-	1	1
1	II – PT	2	1	1	0	-	0	2
	III – SPT	1	1	0	0	-	0	1
	total	**6**			**1**	-	**1**	**4**
	I – PT	2	1	1	0	-	0	0
	II – PF	2	0	0	1	-	0	0
2	III – SPT	1	1	1	0	-	0	0
	IV - SPF	2	1	1	0	-	1	1
	total	**7(6)**			**1**	-	**1**	**1**
	I – PI	3	-	-	1	-	1	1
	II – PF	1	1	0	0	-	0	x
3	III – PT	2	1	1	0	-	0	x
	IV – SPT	1	1	0	0	-	0	x
	total	**7(4)**			**0**	-	**0**	**x**
	I – PF	3	1	2	1	-	1	1
4	II – PT	2	1	1	0	-	0	0
	III - SPT	1	1	0	0	-	0	0
	total	**6**			**1**	-	**1**	**1**

A – a new storage with **no part** added in the robots' **common** area;
B – a new storage with **a part** added in the robots' **common** area;
C – a new storage with **no part** added in a **specific** working area;
D – a new storage with **a part** added in a **specific** working area.

Information (PI), respectively, these notations being used in Table I. To differentiate the same type of goal when it is sent by a product holon, respectively by a resource holon (as sub-goal), the prefix "S" is used to mark the sub-goals. The order of the goals' type is indicated with Roman numbers.

Figs. 2a and b reveal the comparison for the already mentioned case, when the centralized and operation strategies are used. Some quantitative results are presented in Table I, where all four strategies are considered. This is organized in four sections: section 1 regards the centralized strategy, section 2 is for the operation strategy, then the hybrid strategy is considered, and the last section represents the goal set strategy). Table I contains information about the goal types issued in the system, the number of resource holons which received goals and can handle them (the third column), and the number of received bids (the forth column indicates the cases when contractors can find a solution, while the next column regards contractors that can provide no solution). The right part of the table (the last four columns) indicates how much the number of goals increases when a storage device is added in various working areas of the robots, and this storage contains or not the needed object (the legend under the Table I gives the explanation regarding the additional storage device). When a column provides no results (like the column entitled B in Table I) it means that the respective scenario does not correspond to the represented case. For example, if one adds a storage with a part in the robots' common area, then the initiating goal has two solutions, while the analysis is made for one solution problems.

For the hybrid and goal set strategies the interaction diagrams were not presented, because these would be similar to the one of the Fig. 2a. According to the third column in the Table I the total number of handled goals is comparable for the four strategies. An improvement for the operation strategy can be obtained at the robots' resource holon level when a sub-goal for cooperation (an SPT goal) appears. The contractor holon should look for the object only in its specific working area and exclude the common area with the manager holon, because this possibility was already checked at the manager holon level. This enhancement reduces the goal number to 6 and it is further considered for the other cases. For the hybrid strategy two figures on the goal number appear in the Table I. Namely a reduced goal number can be considered (only 4 goals) if the information acquiring goals are not counted. This is justified by the fact that these goals do not suppose a proper planning process and the use of CNP, but only an information retrieving.

The last column in Table I shows a depreciation for the centralized strategy when a storage device with a part is added in the working area of the Robot 1 (see Fig. 2a) which has to be taken into account by the agent that is sub-contractor. This is consistent with the already given explanation, namely the way this strategy has to issue distinct operation goals for each existing object. When considering this case for the hybrid strategy, the goals' order is changed and that is why no results were provided. In fact, the hybrid strategy produces the same number of goals as the operation strategy for this situation.

About the object flexibility, this is better revealed by the second case, i.e. the situation when the HMS is able to provide two solutions for a goal, according to the parts existing in the storage devices. This case is sketched in Figs. 3a and b, and the goals' exchange is summarized in the Table II. This time, the final position of the object is in the common working area of the robots, two parts being located in the specific robots' zones. In the Table II the hybrid strategy was not represented

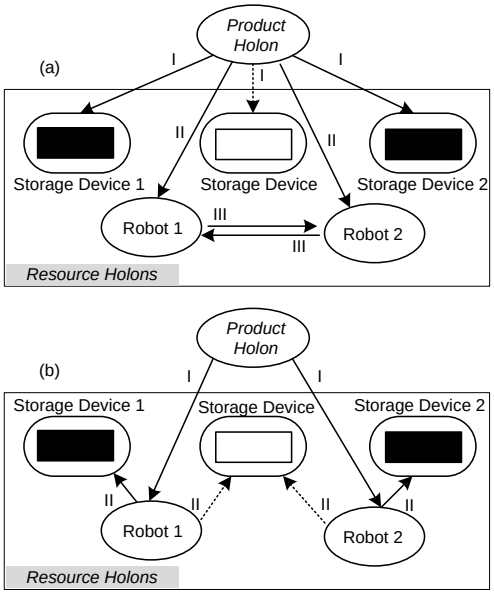


Fig. 3. Holons' interaction scheme using a) centralized strategy and b) operation strategy for the second case (two solutions).

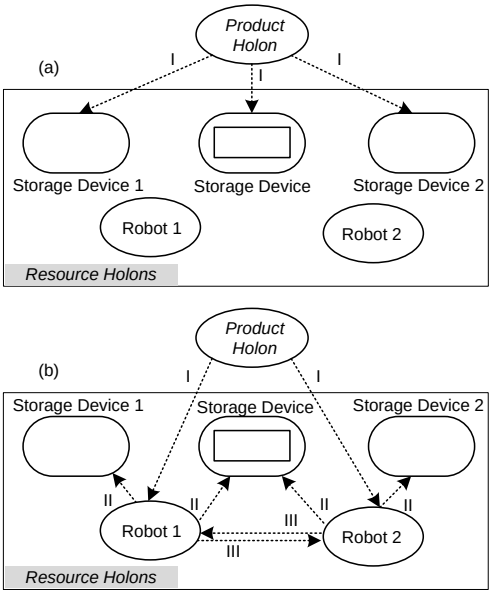


Fig. 4. Holons' interaction scheme using a) centralized strategy and b) operation strategy for the last case (no solution).

because it determines the same results as the operation strategy (no switching appears) and the goal set strategy provides the best results.

The last case when the HMS provides no solution for the proposed goal is illustrated in Fig. 4. Here it can be clearly seen that the operation strategy requires more goals to figure out the result. This is caused by the way each agent tries to become manager and proposes supplementary goals. The goal number difference between the operation strategy and the others can be seen in the Table III. Here the last two strategies were not represented, because they produce the same results as the centralized one. By comparing the last two cases, the following remark is supported. The centralized strategy provides a faster response when the system determines one or no solution (there is a single bid or no bid for the reserving goals), and a delayed result when the object flexibility is possible.

The results presented in the previous tables were completed with a further analysis regarding the modification of the storage device number. This is reflected by the graphs of Figs. 4 a, b and c, corresponding to the three considered cases. The x axis represents the number of storages. It starts from our experimental environment (with three storages) and the labels used for the added storages have the same meaning as in the already presented tables. The y axis represents the number of interchanged goals. In Fig. 5a, the one regarding the case with one solution (a single bid sent to the product holon), it is to remark how the object flexibility appears when a storage containing a part is added in the area of the Robot 1 – the vertical line. At that moment the hybrid strategy switches from the initial use of the centralized strategy to the operation strategy, thus avoiding the increase of the goals' number. Another advantage of the hybrid strategy that can be remarked in Fig. 5a is the way it keeps a constant number of goals when a single object exists. The graph in Fig. 5b reflects the behavior of various strategies in the second case. The object flexibility appears from the initial layout of the manufacturing environment, as the product holon can choose between two solutions. This explains why the hybrid strategy is settled on the operation strategy from the beginning, which is confirmed by the graph. For this case, the goal set strategy uses the minimum number of goals because it groups the operation goals. The last graph for the case with no solution (Fig. 5c) points out that the centralized strategy is necessary in such an extreme case, by avoiding an unsuccessful search at the resource holons' level. This further justifies the need of a hybrid strategy. In fact all the graphs show that the best results

TABLE II
NUMBER OF GOALS NEEDED FOR THE SECOND CASE*

	Goal Type	Total Goals	Positive Bids	Negative Bids	New SD RH A	B	C	D
	I – PF	3	2	1	1	1	1	1
1	II – PT	4	4	0	0	2	0	2
	III – SPT	2	2	0	0	0	0	1
	total	**9**			**1**	**3**	**1**	**4**
	I – PT	2	2	0	0	0	0	0
2	II –PF	4	2	2	2	2	1	1
	total	**6**			**2**	**2**	**1**	**1**
	I – PF	3	2	1	1	1	1	1
4	II - PT	2	2	0	0	0	0	0
	total	**5**			**1**	**1**	**1**	**1**

* see the notations from the Table 1

TABLE III
NUMBER OF GOALS NEEDED FOR THE THIRD CASE*

	Goal Type	Total Goals	Positive Bids	Negative Bids	New SD RH A	B	C	D
1	I – PF	3	0	3	1	-	1	-
	total	**3**			**1**	**-**	**1**	**-**
	I – PT	2	0	2	0	-	0	-
2	II – PF	4	0	4	2	-	1	-
	III – SPT	2	0	2	0	-	0	-
	IV - SPF	4	0	4	0	-	1	-
	total	**12(10)**			**2**	**-**	**2**	**-**

* see the notations from the Table 1

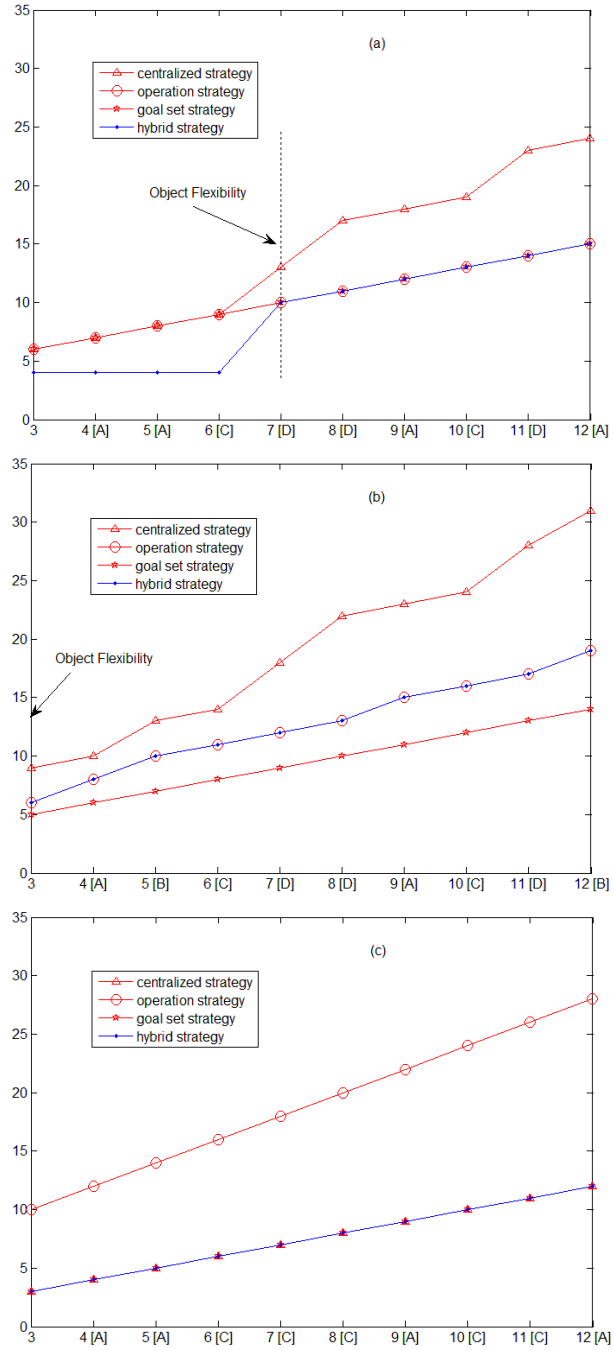


Fig. 5. Goals' number evolution when adding storage devices in the HMS.

are obtained by the hybrid strategy, except for the second case. The use of the goal set strategy creates certain problems (the contractors must possess the capability to manage the goal sets) and thus the hybrid strategy can be preferred as providing a good compromise between complexity and optimality.

V. Conclusions

This contribution stems from a research on HMS development. Using the major points of the usual holonic construction (the main types of holons, the agent based inference mechanism, coordination by means of the CNP) there still remain a lot of open problems regarding the holons' relations and their coordination; thus, a further study is justified. Our analysis on the number of goals handled during the HMS operation is important. Each goal launches a planning process at the contractors' level and thus a goal reduction will clearly increase the HMS efficiency (a faster planning response, avoiding the useless solutions' search). Moreover, if the holon' agent is not kept in a long time goal handling procedure, it will be able to make use of the different types of flexibility. As an example, when the system has no solution due to the lack of parts, the product holon can quickly switch to apply the processing flexibility in order to obtain the unavailable part.

The holonic approach that this contribution is discussing regards a clearer view on the product and resource holons interaction. While the usual holonic approaches give an increased decisional role to the product holons, this paper shows that a greater efficiency is obtained when the decisional process can be switched between the product and resource holons according to the manufacturing context. Though the order holons were not included in this study, it is expected that similar results can be obtained when analyzing the interaction between the order and product holons. As a future work, a link between a specific Petri net model for the HMSs (this was sketched in a first version in [11]) and the manufacturing flexibility is to be studied. This may bring about further evidence that the above proposed decisional scheme is the most appropriate for a holonic architecture.

References

[1] X. Li, C. Zhang, L. Gao, W. Li, and X. Shao, "An agent-based approach for integrated process planning and scheduling," *Expert Systems with Applications*, 37, pp. 1256–1264, 2010.

[2] M. Ghallab, D. Nau, and P. Traverso, *Automated Planning. Theory and Practice*, Morgan Kaufmann, Amsterdam, pp. 85-104, 2004.

[3] Van Brussel, *et. al*., "Reference architecture for holonic manufacturing systems: PROSA," *Computers in Industry*, 37, pp. 255–274, 1998.

[4] D. Panescu, M. Sutu, and C. Pascal, "On the design and implementation of holonic manufacturing systems," *Proc. WRI World Congress on Computer Science and Information Engineering*, 5, pp. 456-461, 2009.

[5] D. M. Dilts, N. P. Boyd, and H. H. Whorms, "The evolution of control architectures for automated manufacturing systems," *Journal of Manufacturing Systems*, 10(1), pp. 79-93, 1991.

[6] M. Huhns and L Stephens, "Multiagent systems and societies of agents," in *Multiagent Systems. A Modern Approach to Distributed Artificial Intelligence*, G. Weiss, Edit., The MIT Press, Cambridge, pp. 96-103, 2001.

[7] D. Panescu, C. Pascal, M. Sutu, and G. Varvara, "Collaborative robotic system obtained by combining planning and holonic architecture," *Proc. AT-EQUAL '09. Advanced Technologies for Enhanced Quality of Life*, pp.138-143, 2009.

[8] D. Panescu, G. Varvara, C. Pascal, and M. Sutu, "On the design and implementation of the resource holons in a PROSA based architecture," *Proceedings of the IEEE 13th International Conference on Intelligent Engineering Systems*, pp. 101-106, 2009.

[9] JACK™ Intelligent Agents, *Agent Manual*, Agent Oriented Software, Carlton South, Victoria, Australia, 2005.

[10] M. Wooldridge, "Intelligent agents", in *Multiagent Systems. A Modern Approach to Distributed Artificial Intelligence*, (Edit. Weiss, G.), The MIT Press, Cambridge, pp. 54-61, 2001.

[11] D. Panescu, and C. Pascal, "Some issues on holonic systems analysis, design and implementation," *Proceedings of the IEEE 19th International Workshop on Robotics in Alpe-Adria-Danube Region*, Obuda University, Budapest, Hungary, 23-27 June, pp. 199-204, 2010.

Algebraic tools for exploring the free-response of discrete-time linear systems: matrix norms versus eigenvalues

Octavian Pastravanu, Mihaela-Hanako Matcovschi*, Alina Doban, and Dorian Florescu

***Abstract*— The eigenvalue location represents a traditional instrument for the analysis of the free response of linear systems. The current paper aims to reveal that, in the discrete-time case, matrix norms also offer powerful tools, supporting investigations complementary to the eigenvalue-based ones. Therefore, we develop a systematic construction that discusses, in parallel, the significance of the mathematical results expressed in terms of eigenvalue location and matrix norms, respectively. Our discussion refers to dynamics described by both single and polytopic state-space representations. A similar study may be proposed for the continuous-time case, focusing on matrix measures instead of matrix norms.**

I. INTRODUCTION

A. Notations

First let us introduce the key notations of our work.

Let $\boldsymbol{v} \in \mathbb{R}^n$ and $\boldsymbol{M} \in \mathbb{R}^{n\times n}$. By $[\![\ ]\!]$ we denote an arbitrary *vector norm*, as well as the induced *matrix norm*, i.e. $[\![\boldsymbol{M}]\!] = \sup_{\boldsymbol{v}\in\mathbb{R}^n, v\neq 0} [\![\boldsymbol{M}\boldsymbol{v}]\!] / [\![\boldsymbol{v}]\!] = \max_{\boldsymbol{v}\in\mathbb{R}^n, [\![\boldsymbol{v}]\!]=1} [\![\boldsymbol{M}\boldsymbol{v}]\!]$. In the particular case of Hölder p-norms, $1 \le p \le \infty$, we use $\|\ \|_p$ for both vector norm and induced matrix norm.

The set $\sigma(\boldsymbol{M}) = \{z \in \mathbb{C} \,|\, \det(z\boldsymbol{I} - \boldsymbol{M}) = 0\}$ denotes the spectrum of $\boldsymbol{M}$ and $\lambda_i(\boldsymbol{M}) \in \sigma(\boldsymbol{M})$, $i = 1,\dots,n$, represent its eigenvalues. The nonnegative value $\rho(\boldsymbol{M}) = \max_{i=1,\dots,n} |\lambda_i(\boldsymbol{M})|$ is the spectral radius of $\boldsymbol{M}$.

If $\boldsymbol{M} \in \mathbb{R}^{n\times n}$ is a symmetrical matrix, $\boldsymbol{M} \prec 0$ ($\boldsymbol{M} \succ 0$) means that matrix $\boldsymbol{M}$ is negative (positive) definite.

If $\boldsymbol{X} \in \mathbb{R}^{n\times m}$, then $|\boldsymbol{X}|$ represents the nonnegative matrix (for $m \ge 2$) or vector (for $m = 1$) defined by taking the absolute values of the entries of $\boldsymbol{X}$.

If $\boldsymbol{X}, \boldsymbol{Y} \in \mathbb{R}^{n\times m}$, then "$\boldsymbol{X} \le \boldsymbol{Y}$", "$\boldsymbol{X} < \boldsymbol{Y}$" mean componentwise inequalities.

B. Preliminary discussion on the role of the algebraic tools

Consider the discrete-time linear system

$$\boldsymbol{x}(t+1) = \boldsymbol{A}\boldsymbol{x}(t),\ \boldsymbol{x}(t_0) = \boldsymbol{x}_0,\ t, t_0 \in \mathbb{Z}_+,\ t \ge t_0, \qquad (1)$$

where $\boldsymbol{x}, \boldsymbol{x}_0 \in \mathbb{R}^n$, $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, and denote by $\boldsymbol{x}(t; t_0, \boldsymbol{x}_0)$ the solution to (1).

The exponential stability represents an important property of the free response of system (1). Although this property is well-known, we briefly recall the ε-δ definition and the necessary and sufficient condition based on the eigenvalue location.

Definition 1 [1].

System (1) is called *exponentially stable* if the equilibrium $\{0\}$ is exponentially stable, i.e

$$\exists\ 0 < r < 1 :\ \forall \varepsilon > 0,\ \forall t_0 \in \mathbb{Z}_+,\ \exists \delta = \delta(\varepsilon) > 0 :$$
$$[\![\boldsymbol{x}_0]\!] \le \delta \Rightarrow [\![\boldsymbol{x}(t; t_0, \boldsymbol{x}_0)]\!] \le \varepsilon\, r^{t-t_0},\ \forall t \ge t_0. \qquad \blacksquare\ (2)$$

Theorem 1 [1]

System (1) is exponentially stable if and only if matrix $\boldsymbol{A}$ is Schur stable, i.e.

$$\rho(\boldsymbol{A}) < 1 \qquad \blacksquare\ (3)$$

Note that the existence of a norm (or norms) such that $[\![\boldsymbol{A}]\!] < 1$ means a sufficient condition for inequality (3), i.e. a sufficient condition for the exponential stability of system (1), since $\rho(\boldsymbol{A}) \le [\![\boldsymbol{A}]\!]$, for any matrix norm $[\![\ ]\!]$.

At this stage of our exposition, the usage of *the matrix norms* appears *conservative* compared to the eigenvalue location. To get a deeper insight into the role of the algebraic tools we discuss the particular case of system (1) corresponding to the following assumption.

Assumption 1.

All the eigenvalues of the matrix $\boldsymbol{A}$ of system (1) are real and there exists a nonsingular matrix $\boldsymbol{V}$ such that $\boldsymbol{V}^{-1}\boldsymbol{A}\boldsymbol{V} = \mathrm{diag}\{\lambda_1(\boldsymbol{A}), \dots, \lambda_n(\boldsymbol{A})\}$. ■

Under this assumption, $\|\boldsymbol{V}^{-1}\boldsymbol{A}\boldsymbol{V}\|_2 = \rho(\boldsymbol{A})$ and we can formulate a necessary and sufficient condition for the exponential stability of (1) based on norms.

Preliminary result 1.

Let Assumption 1 hold true. System (1) is exponentially stable if and only if

$$\|\boldsymbol{V}^{-1}\boldsymbol{A}\boldsymbol{V}\|_2 < 1. \qquad \blacksquare\ (4)$$

Manuscript received September 1, 2010. This work was supported by CNMP, Romania, under Grant 12100/1.10.2008 - SICONA.

The authors are with the Department of Automatic Control and Applied Informatics from the Technical University "Gh. Asachi" of Iasi, Blvd. Mangeron 27, 700050 Iasi, Romania (e-mail: mhanako@ac.tuiasi.ro).

Obviously, inequality (4) expresses an *algebraic result* which is equivalent to condition (3). Nevertheless, a natural question arises: Is this algebraic issue the only sense of (4)? Or, from the point of view of *free-response exploration*, the usage of the matrix norm in inequality (4) corresponds to *a more refined analysis* (not approachable in terms of eigenvalue location ensured by (3))?

The answer should take into consideration the fact that the exponential stability of system (1) is a norm-independent property. In order to investigate the role of the norm $\|\boldsymbol{V}^{-1}\boldsymbol{A}\boldsymbol{V}\|_2$ in relation (4), we apply the state space transformation $\boldsymbol{\xi}=[\xi_1 \dots \xi_n]^T=\boldsymbol{V}^{-1}\boldsymbol{x}$ to system (1) that yields the decoupled system:

$$\xi_i(t+1)=\lambda_i(\boldsymbol{A})\xi_i(t),\ t\ge t_0,\ i=1,\dots,n,$$
$$\boldsymbol{\xi}(t_0)=\boldsymbol{\xi}_0=\boldsymbol{V}^{-1}\boldsymbol{x}_0. \qquad (1')$$

Since $|\xi_i(t;t_0,\boldsymbol{\xi}_0)|=|\lambda_i(\boldsymbol{A})|^{t-t_0}|\xi_i(t_0)|\le\rho(\boldsymbol{A})^{t-t_0}|\xi_i(t_0)|$, system (1') has the property.

$$\forall\varepsilon>0:\ \|\boldsymbol{\xi}_0\|_2\le\varepsilon\Rightarrow$$
$$\Rightarrow\|\boldsymbol{\xi}(t;t_0,\boldsymbol{\xi}_0)\|_2\le\varepsilon\rho(\boldsymbol{A})^{t-t_0},\ t\ge t_0, \qquad (5)$$

which can be rewritten as a property of system (1):

$$\forall\varepsilon>0:\ \|\boldsymbol{V}^{-1}\boldsymbol{x}_0\|_2\le\varepsilon\Rightarrow$$
$$\Rightarrow\|\boldsymbol{V}^{-1}\boldsymbol{x}(t;t_0,\boldsymbol{x}_0)\|_2\le\varepsilon\rho(\boldsymbol{A})^{t-t_0},\ t\ge t_0. \qquad (6)$$

Implication (6) reveals the (*positive*) *invariance* of the set

$$X^{\varepsilon}_{\rho(\boldsymbol{A})}(t;t_0)=\left\{\boldsymbol{x}\in\mathbb{R}^n\,\middle|\,\|\boldsymbol{V}^{-1}\boldsymbol{x}\|_2\le\varepsilon\rho(\boldsymbol{A})^{t-t_0}\right\},$$
$$t\in\mathbb{Z}_+,\ t\ge t_0,\ \varepsilon>0, \qquad (7)$$

with respect to system (1), i.e. the trajectories of system (1) initialized inside $X^{\varepsilon}_{\rho(\boldsymbol{A})}(t_0;t_0)$ remain inside $X^{\varepsilon}_{\rho(\boldsymbol{A})}(t;t_0)$, for any $t\in\mathbb{Z}_+$, $t>t_0$. Moreover, the complete interpretation of the norm-based condition (4) comprises the fact that the invariant set (7) is *exponentially contractive* with the contraction factor $\rho(\boldsymbol{A})<1$.

Preliminary result 2.

Let Assumption 1 hold true. Inequality (4) is a necessary and sufficient condition for the *invariance* of the *exponentially-contractive set* (7) with respect to system (1). ■

From the above construction it is pretty clear that Preliminary result 2 refers to the particular case of the vector norm $[\![\boldsymbol{x}]\!]=\|\boldsymbol{V}^{-1}\boldsymbol{x}\|_2$ and the induced matrix norm $[\![\boldsymbol{A}]\!]=\|\boldsymbol{V}^{-1}\boldsymbol{A}\boldsymbol{V}\|_2$, with $\boldsymbol{V}$ satisfying Assumption 1.

Simple numerical examples show that Preliminary result 2 does not hold true, if, instead of $\boldsymbol{V}$, one uses a nonsingular matrix $\tilde{\boldsymbol{V}}$ satisfying $\rho(\boldsymbol{A})<\|\tilde{\boldsymbol{V}}^{-1}\boldsymbol{A}\tilde{\boldsymbol{V}}\|_2<1$. This shows the role of norm-based conditions in the analysis of the free-response of system (1) requires supplementary scrutiny.

C. Paper objective and organization

The paper reveals the importance of the conditions *expressed in terms of matrix norms* for the characterization of the set invariance properties of discrete-time systems. In parallel we comment on the conditions *expressed in terms of eigenvalue location*.

Section 2 presents a general approach for system (1); the development does not require Assumption 1, and the results are formulated by using arbitrary vector norms (and the induced matrix norms). We show that set invariance implies exponential stability, whereas the converse statement is, in general, not true; from the point of view of the algebraic tools, the norm-based criteria are more restrictive than the (well-known) eigenvalue-based criteria.

Section 3 analyzes the free response of a polytopic system described by (1) with matrix A belonging to a convex hull of matrices, denoted by $\mathcal{A}$. For the set invariance property, we provide a norm-based necessary and sufficient condition that needs testing only the vertex matrices of the convex hull $\mathcal{A}$. From this result, we also derive two corollaries that use a single test matrix, built as a common majorant for all the vertex matrices of $\mathcal{A}$. Instead, the eigenvalues of the vertex matrices cannot be used for the evaluation of the exponential stability of the polytopic system.

II. Free Response Analysis Of System (1) – General Approach

This section develops a general approach to matrix-norm-based conditions in the analysis of set invariance with respect to the free response of system (1). Assumption 1 is no longer valid and the invariant sets are defined by arbitrary vector norms (that replace the invariant sets with particular forms (7) used in the preliminary discussion in Introduction).

Definition 2.

Let $0<r<1$ and $\varepsilon>0$. The exponentially contractive set

$$X^{\varepsilon}_{r}(t;t_0)=\left\{\boldsymbol{x}\in\mathbb{R}^n\,\middle|\,[\![\boldsymbol{x}]\!]\le\varepsilon r^{t-t_0}\right\},\ t_0\in\mathbb{Z}_+,\ t\ge t_0, \qquad (8)$$

is called (*positive*) *invariant* with respect to system (1) if once the initial condition $\boldsymbol{x}(t_0)=\boldsymbol{x}_0$ belongs to $X^{\varepsilon}_{r}(t_0;t_0)$, the corresponding solution $\boldsymbol{x}(t;t_0,\boldsymbol{x}_0)$ is inside $X^{\varepsilon}_{r}(t;t_0)$, for any $t\in\mathbb{Z}_+$, $t>t_0$, i.e.

$$\forall\boldsymbol{x}_0\in\mathbb{R}^n:[\![\boldsymbol{x}_0]\!]\le\varepsilon\Rightarrow[\![\boldsymbol{x}(t;t_0,\boldsymbol{x}_0)]\!]\le\varepsilon r^{t-t_0},\ \forall t\ge t_0. \blacksquare (9)$$

Remark 1.

Note that, due to the linearity of system (1), if the set

$X_r^1(t;t_0)$, defined by (8) for $\varepsilon = 1$, is invariant with respect to system (1), then for every $\varepsilon > 0$ the set $X_r^\varepsilon(t;t_0)$ is also invariant with respect to system (1). ■

The invariance of an exponentially contractive set defined by (8) with respect to system (1) is characterized by the following theorem.

Theorem 2.

Let $0 < r < 1$. The inequality

$$[\![\boldsymbol{A}]\!] \le r, \tag{10}$$

is a necessary and sufficient condition for the invariance of the exponentially-contractive set $X_r^\varepsilon(t;t_0)$ defined by (8) with respect to system (1).

Proof: Sufficiency: We give a proof by contradiction. Since inequality (10) is true, then for all $\boldsymbol{x}_0 \in \mathbb{R}^n$ the corresponding solution to (1) satisfies inequality:

$$[\![\boldsymbol{x}(t+1)]\!] = [\![\boldsymbol{A}\boldsymbol{x}(t)]\!] \le r [\![\boldsymbol{x}(t)]\!], \ \forall t \ge t_0 . \tag{11}$$

Assume that for some $\varepsilon > 0$, the exponentially contractive set $X_r^\varepsilon(t;t_0)$ is not invariant with respect to system (1). Then there exists an initial condition $\tilde{\boldsymbol{x}}_0 \in X_r^\varepsilon(t_0;t_0)$, i.e. $[\![\boldsymbol{x}_0]\!] \le \varepsilon$, so that condition (9) is violated by the corresponding solution to (1), $\tilde{\boldsymbol{x}}(t) = \tilde{\boldsymbol{x}}(t;t_0,\tilde{\boldsymbol{x}}_0)$. This means that there exists $t^* \in \mathbb{Z}_+$, $t^* \ge t_0$, such that $[\![\tilde{\boldsymbol{x}}(t)]\!] \le \varepsilon r^{t-t_0}$ for $t \le t^*$ and $[\![\tilde{\boldsymbol{x}}(t^*+1)]\!] > \varepsilon r^{t^*+1-t_0}$. Thus we get $r[\![\tilde{\boldsymbol{x}}(t^*)]\!] < [\![\tilde{\boldsymbol{x}}(t^*+1)]\!]$, which contradicts (11).

Necessity: Since (9) is true for arbitrary $\varepsilon > 0$, by taking $[\![\boldsymbol{x}_0]\!] = \varepsilon = 1$ and $t = t_0 + 1$ in (9), we obtain $[\![\boldsymbol{A}\boldsymbol{x}_0]\!] = [\![\boldsymbol{A}\boldsymbol{x}(t_0)]\!] = [\![\boldsymbol{x}(t_0+1)]\!] \le r$, for all $\boldsymbol{x}_0$ with $[\![\boldsymbol{x}_0]\!] = 1$. Hence, $[\![\boldsymbol{A}]\!] = \sup_{[\![\boldsymbol{x}_0]\!]=1} [\![\boldsymbol{A}\boldsymbol{x}_0]\!] \le r$. ■

Remark 2.

The invariance of the contractive set $X_r^\varepsilon(t;t_0)$ (8) with respect to system (1) implies the exponential stability of system (1). Indeed, if condition (9) from Definition 2 is satisfied, then condition (2) from Definition 1 is satisfied with $\delta(\varepsilon) = \varepsilon$. Conversely, if for a certain norm $[\![\ ]\!]$, condition (2) is true when $\delta(\varepsilon) > \varepsilon$, but not for $\delta(\varepsilon) = \varepsilon$, then condition (9) is not met. In other words system (1) may be exponentially stable, without having invariant sets defined by the norm $[\![\ ]\!]$.

Regardless of the norm $[\![\ ]\!]$, the contraction factor $0 < r < 1$ of the invariant sets $X_r^\varepsilon(t;t_0)$ (8) cannot be smaller than the spectral radius $\rho(\boldsymbol{A})$. The preliminary discussion in Section 1 corresponds to the limit case $r = \rho(\boldsymbol{A})$, satisfied due to Assumption 1; therefore conditions (3) and (4) express the same numerical inequality, and the two properties (exponential stability and set invariance) look alike. ■

Remark 3.

Let us analyze the shapes of the invariant sets $X_r^\varepsilon(t;t_0)$ (8), when the norm $[\![\ ]\!]$ is defined by weighted Holder norms. If $\boldsymbol{W} \in \mathbb{R}^{n\times n}$ is a nonsingular matrix, these shapes are hyper-ellipsoids for the vector norm $[\![\boldsymbol{x}]\!] = \| \boldsymbol{W}^{-1}\boldsymbol{x} \|_2$ and hyper-parallelepipeds for the vector norms $[\![\boldsymbol{x}]\!] = \| \boldsymbol{W}^{-1}\boldsymbol{x} \|_1$ and $[\![\boldsymbol{x}]\!] = \| \boldsymbol{W}^{-1}\boldsymbol{x} \|_\infty$. In the case when $\boldsymbol{W} = \mathrm{diag}\{w_1,\dots,w_n\}$, $w_i > 0$, $i = 1,\dots,n$, the coordinate axes represent the n symmetry axes of the sets $X_r^\varepsilon(t;t_0)$, and the quantities $w_i \varepsilon r^{t-t_0} > 0$, $i = 1,\dots,n$, define the sizes of the semiaxes at each moment $t \ge t_0$. The case when $[\![\boldsymbol{x}]\!] = \| \boldsymbol{W}^{-1}\boldsymbol{x} \|_\infty$, with $\boldsymbol{W}$ diagonal as above, was separately studied by [2] in the mid 90's, as a property called *componentwise stability* that constrains the trajectories of system (1) by components $-w_i \varepsilon r^{t-t_0} \le x_i(t;t_0,\boldsymbol{x}_0) \le w_i \varepsilon r^{t-t_0}$, $i = 1,\dots,n$, $t \ge t_0$. ■

Remark 4.

Let us investigate the algebraic significance of inequality (10), when the norm $[\![\ ]\!]$ is defined by weighted Holder norms, $[\![\boldsymbol{A}]\!] = \| \boldsymbol{W}^{-1}\boldsymbol{A}\boldsymbol{W} \|_p$, $1 \le p \le \infty$. As discussed in the sequel, inequality (10), regarded as a generic condition for subunitary matrix-norms,

$$[\![\boldsymbol{A}]\!] < 1 \tag{10'}$$

incorporates several concrete conditions well-known from matrix algebra:

a) When $\boldsymbol{W} \in \mathbb{R}^{n\times n}$ is a nonsingular matrix, inequality (10') with $[\![\boldsymbol{A}]\!] = \| \boldsymbol{W}^{-1}\boldsymbol{A}\boldsymbol{W} \|_2$ is equivalent to the Stein inequality (discrete-time Lyapunov inequality) $\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A} - \boldsymbol{Q} \prec 0$, where $\boldsymbol{Q} = (\boldsymbol{W}\,\boldsymbol{W}^T)^{-1}$, which is also equivalent to the Schur stability of matrix $\boldsymbol{A}$.

b) When $\boldsymbol{W} = \mathrm{diag}\{w_1,\dots,w_n\}$, $w_i > 0$, $i = 1,\dots,n$, then condition (10') can be particularized as follows:

b1) For $p = 2$, if $\| \boldsymbol{W}^{-1}\boldsymbol{A}\boldsymbol{W} \|_2 < 1$, then the Stein inequality $\boldsymbol{A}^T\boldsymbol{Q}\boldsymbol{A} - \boldsymbol{Q} \prec 0$ has the diagonal solution $\boldsymbol{Q} = \mathrm{diag}\{1/w_1^2,\dots,1/w_n^2\}$, fact that characterizes the Schur diagonal stability of matrix $\boldsymbol{A}$.

b2) For $p = 1$, if $\| \boldsymbol{W}^{-1}\boldsymbol{A}\boldsymbol{W} \|_1 < 1$, then the generalized Gershgorin's disks of $\boldsymbol{A}$, defined for columns by

$|z-a_{jj}| \le \sum_{i=1, i\neq j}^{n} \frac{w_j}{w_i} |a_{ij}|$, $j=1,\dots,n$, are located strictly inside the complex unit disk $|z| \le 1$.

b3) For $p=\infty$, if $\|\boldsymbol{W}^{-1}\boldsymbol{A}\boldsymbol{W}\|_\infty < 1$, then the generalized Gershgorin's disks of $\boldsymbol{A}$, defined for rows by $|z-a_{ii}| \le \sum_{j=1, j\neq i}^{n} \frac{w_j}{w_i} |a_{ij}|$, $i=1,\dots,n$, are located strictly inside the complex unit disk $|z| \le 1$. ■

Remark 5.

Both exponential stability and set invariance criteria, i.e. inequalities (3) and (10'), respectively, refer to subunitary quantities. Subsequently, the positive values $1-\rho(\boldsymbol{A})$ and $1-[\![\boldsymbol{A}]\!]$ measure the robustness of the two discussed properties. The value $1-\rho(\boldsymbol{A})$ has been intensively used in literature under the nomenclature of *stability margin* for system (1), whereas the role of $1-[\![\boldsymbol{A}]\!]$ remained obscure, despite the similarity of the mathematical expressions. According to Theorem 2, the positive value $1-[\![\boldsymbol{A}]\!]$ quantifies a *contraction margin* for the invariant set $X_r^\varepsilon(t;t_0)$ defined by (8) with the vector norm $[\![\ ]\!]$. For the unperturbed matrix $\boldsymbol{A}$ the contraction factor of $X_r^\varepsilon(t;t_0)$ is $[\![\boldsymbol{A}]\!] < 1$. For any perturbed matrix $\tilde{\boldsymbol{A}}$ the set $X_r^\varepsilon(t;t_0)$ defined by (8) with the same vector norm, is invariant, but no longer contractive once $[\![\tilde{\boldsymbol{A}}]\!] \ge 1$. In other words, for $[\![\tilde{\boldsymbol{A}}]\!] \ge 1$, the discussed sets are still invariant, but they are constant (when $[\![\tilde{\boldsymbol{A}}]\!] = 1$) or exponentially increasing in time (when $[\![\tilde{\boldsymbol{A}}]\!] > 1$), such that their study loses any practical motivation. ■

III. Free Response Analysis of a Polytopic System

Consider the polytopic systems described by

$$\boldsymbol{x}(t+1) = \boldsymbol{A}\boldsymbol{x}(t),\ \boldsymbol{x}(t_0) = \boldsymbol{x}_0,\quad t, t_0 \in \mathbb{Z}_+,\ t \ge t_0,\ \boldsymbol{A} \in \mathcal{A},\ \mathcal{A} \subset \mathbb{R}^{n\times n}, \tag{12}$$

where $\mathcal{A}$ is a convex hull of matrices generated by the set of vertex matrices $\mathcal{V} = \{A_1, A_2, \dots, A_K\} \subset \mathbb{R}^{n\times n}$:

$$\mathcal{A} = \left\{ \boldsymbol{A} \in \mathbb{R}^{n\times n} \,\middle|\, \boldsymbol{A} = \sum_{k=1}^{K} \gamma_k \boldsymbol{A}_k,\ \gamma_k \ge 0,\ \sum_{k=1}^{K} \gamma_k = 1 \right\}. \tag{13}$$

We assume that the parameters of system (12) are not time-varying; the matrix $\boldsymbol{A}$ of system (12) is fixed, but arbitrarily taken from the matrix set $\mathcal{A}$ defined by (13). Consequently, once A is arbitrarily selected from $\mathcal{A}$, the trajectory initialized as $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$, namely $\boldsymbol{x}(t) = \boldsymbol{x}(t;t_0,\boldsymbol{x}_0) = \boldsymbol{A}^{t-t_0}\boldsymbol{x}_0$, is defined for all $t \in \mathbb{Z}_+$, $t \ge t_0$, and has the same generic expression as for system (1).

Therefore, for the polytopic system (12), the concepts of exponential stability and set invariance are introduced on the basis of Definitions 1 and 2, extended for the convex hull $\mathcal{A}$ (13). Thus:

- The polytopic system (12) is exponentially stable if Definition 1 is satisfied $\forall \boldsymbol{A} \in \mathcal{A}$.
- The contractive set $X_r^\varepsilon(t;t_0)$ (8) is invariant with respect to the polytopic system (12) if Definition 2 is satisfied $\forall \boldsymbol{A} \in \mathcal{A}$.

The following result shows that the norm-based conditions are appropriate for exploring the free response of polytopic systems.

Theorem 3.

Let $0 < r < 1$. Consider the polytopic system (12)&(13) with the vertex matrices $\mathcal{V} = \{\boldsymbol{A}_1, \boldsymbol{A}_2, \dots, \boldsymbol{A}_K\} \subset \mathbb{R}^{n\times n}$. The fulfillment of the following inequalities

$$[\![\boldsymbol{A}_k]\!] \le r,\ k = 1,\dots,K, \tag{14}$$

is a necessary and sufficient condition for the invariance of the exponentially-contractive set $X_r^\varepsilon(t;t_0)$ (defined by (8) for any $\varepsilon > 0$) with respect to the polytopic system (12).

Proof: Sufficiency: If condition (14) holds true, then, from the convexity of the matrix norm, we get

$$\forall \boldsymbol{A} \in \mathcal{A},\ \boldsymbol{A} = \sum_{k=1}^{K} \gamma_k \boldsymbol{A}_k:\quad [\![\boldsymbol{A}]\!] \le \sum_{k=1}^{K} \gamma_k [\![\boldsymbol{A}_k]\!] \le \sum_{k=1}^{K} \gamma_k r = \left(\sum_{k=1}^{K} \gamma_k \right) r = r.$$

Necessity: It is obvious, since $\forall \boldsymbol{A} \in \mathcal{A}$, $[\![\boldsymbol{A}]\!] \le r$, and $\boldsymbol{A}_k \in \mathcal{A}$, $k = 1,\dots,K$. ■

Inequalities (14) from Theorem 3 require the test of all vertex matrices $\boldsymbol{A}_1, \boldsymbol{A}_2, \dots, \boldsymbol{A}_K \in \mathbb{R}^{n\times n}$. The following Corollaries 1 and 2 are going to show that, in some particular cases, one can find a *single test matrix* $\boldsymbol{A}^* \in \mathbb{R}^{n\times n}$, such that the satisfaction of inequality:

$$[\![\boldsymbol{A}^*]\!] \le r \tag{15}$$

guarantees the fulfillment of (14).

Corollary 1.

Consider the following hypotheses.

(H1) The vector norm $[\![\ ]\!]$ is a *symmetric gauge function* [3, pp. 438], i.e. it is an absolute vector norm that is a

permutation invariant function of the entries of its argument.

(H2) There exists a matrix $\boldsymbol{A}^* \in \mathbb{R}^{n\times n}$, such that the componentwise inequalities:

$$\boldsymbol{Q}_k \mid \boldsymbol{A}_k \mid \boldsymbol{P}_k \le \boldsymbol{A}^*, k = 1,...,K, \tag{16}$$

hold true for some permutation matrices $\boldsymbol{Q}_k, \boldsymbol{P}_k \in \mathbb{R}^{n\times n}$, $k = 1,...,K$.

(a) If hypotheses (H1), (H2) are satisfied, then inequality (15) is a *sufficient condition* for inequalities (14).

(b) If hypotheses (H1), (H2) are satisfied and there exists a vertex matrix $\boldsymbol{A}^{**} \in \mathcal{V}$ such that $\|\boldsymbol{A}^{**}\| = \|\boldsymbol{A}^*\|$, then inequality (15) is a a *necessary and sufficient condition* for inequalities (14).

Proof. (a) We organize the proof in two parts. Part I proves the following results:

(R1) If $\boldsymbol{P}, \boldsymbol{Q} \in \mathbb{R}^{n\times n}$ are permutation matrices, then

$$\forall \boldsymbol{\Phi} \in \mathbb{R}^{n\times n}: \|\boldsymbol{Q}\boldsymbol{\Phi}\boldsymbol{P}\| = \|\boldsymbol{\Phi}\|. \tag{17}$$

(R2) $$\forall \boldsymbol{\Phi} \in \mathbb{R}^{n\times n}: \|\boldsymbol{\Phi}\| \le \|\,|\boldsymbol{\Phi}|\,\|. \tag{18}$$

(R3) $$\forall \boldsymbol{\Phi}, \boldsymbol{\Theta} \in \mathbb{R}^{n\times n}, 0 \le \boldsymbol{\Phi} \le \boldsymbol{\Theta}: \|\boldsymbol{\Phi}\| \le \|\boldsymbol{\Theta}\|. \tag{19}$$

Part II uses (R1) - (R3) to show that (15) implies (14).

Proof of Part I:

(R1) Let $\boldsymbol{P}, \boldsymbol{Q} \in \mathbb{R}^{n\times n}$ be permutation matrices. From the definition of the matrix norm, there exists $\boldsymbol{y}^* \in \mathbb{R}^n$, $\|\boldsymbol{y}^*\| = 1$, such that $\|\boldsymbol{P}\| = \|\boldsymbol{P}\boldsymbol{y}^*\|$. Since the considered vector norm $\|\ \|$ is permutation invariant, we have $\|\boldsymbol{y}^*\| = \|\boldsymbol{P}\boldsymbol{y}^*\| = 1$. Hence, $\|\boldsymbol{P}\| = 1$, and, similarly, $\|\boldsymbol{Q}\| = 1$. Now, for an arbitrary matrix $\boldsymbol{\Phi} \in \mathbb{R}^{n\times n}$, we can write $\|\boldsymbol{Q}\boldsymbol{\Phi}\boldsymbol{P}\| \le \|\boldsymbol{Q}\|\|\boldsymbol{\Phi}\|\|\boldsymbol{P}\| = \|\boldsymbol{\Phi}\|$. Since $\boldsymbol{P}^T, \boldsymbol{Q}^T \in \mathbb{R}^{n\times n}$ are also permutation matrices and $\boldsymbol{Q}^T\boldsymbol{Q} = \boldsymbol{I}$, $\boldsymbol{P}\boldsymbol{P}^T = \boldsymbol{I}$, we also have the inequality

$$\|\boldsymbol{\Phi}\| = \left\|(\boldsymbol{Q}^T\boldsymbol{Q})\boldsymbol{\Phi}(\boldsymbol{P}\boldsymbol{P}^T)\right\| = \left\|\boldsymbol{Q}^T(\boldsymbol{Q}\boldsymbol{\Phi}\boldsymbol{P})\boldsymbol{P}^T\right\| \le$$
$$\le \left\|\boldsymbol{Q}^T\right\|\|\boldsymbol{Q}\boldsymbol{\Phi}\boldsymbol{P}\|\left\|\boldsymbol{P}^T\right\| = \|\boldsymbol{Q}\boldsymbol{\Phi}\boldsymbol{P}\|.$$

As a consequence, we get (17).

(R2) From the definition of the matrix norm, there exists $\boldsymbol{y}^* \in \mathbb{R}^{n\times n}$, $\left\|\boldsymbol{y}^*\right\| = 1$, such that $\|\boldsymbol{\Phi}\| = \left\|\boldsymbol{\Phi}\boldsymbol{y}^*\right\|$. Since $\|\ \|$ is a symmetric gauge function, it is also an absolute vector norm, and, equivalently, a monotonic vector norm [3, Theorem 5.5.10]. From the componentwise vector inequality $|\boldsymbol{\Phi}\boldsymbol{y}^*| \le \left|\,|\boldsymbol{\Phi}|\,|\boldsymbol{y}^*|\,\right|$, we have $\left\|\boldsymbol{\Phi}\boldsymbol{y}^*\right\| \le \left\|\,|\boldsymbol{\Phi}|\,|\boldsymbol{y}^*|\,\right\| \le \|\,|\boldsymbol{\Phi}|\,\|\left\|\,|\boldsymbol{y}^*|\,\right\|$, where $\left\|\,|\boldsymbol{y}^*|\,\right\| = \left\|\boldsymbol{y}^*\right\| = 1$. It results (18).

(R3) From the definition of the matrix norm, there exists $\boldsymbol{y}^* \in \mathbb{R}^n$, $\|\boldsymbol{y}^*\| = 1$, such that $\|\boldsymbol{\Phi}\| = \|\boldsymbol{\Phi}\boldsymbol{y}^*\|$. For $0 \le \boldsymbol{\Phi} \le \boldsymbol{\Theta}$, we have $|\boldsymbol{\Phi}\boldsymbol{y}^*| \le \left|\,|\boldsymbol{\Phi}|\,|\boldsymbol{y}^*|\,\right| \le \left|\,|\boldsymbol{\Theta}|\,|\boldsymbol{y}^*|\,\right|$, and, from the monotonicity of the vector norm $\|\ \|$ we can write $\|\boldsymbol{\Phi}\boldsymbol{y}^*\| \le \|\,|\boldsymbol{\Theta}|\,|\boldsymbol{y}^*|\,\| \le \|\,|\boldsymbol{\Theta}|\,\|\|\,|\boldsymbol{y}^*|\,\|$, where $\|\,|\boldsymbol{y}^*|\,\| = \|\boldsymbol{y}^*\| = 1$ because $\|\ \|$ is an absolute norm. Hence, it results (19).

Proof of Part II: For all $k = 1,...,K$, from (17), according to (R3) we get $\|\boldsymbol{Q}_k \mid \boldsymbol{A}_k \mid \boldsymbol{P}_k\| \le \left\|\boldsymbol{A}^*\right\|$; we also have $\|\boldsymbol{Q}_k \mid \boldsymbol{A}_k \mid \boldsymbol{P}_k\| = \|\,|\boldsymbol{A}_k|\,\|$ by (R1), and $\|\boldsymbol{A}_k\| \le \|\,|\boldsymbol{A}_k|\,\|$ by (R2). Finally, it results $\|\boldsymbol{A}_k\| \le \left\|\boldsymbol{A}^*\right\|$ which together with (15) lead to (14), and one can apply Theorem 3.

(b) The sufficiency is proved by (a). The necessity is ensured by the equality $\left\|\boldsymbol{A}^*\right\| = \left\|\boldsymbol{A}^{**}\right\|$ and the inequality $\left\|\boldsymbol{A}^{**}\right\| \le r$ (resulting from $\boldsymbol{A}^{**} \in \mathcal{V}$). ■

Corollary 2.

Consider the following hypotheses.

(H1) The vector norm $\|\ \|$ is an absolute vector

(H2) There exists a matrix $\boldsymbol{A}^* \in \mathbb{R}^{n\times n}$, such that the componentwise inequalities:

$$|\boldsymbol{A}_k| \le \boldsymbol{A}^*, \ k = 1,...,K, \tag{20}$$

hold true.

(a) If hypotheses (H1), (H2) are satisfied, then inequality (15) is a *sufficient condition* for inequalities (14).

(b) If hypotheses (H1), (H2) are satisfied and there exists a vertex matrix $\boldsymbol{A}^{**} \in \mathcal{V}$ such that $\left\|\boldsymbol{A}^{**}\right\| = \left\|\boldsymbol{A}^*\right\|$, then inequality (15) is a a *necessary and sufficient condition* for inequalities (14).

Proof: We use the same technique as in the proof of Corollary 1. ■

Remark 6.

For the sufficiency part of Corollaries 1 and 2, the hypothesis (16) and, respectively, (20) can be relaxed, if, instead of a single test matrix $\boldsymbol{A}^*$, one uses several test matrices $\boldsymbol{A}_1^*, \boldsymbol{A}_2^*, \ldots, \boldsymbol{A}_L^*$, with L significantly smaller than K. Each matrix $\boldsymbol{A}_\ell^*$, $\ell = 1,\ldots,L$, will have to satisfy (16) or (20) for a group of vertex matrices $\mathcal{V}_\ell \subseteq \mathcal{V}$, such that $\bigcup_{\ell=1}^{L} \mathcal{V}_\ell = \mathcal{V}$, $\mathcal{V}_{\ell_1} \bigcap \mathcal{V}_{\ell_2} = \varnothing$, $\ell_1 \ne \ell_2$, $\ell_1, \ell_2 = 1,\ldots,L$. ■

With reference to the eigenvalue location, there exists no connection between the vertex matrices $\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_K \in \mathbb{R}^{n\times n}$ and an arbitrary matrix $\boldsymbol{A}$ belonging to the convex hull $\mathcal{A}$ defined by (13). Simple numerical examples show that the Schur stability of the vertex matrices does not imply the Schur stability of any $\boldsymbol{A} \in \mathcal{A}$. However

part a) of Corollary 2 can be transformed so as to provide a sufficient condition for the spectral radii of all the matrices $\boldsymbol{A} \in \mathcal{A}$.

Corollary 3.

If there exists a matrix $\boldsymbol{A}^* \in \mathbb{R}^{n\times n}$, such that the componentwise inequalities (20) hold true, then

$$\forall \boldsymbol{A} \in \mathcal{A}: \ \rho(\boldsymbol{A}) \le \rho(\boldsymbol{A}^*) . \tag{21}$$

Proof: For any $\boldsymbol{A} \in \mathcal{A}$, we have

$$|\boldsymbol{A}| \le \sum_{k=1}^{K} \gamma_k |\boldsymbol{A}_k| \le \sum_{k=1}^{K} \gamma_k \boldsymbol{A}^* = \left(\sum_{k=1}^{K} \gamma_k \right) \boldsymbol{A}^* = \boldsymbol{A}^* .$$

Thus, in accordance with Theorem 8.1.18 in [3], we can write $\rho(\boldsymbol{A}) \le \rho(|\boldsymbol{A}|) \le \rho(\boldsymbol{A}^*)$. ■

Remark 7.

The lack of a characterization in terms of eigenvalues for the exponential stability of the polytopic system (12) yielded the concept of "*quadratic stability*" for polytopic systems, e.g. [4, pp. 213]. This concept is stronger than the exponential stability of the polytopic system (12), since it requires the existence of a unique $\boldsymbol{Q} \succ 0$ satisfying the matrix Stein inequality $\boldsymbol{A}^T \boldsymbol{Q} \boldsymbol{A} - \boldsymbol{Q} \prec 0$ for all $\boldsymbol{A} \in \mathcal{A}$. As a matter of fact, the quadratic stability is equivalent to the invariance of the set $X_r^{\varepsilon}(t;t_0)$ (8) with respect to the polytopic system (12), when the norm used in (8) is $[\![\boldsymbol{x}]\!] = \| \boldsymbol{W}^{-1}\boldsymbol{x} \|_2$. (We consider $\boldsymbol{Q} = (\boldsymbol{W}\ \boldsymbol{W}^T)^{-1}$ in the Stein inequality - similar to Remark 4 (a) referring to system (1).)■

For a rather restrictive class of polytopic systems, from Corollary 3 we can derive a necessary and sufficient condition for the exponential stability. For such polytopic systems, the quadratic stability and the exponential stability coincide.

Corollary 4.

Let $\boldsymbol{A}^* \in \mathcal{V}$ be a vertex matrix of the convex hull $\mathcal{A}$ (13), which satisfies the componentwise inequalities (20). The polytopic system (12) is exponentially stable, if and only if

$$\rho(\boldsymbol{A}^*) < 1 . \quad \blacksquare \tag{22}$$

Remark 8.

The literature on dynamic systems includes results representing particular cases of Corollaries 3 and 4, which refer to interval systems, where the convex hull of matrices has the particular form $\mathcal{A}_I = \left\{ \boldsymbol{A} \in \mathbb{R}^{n\times n} \,\middle|\, \boldsymbol{A}^- \le \boldsymbol{A} \le \boldsymbol{A}^+ \right\}$, with $\boldsymbol{A}^+ = (a_{ij}^+)$, $\boldsymbol{A}^- = (a_{ij}^-)$. For an interval matrix $\mathcal{A}_I$, the matrix $\boldsymbol{A}^* = (a_{ij}^*)$ defined by

$$a_{ij}^* = \max\{|a_{ij}^-|, |a_{ij}^+|\}, \ i,j = 1,\ldots,n , \tag{23}$$

satisfies condition (20), such that our Corollary 4 generalizes Corollary 1 in [5], Corollary 2.3 in [6] and Theorem 5.2 in [7]. ■

Remark 9.

The approach to interval systems based on the majorant matrix $\boldsymbol{A}^*$ (23) was used in [8] to formulate a necessary and sufficient condition for the property called componentwise stability. (See Remark 3 for the presentation of this property in the case of system (1), which is naturally extended to interval systems by [8]). Corollary 2 (ii) in [8] is incorporated by our Corollary 2 (b) as a particular case, where the invariant set $X_r^{\varepsilon}(t;t_0)$ is defined by the vector norm $[\![\boldsymbol{x}]\!] = \| \boldsymbol{W}^{-1}\boldsymbol{x} \|_{\infty}$, with $\boldsymbol{W} = \mathrm{diag}\{w_1,\ldots,w_n\}$, $w_i > 0$, $i = 1,\ldots,n$. ■

IV. Conclusions

In exploring the free response of discrete-time linear systems the mathematical results formulated in terms of eigenvalue location or matrix norms are equally helpful. Each of the two types of instruments provides necessary and sufficient conditions for the characterization of two different properties; the former supports the stability analysis, whereas the latter is used for the study of set invariance. Our work considers the fundamental case of a single state-space representation, as well as the multi-model case described by a polytopic system.

References

[1] A. Michel, K. Wang, and B. Hu, *Qualitative Theory of Dynamical Systems. The Role of Stability Preserving Mappings*, 2nd Edition, Marcel Dekker, Inc.: New York, 2001

[2] A. Hmamed, Componentwise stability of 1-D and 2-D linear discrete-time systems, *Automatica*, vol. **33**, no. 9, pp. 1759-1762, 1997.

[3] R.A. Horn, and C.R. Johnson. *Matrix Analysis*, Cambridge University Press: Cambridge, 1990

[4] F. Blanchini, and S. Miani. *Set-Theoretic Methods in Control,* Birkhäuser: Boston, 2008.

[5] P.H. Bauer, and K. Premaratne. Time-invariant versus time-variant stability of interval matrix systems. In *Fundamentals of Discrete-Time Systems* (Jamshidi M., Mansour M., Anderson B.D.O., Eds.), TSI Press: Albuquerque, pp. 181-188, 1993.

[6] M.E. Sezer, and D.D. Šiljak, On stability of interval matrices, *IEEE Trans. on Automatic Control*, vol. **39**, no. 2, pp. 368-371, 1994.

[7] A. Molchanov and D. Liu, Robust absolute stability of nonlinear time-varying discrete-time systems, *IEEE Trans. on Circuits and Systems-I: Fund. Theory and App.,* vol. **49**, no.8, pp.1129-1137, 2002.

[8] O. Pastravanu, and M. Voicu, Necessary and sufficient conditions for componentwise stability of interval matrix systems, *IEEE Trans. on Automatic Control*, vol. **49**, pp. 1016-1021, 2004.

Dynamic Fuzzy Controlled Regressor Encapsulation in Evolving Nonlinear Models

A. Patelli and L. Ferariu

***Abstract* — An enhanced crossover operator, based on fuzzy controlled regressor encapsulation, is used as part of a customized genetic programming tool for nonlinear systems identification. An evolutionary approach was chosen for solving the aforementioned engineering problem due to its robustness, its scarce aprioric requirements and its natural capacity of successfully addressing multiobjective optimization issues. To help diminish computational resource consumption, by encouraging the production of fitter offspring, the paper suggests the use of a fuzzy controller in deciding which model terms would bring the most relevant gain in fitness, if swapped via crossover. The thresholds involved in membership functions definitions are dynamically computed to better reflect regressor relevance. In addition, the basic evolutionary loop is upgraded with original similarity analysis and genetic material refreshment mechanisms aimed at preserving population diversity. The practical usefulness of the proposed tool is demonstrated within an experimental trial involving the identification of a complex nonlinear industrial system within the Sugar Factory of Lublin, Poland.**

I. Introduction

Obtaining accurate and parsimonious mathematical models for nonlinear systems is a key step to take in all automatic control applications, as an appropriate closed loop control law is difficult to determine otherwise. The most straightforward form a nonlinear model may be generated in is a polynomial equation, as it is fairly simple and easily exploited by numerical algorithms. In addition, it has been rigorously demonstrated that any continuous bounded function can be approximated by a polynomial model, to any desired degree of accuracy [1], therefore extensive research has been conducted concerning **N**onlinear **L**inear in **P**arameter (NLP) models.

In short, NLP representations are linear combinations of nonlinear atoms called regressors. When given access to a representative set of experimental data, it is easy to include all possible combinations of regressors in a maximal model. The redundant terms will afterwards be eliminated via pruning techniques which are usually time consuming and may take up significant computational memory resources [2]. Another possible approach is to incrementally build an NLP model by gradually adding regressors as result of blind search or other deterministic methods that will most likely provide a solution only after prolonged runtime [3].

A feasible alternative to this line of research is represented by evolutionary techniques, namely **G**enetic **P**rogramming (GP). In the context of this approach, several potential models are generated across the search space in the form of tree encrypted individuals [4] which are subsequently evolved according to the Darwinian principle of the survival of the fittest. In consequence, GP based identification tools are robust, as they feature no derivatives and work on populations of candidate solutions, instead of singular individuals [4]. Another useful feature is their ability to cope with scarce *a priori* information, as they can work with any type of search space landscape [5]. As far as initial algorithm configuring is concerned, most internal working parameters only require "trial and error" tuning at no significant supplementary computational costs [5]. All these facts make the GP approach suitable for cases where the shape and dimension of the final solution are not known beforehand, where there are unexpected relations between variables, otherwise difficult to capture, or where other analytical methods impose unrealistic working hypothesis [6].

Within the evolutionary framework described above, the authors suggest a novel identification tool enhanced to address the specific requirements of Multi Objective Optimization (MOO). Therefore, diversity preservation is encouraged by means of similarity analysis and fresh genetic material injection [7], whereas the main contribution of the paper resides in the use of adaptive, fuzzy controlled encapsulation to guide the crossover operator in effectively selecting the regressors to be swapped. After identifying similar terms featured by several trees in the current population, the proposed procedure assigns an encapsulation probability to each of them according to a set of fuzzy rules, which employ dynamically configured membership functions. This is meant to encourage the production of fitter offspring in relation to their parents, thus speeding up the search process and reducing the overall runtime. Additionally, the authors propose a memetic approach, residing in the symbiosis between structure selection via genetic operators and parameter computation by means of a deterministic local optimization plug-in based on QR decomposition. The reason behind this is that parameter

Manuscript received April 30, 2010. This work was supported by The National Centre for Programs Management from Romania under the research grant SICONA – 12100/2008.

A. Patelli is with the Department of Automatic Control and Industrial Informatics, "Gheorghe Asachi" Technical University of Iasi, Romania, Bd. D. Mangeron 27, IS 700050 RO (corresponding author, phone: 0728214932 e-mail: apatelli@ ac.tuiasi.ro).

L. Ferariu is with the Department of Automatic Control and Industrial Informatics, "Gheorghe Asachi" Technical University of Iasi, Romania, Bd. D. Mangeron 27, IS 700050 RO (e-mail: lferaru@ac.tuiasi.ro).

wise linearity invites the use of a deterministic procedure for fast and accurate coefficient determination. Another advantage of this hybridization is the fact that structure selection and parameter computation are carried out in an interdependent manner, whilst other non-evolutionary identification procedures perform the two tasks independently.

The paper initially browses through the most significant related work in the field. Section III describes the tree generation and evaluation mechanisms, whilst the following section details the enhanced genetic operators involved in offspring production. Section V summarizes the elite specific enhancements implemented by the authors. A representative experimental trial and a set of conclusions are included in the final two sections, respectively.

II. Related Work

Early attempts of generating nonlinear models by evolving NLP compliant tree individuals employed a single optimization objective, accuracy, thus output only one system model [8]. Following in that trend, a transformation mechanism designed to assure parameter wise linearity for all involved trees was introduced [9]. According to it, the ill positioned operator nodes featured by the randomly generated individuals would be identified and replaced by appropriate alleles. Although rapid and easy to implement, the procedure increases the risk of regressor bloat with a negative influence on model parsimony. In exchange, the authors suggest a different idea that promotes an even distribution of the trees in the initial population over the search space, and afterwards employs a transformation routine that guarantees mathematical equivalence with two immediate advantages: the preservation of the initial problem domain coverage and controlling regressor size.

As the area of automatic control imposes specific requirements in model quality assessment, an appropriate individuals' evaluation should be subject to multiple optimization criteria. Selecting the number and nature of the objectives to use is a delicate decision. Some researches employ an increased number of evaluation functions in the quest for a highly accurate tree assessment [10]. The main downside is that excessively harsh evaluation may prematurely exclude trees from the population, draining the pool of genetic material subject to the action of genetic operators. Balancing the importance of each objective is also difficult. Therefore the approach presented in this paper resorts to two assessment criteria: accuracy and parsimony.

After settling for the appropriate optimization objectives to use, the next step is combining all the evaluation related information into a single value, called fitness, later on used to select trees for the reproduction pool. To manage that, especially when dealing with conflicting objectives, Deb suggested assigning fitness via non-dominance analysis [11]. The authors have proposed population clustering and adaptive migration as enhancements to Deb's approach to better suit the engineering related requirements of the identification problem [12]. Nondominated sorting, in Deb's view, is carried out in the objectives space which is easier to exploit than the decision space composed of individuals of various sizes (each individual features a different number of parameters). However, the literature contains references that employ neural networks to map the objectives space onto the search one [13]. Furthermore, techniques have been documented that direct the search to specific areas of the objectives space, called Pareto knees, by dynamically aggregating the initial objective functions into new ones, adapted to the interest regions [14].

To speed up the search process, increasing selection pressure has been suggested, by storing external archives of elite individuals copies, used for fitness computation purposes only [15]. Other efforts aimed at limiting algorithm runtime involve hybridizing the evolutionary loop with deterministic procedures, like the **O**rthogonally **L**east **S**quare (OLS) tool that computes error ratios for each regressor to determine the least significant ones and eliminate them [9]. To avoid the risk of premature model term elimination, brought on by the above procedure, the authors evaluate regressor relevance by means of a stochastic encapsulation procedure overlooked by an adaptive fuzzy controller, as detailed in paragraph IV.

III. Generation, Transformation and Evaluation of Trees

A healthy evolution depends on the distribution of the initial genetic material over the search space. In response to that requirement, the authors suggest an upgrade of the classic tree generation routine based on random recursive insertion of nodes. The enhancement refers to a set of rules that mainly consist in including each available terminal (1) only once in each chromosome, in assuming a slightly higher probability for terminals insertion than the one of operator nodes, and in filling empty leaf slots with constants [12]. The result is the generation of individuals rich in non-redundant genetic material, evenly distributed across the problem domain, and encrypted by well balanced trees.

Due to the organization of the terminal set **x**, which contains lagged input u_i and output y_i values, n_u and n_y being the maximum allowed lags, the dynamic nature of the working models is implicitly guaranteed.

$$\mathbf{x}(k) = [u_i(k),..,u_i(k-n_u), y_j(k-1),..,y_j(k-n_y)],$$
$$i = \overline{1,m},\ j = \overline{1,n} \tag{1}$$

That reason, combined with the parameter wise linearity of NLP models leads to a minimally sufficient operator set O = {+, *}. The mandatory closure and sufficiency properties [5] of sets **x** and O are further exploited by the three rule tree building algorithm for providing the search process with a good start.

Given the layout of the NLP formalism, compliant models can be described by the following matrix equation:

$$\begin{bmatrix} F_{i1}(\mathbf{x}(1)) & .. & \cdots & F_{ir}(\mathbf{x}(1)) \\ F_{i1}(\mathbf{x}(2)) & .. & \cdots & F_{ir}(\mathbf{x}(2)) \\ \vdots & \vdots & & \vdots \\ F_{i1}(\mathbf{x}(p)) & .. & \cdots & F_{ir}(\mathbf{x}(p)) \end{bmatrix} \begin{bmatrix} c_{i1} \\ c_{i2} \\ \vdots \\ c_{ir} \end{bmatrix} = \begin{bmatrix} \hat{y}_i(1) \\ \hat{y}_i(2) \\ \vdots \\ \hat{y}_i(p) \end{bmatrix},$$

$$\mathbf{F}_i \cdot \mathbf{c}_i = \hat{\mathbf{y}}_i, \quad \mathbf{F}_i \in \Re^{p \times r}; \mathbf{c}_i \in \Re^r; \hat{\mathbf{y}}_i \in \Re^p, i = \overline{1,n}, \qquad (2)$$

where **F** is the regression matrix, **c** is the parameter vector, y_i denotes the i^{th} plant output and p stands for the training data set length. The suggested tree generation algorithm does not guarantee NLP compliance, so, in order to be able to attach an equation like (2) to each chromosome, a transformation procedure is necessary. To reposition the ill placed "+" operators inside the trees, while preserving mathematical equivalence with the original individuals, the nodes are reconfigured according to an elementary arithmetic identity, $a \cdot (b + c) = a \cdot b + a \cdot c$. This additional tree processing effort is directed towards facilitating the symbiosis with the deterministic QR local optimization procedure that can easily solve (2) and determine the optimal set of parameters for each tree encrypted structure.

Once their structure and parameters have been established, the trees in the initial population are ready for evaluation, via the two following objective functions:

$$SEF(M) = \frac{1}{2} \sum_{k=1}^{p} (y_i(k) - \hat{y}_i(k))^2, \quad k = \overline{1,p} \quad \text{and}$$

$$CF\ (M\ ) = r + \frac{t}{n_u + n_y + 1} - \sum_{q=1}^{r} \lg c_{iq}\ . \qquad (3)$$

The **S**quared **E**rror **F**unction (*SEF*) measures tree accuracy, whilst the **C**omplexity **F**unction (*CF*) is in charge of evaluating chromosome parsimony. Information about the total number of regressors, *r*, the terminals included in those regressors, relative to the number of available ones, *t*, as well as parameter relevance is reflected, in that order, by the three terms of the complexity criterion that the present approach employs. The last two components of *CF* are meant to penalize candidate models that contain a low number of large regressors as well as the ones featuring insignificant parameters, as neither have any practical significance.

Even though model parsimony is encouraged by other inherent components of the evolutionary algorithm (tree generations, enhanced genetic operators), considering a separate complexity objective is preferred, as the algorithm can output a whole set of nondominated solutions from which the designer can afterwards choose one or several tradeoff models, depending on the identification application's specific. However, not all solutions from the Pareto front are of practical use. The models situated on the extremes are either inaccurate or overfitted, with unsatisfactory generalization capacities. To avoid those areas and focus tree generation on the central Pareto front area, the authors suggest clustering procedures that target both regular trees [12] and elites (section V).

IV. Adaptive Fuzzy Encapsulation and Enhanced Genetic Operators

Towards the end of the evolutionary loop, the trees are expected to get closer to the optimal Pareto front. In consequence, it is highly probable for most of them to feature similar regressors that have survived over generations as well adapted model terms, which significantly contribute to increasing overall tree performances. Therefore they should be protected from division by crossover, so that offspring individuals should also benefit from them. Allowing well adapted model terms to pass on to future generations is called exploitation. On the other hand, population diversity must also be encouraged, as the search process should explore new areas of the problem domain by generating children trees which are better than their parents and significantly different at the same time. To do so, the authors suggest an original crossover operator designed to identify and avoid similar regressors, while selecting the remaining, non-resembling model terms instead. They are the ones to be swapped in an effort to generate better, yet diverse offspring, thus balancing between exploitation and exploration.

In practice, the first step to take is determining which similar regressors are truly well adapted model terms. The proposal is to employ a set of fuzzy rules. The trees in the current population are processed and all regressor featured by at least two individuals will be encapsulated with a probability that reflects how well adapted they truly are. Five fuzzy sets are used to that end. The first three are related to the fuzzy variable *var*(*SEF*) which stands for the variance of the *SEF* values (3) associated to all trees that feature the current regressor being encapsulated. They are labeled *low*, *medium* and *high* (Fig. 1). The remaining fuzzy sets are defined in relation to the size of the targeted regressor, *size*(*REG*), which can either be acceptable (fuzzy set is labeled *ok*) or not (fuzzy set is labeled *large*). The **M**embership **F**unctions (MF) associated to the two fuzzy variables and the five fuzzy sets are shown in Fig. 1.

For a complete definition of the membership functions, the values for the trapezes' vertices, v_i, i = 1, .., 6, must be accurately selected. On the one hand, the aforementioned parameters need to be flexible enough to allow all trees in the current population to be categorized into the appropriate fuzzy set. That goal may be achieved by dynamically computing a set of values for v_i, i = 1, .., 6 at each generation, based on an analysis of the individuals

performances. However, that is not a valid approach, as the six parameters have to be constant throughout the evolutionary process for a relevant assessment of all potential models, regardless of the generation they belong to. To settle the issue, the authors suggest selecting accurate v_i, i = 1, .., 6 values as result of a separate learning process detailed later on in this section.

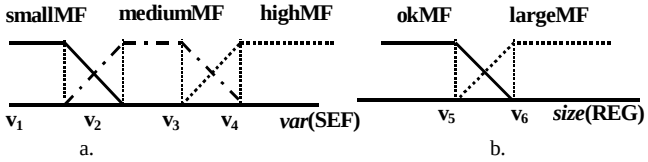


Fig. 1. Fuzzy sets and membership functions

IF *var*(*SEF*) **IS** *small* **AND** *size*(*REG*) **IS** *ok* **THEN** *P*[*enc*] = 1
IF *var*(*SEF*) **IS** *medium* **AND** *size*(*REG*) **IS** *ok* **THEN** *P*[*enc*] = 0.8
IF *var*(*SEF*) **IS** *high* **AND** *size*(*REG*) **IS** *ok* **THEN** *P*[*enc*] = 0
IF *var*(*SEF*) **IS** *small* **AND** *size*(*REG*) **IS** *large* **THEN** *P*[*enc*] = 0.9
IF *var*(*SEF*) **IS** *medium* **AND** *size*(*REG*) **IS** *large* **THEN** *P*[*enc*] = 0.5
IF *var*(*SEF*) **IS** *high* **AND** *size*(*REG*) **IS** *large* **THEN** *P*[*enc*] = 0

Fig. 2. Fuzzy rules

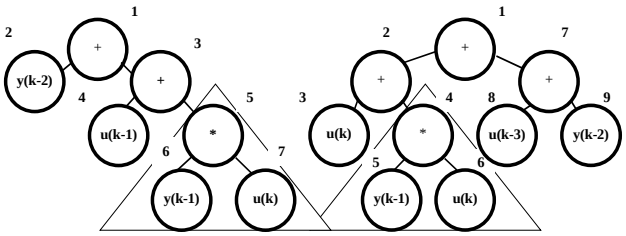


Fig. 3. Parents featuring one identical regressor

The encapsulation probability to be attached to the current similar regressors is computed by means of the fuzzy rules indicated in Fig. 2. To better understand the encapsulation process and the way it guides the crossover operator in selecting swap regressors, let us consider the example presented in Fig. 3. The two parents selected by crossover feature one identical regressor marked by a triangle (Fig. 3). Let us assume that, for all trees in the current population that feature the exact same regressor as the one inside the triangles, *var*(*SEF*) results *small* and that *size*(*REG*) is rather *large*. In that case, the fourth fuzzy rule (Fig. 2) is activated and the regressor is encapsulated with an encapsulation probability of 0.9 in all the trees that contain it, including the two parents presented in Fig. 3. The probability of selecting a cut point node within the identical regressor is 1 minus the encapsulation probability, which, for this example, is evaluated to 0.1. In other words, it is highly unlikely for crossover to divide this particular regressor, as the fuzzy control algorithm has found it useful. This example is meant to illustrate the opposing nature of encapsulation on the one hand and crossover on the other. The former is protective, meant to encourage exploitation, whilst the latter is dividing, aimed at favoring exploration.

As mentioned before, the encapsulation procedure is useful only in the final stages of the evolution process. In early generations, similar regressors are mostly coincidental and cannot be interpreted as a sign of tree adaptation. Note that, in the initial population, each tree includes all available terminals, however, the resulting similar regressors offer no indication as to adaptation. That is why encapsulation is applied only in the final *max_gen_no*/10 generations (*max_gen_no* stand for the maximum number of generations that the algorithm is allowed to run for). The initial generations are used as a "training period" to configure the parameters of the MF, v_i, i = 1, .., 6 (Fig. 1), by storing the variance of *SEF* values and the one of *CF* values for each population, in the vectors *var*(*SEF*) and *var*(*CF*). When the *max_gen_no*/10 threshold is reached, the mean value for each of the two vectors is computed and the parameters v_i, i = 1, .., 6 are evenly distributed as follows:

$$\begin{aligned} v_i &= [mean\ (var(\ SEF\ ))/4]*i, \quad i = 1,2,3,4, \\ v_i &= [mean\ (var(\ CF\ ))/2]*i, \quad i = 5,6 \end{aligned} \tag{4}$$

The other genetic operator, mutation is enhanced to avoid compensation [12]. Thus, both crossover and mutation, customized by the authors in the manner described above, have a double role: controlling tree complexity, as well as encouraging the generation of fitter offspring. The additional tree processing performed by the upgraded genetic operators is in fact computationally cheaper than generating poorly adapted children, a genuine risk when working with raw versions of the evolutionary tools.

V. Elite Related Upgrades

To increase algorithm search speed, the authors suggest an elitist approach upgraded by several original enhancements meant to focus tree generation on the feasible area of the Pareto front (denoted group I in Fig. 4) and to encourage diversity within each population. At each generation, copies of the nondominated trees are stored in an external archive and used as reference in computing fitness values for the dominated individuals [7]. Once a new nondominated set has been inserted in the archive, also called elite set, the latter is updated by eliminating the eventual dominated trees. However, a snapshot of the global elite set at each generation, after the update stage, is stored for later use.

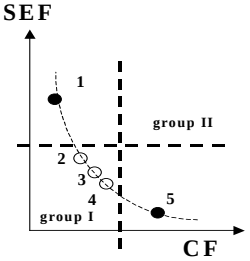


Fig. 4. Global elite set clustered to highlight feasible solutions

The feasible region of the Pareto front contains candidate models of practical significance (group I in Fig. 4). In order to identify it, the average performances of the population are computed relative to each objective [7], thus partitioning the elite set in two groups. The individuals within the first, namely elites 2, 3 and 4 in Fig. 4, are allowed to generate offspring of their own, separately from the trees in the regular population. The resulting children are then injected among the common individuals, to draw the trees closer to the interest area of the Pareto front.

Note, however, that elites are merely copies of nondominated individuals and there is a good chance that some of the originals still exist and produce offspring in the general population, although that is not a certainty. Ergo, some of the elite children being injected might resemble already existing trees. To prevent the insertion of redundant genetic material, the authors have implemented a similarity analysis based on *SEF* variance, to validate elite children inclusion in the common population [7]. If, in spite of that precaution, overall diversity drops under a certain level, the current genetic pool is refreshed by including one of the elite set snapshots from a previous generation, determined dynamically in accordance with the magnitude of diversity decrease [7].

VI. Application

The described multiobjective, elitist, GP based tool, featuring **S**imilarity **A**nalysis and **D**ynamic fuzzy **E**ncapsulation (SADE) was employed to identify a complex industrial nonlinear system, namely the steam subsection of the evaporation station within the sugar factory of Lublin, Poland. The targeted system features one input (steam temperature) and one output (steam pressure) and has no available mathematical model. The SADE algorithm was run for five times, each considering a different size of the initial population. A reference MOO approach (RMOO), featuring none of the suggested upgrades was also run under the same initial conditions, for the sake of comparison. The results, obtained on a 190 data point training set and validated on a 290 entry set, are shown in Table 1.

TABLE1: RMOO AND SADE PERFORMANCES

run	*ind_ no*	*RMOO*				*SADE*			
		gen	*elite _no*	*var* (*SEF*)	*mean* (*reg*)	*gen*	*elite _no*	*var* (*SEF*)	*mean* (*reg*)
R1	20	75	5	0.820	5	23	18	0.623	7
R2	75	100	7	1.003	12	55	23	0.429	9
R3	150	100	12	7.115	7	72	34	0.515	6
R4	300	100	15	21.31	35	63	29	0.329	8
R5	570	100	13	105.7	41	71	31	0.418	9

ind_no - initial population size (number of trees);
gen - number of generations in which the final elite set is output;
var(*SEF*) - *SEF* variance for the final elite set, computed on the validation data;
mean(*reg*) - average number of regressors in the trees in the final elite set.

TABLE II STATIC VS DYNAMIC CONFIGURATION OF MF PARAMETERS

gen	static v_i			dynamic v_i		
	reg(0)	*reg*(1)	*var* (*SEF*)	*reg*(0)	*reg*(1)	*var* (*SEF*)
60	7	5	0.920	13	12	0.725
65	12	3	0.325	5	9	0.255
70	13	7	0.213	3	19	0.203

gen - current generation
reg(0) - number of regressor with encapsulation probability 0
reg(1) - number of regressor with encapsulation probability 1
var(*SEF*) - *SEF* variance

As the number of trees in the initial population increases, the RMOO algorithm finds it difficult to handle the excess genetic material and cannot complete the evolution process (R2→R5 exit by reaching the maximum number of generations). The number of elite individuals on the final non-dominated fronts seems to increase when the size of the initial population is greater, which is not a desired behavior, as the identification problem remains the same, and should have resembling solutions, regardless of algorithm configuration. In addition, elite accuracy and complexity values are scattered over the objectives space, even in the regions of no practical relevance, as shown by the high values of the variance and average indicators. On the other hand, the SADE alternative manages to meet the accuracy termination criterion before the maximum number of generations expires, due to its elitist nature and the preservation of well adapted regressors via encapsulation. Due to the diversity conservation enhancements, the elites in the final set are much more evenly distributed than in the case of RMOO, as indicated by the low *SEF* variance values. Complexity is quasi constant on all runs, as a positive side effect of the two enhanced genetic operators.

To illustrate the contribution of the dynamically computed MF parameters, the SADE algorithm was compared against a similar version with the sole difference that the latter used static v_i values (Table II). When static MF parameters are used, there seems to be no dependency between the number of regressors considered to be well adapted, *reg*(1), and the evolution of *SEF* variance, which is an indicator of a high miss rate throughout the encapsulation process. The SADE version does better. As *var*(*SEF*) drops, which illustrates the generation of more accurate populations, ergo an increase in well adapted terms occurrence, the number of 0 labeled regressors also decreases, while the 1 labeled ones become more frequent.

The final nondominated sets generated by the RMOO and SADE algorithms, as result of a separate experiment that targeted the steam subsection described above, are presented in Fig. 5. Fig. 6 depicts the generalization capacities of the most accurate elite situated on the SADE generated Pareto front. The considered size of the initial population is 150 individuals, and the maximum allowed number of generations is 250. The raw tool, featuring none of the suggested enhancements, has produced tradeoff solutions throughout the entire span of the Pareto front, including the nonfeasible extremities. The generated models are clustered,

leaving parts of the interest zone uncovered. The SADE alternative offers a much better distributed set of solutions, located only in the feasible region of the Pareto front. Also note that the proposed identification tool provides the tradeoff set in less than half the run time of the RMOO, measured in generations, thus illustrating the practical benefits of the implemented upgrades.

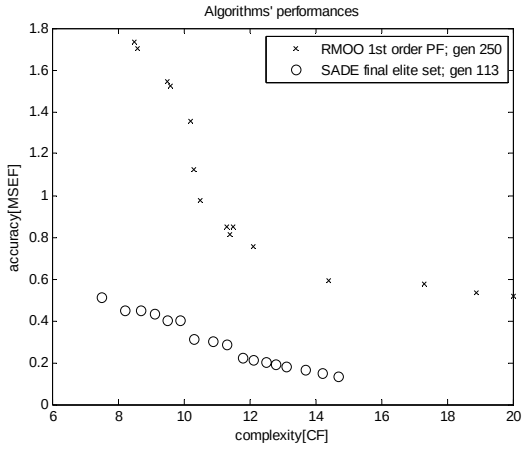


Fig 5. RMOO and SADE generated Pareto fronts

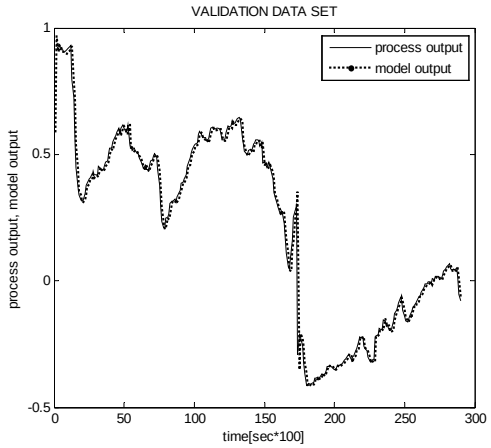


Fig 6. Validation set performances of a SADE generated model

VII. CONCLUSIONS

The nonlinear systems identification tool suggested in this paper is based on an elitist, multiobjective GP algorithm, enhanced with a fuzzy controlled dynamic encapsulation procedure and similarity analysis. The implemented upgrades are aimed at controlling model complexity, whilst balancing exploration and exploitation tendencies.

The protection of well adapted nonlinear terms is achieved by evaluating regressor usefulness via a fuzzy controller with dynamically configured parameters. The resulting encapsulation probabilities are attached as labels to each of the targeted regressors and afterwards used to guide crossover in its search for appropriate cut nodes. Search space exploration is also encouraged by means of discarding redundant genetic material with the help of similarity analysis. Solution diversity is upkept by refreshing the population with specifically chosen individuals, whenever necessary.

As the generated solutions are compliant with the NLP formalism, a demonstrated universal approximator, formally compatible with numerical applications, the suggested algorithm is a valid approach for nonlinear model generation, in the framework of automatic control problems. The QR hybridization, as well as the featured upgrades targeted at increasing search speed and reducing computational resource consumption, recommends the proposed approach for solving complex problems with reduced pre-design available information and difficult search space landscapes.

REFERENCES

[1] M. Young, The Stone-Weierstrass Theorem, MATH 328 Notes, Queen's University at Kingston, winter term, 2006. Available: http://www.mast.queensu.ca/~speicher/Section14.pdf.

[2] H. Wey, S. A. Billings, J. Lui, "Term and Variable Selection for Nonlinear Models", *Int. J. Control* 77, 2004, pp. 86-110.

[3] O. Nelles, Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models, Springer-Verlag, 2001.

[4] J. R Koza, Genetic Programming – On the Programming of Computers by Means of Natural Selection, Cambridge, MA: MIT Press, pp. 73-190, 1992

[5] T. Back, D. Fogel, Z. Michalewicz, Evolutionary Computation 2.Advanced Algorithms and Operators. US: Institute of Physics Publishing, 2000.

[6] R. Poli, W.B. Langdon, N.F. McFee, J.R. Koza, A Field Guide to Genetic Programming. Published via http://lulu.com and freely available at http://www.gp-field-guide.org.uk, 2008.

[7] A. Patelli, L. Ferariu, Elitist Multiobjective Nonlinear Systems Identification with Insular Evolution and Diversity Preservation, IEEE World Congress on Computtaional Intelligence, July, Barcelona, 2010 - accepted paper.

[8] P. J. Flemming, R. C. Purshouse "Evolutionary Algorithms in Control Systems Engieering: A Survey", *Control Engineering Practice* 10, 2002, pp. 1223-1241.

[9] J. Madar, J. Abonyi, F. Szeifert Genetic Programming for System Identification [Online]. Available: http://www.fmt.vein.Hu/softcomp/ isda04_gpolsnew.pdf, 2005.

[10] K. Rodriguez-Vasquez, C. M. Fonseca, P. J. Flemming Identifying the Structure of Nonlinear Dynamic Systems Using Multiobjective Genetic Programming. IEEE Transactions on Systems Man and Cybernetics, Part A – Systems and Humans, 34, 531-534, 2004.

[11] K. Deb, Multi - Objective Optimization using Evolutionary Algorithms, Wiley, USA, 2001

[12] L. Ferariu, A. Patelli Multiobjective Genetic Programming for Nonlinear Systems Identification, in curs de publicare, Lecture Notes on Compuer Science, Springer Verlag, 2009.

[13] Adra S. F., Dodd T., Griffin I. A., Fleming P., Convergence Acceleration Operator for Multiobjective Optimisation, IEEE Transactions on Evolutionary Computation, 13 (4), 825-846, 2009

[14] L. Rachmawati, D. Srinivasan, Multiobjective Evolutionary Algorithm with Controllable Focus on the Knees of the Pareto Front, IEEE Transactions on Evolutionary Computation, 13 (4), 810-824, 2009.

[15] C.A. Coello Coello, G.B. Lamont, D.A. Van Veldhuizen, Evolutionary Algorithms for Solving Multi-Objective Problems, second edition, Springer, 2007.

Benefits of Virtualization in HMI/SCADA Systems

Miloš Pavlík[1], Iveta Zolotová[1], Rastislav Hošák[1], Lenka Landryová[2]

[1]Dept. of Cybernetics and Artificial Intelligence, FEI TU Košice, Slovak Republic,
{milos.pavlik, iveta.zolotova, rastislav.hosak}@tuke.sk
[2]Dept. of Control Systems and Instrumentation, FME TU Ostrava, Czech Republic,
lenka.landryova@vsb.cz

***Abstract* - Virtualization is becoming a standard information technology practice. Virtualization solutions are being broadly used in development and production environments, providing tremendous benefits like scalability, flexibility and ability to reduce the maintenance costs. It allows most efficient using of the computer resources. The point of article is to show how we used the virtualization technology for HMI/SCADA (Human Machine Interface/Supervisory control and data acquisition) research and education. HMI/SCADA applications are using a lot of different servers and clients which are localized in laboratory or remotely. There are OPC servers for real models, Wonderware Application Server, Wonderware Information Server and few local servers at department, or in laboratory. Because of new trends in technology and knowledge is necessary to upgrade the overall architecture and one of the benefit solution is to use virtualization technology.**

I. INTRODUCTION

Virtualization technology has made big improvement in allowing the creation of the next generation of efficient, easily manageable, highly available, flexible and dynamic data centers. Future developments in virtualization as a technology, the processes involved and hardware technology employed will be very interesting. This technology can be used in almost any company, as long as you have more than one server. In addition to workload consolidation, other benefits of virtualization include high availability, live migration, streamlined backups and fault tolerance.

Virtualization abstracts the underlying resources such as the memory, storage, network so that multiple operating systems can be run on a single physical system at once. This improves resource utilization, because normal servers utilize about 20% of its resources, whereas a virtual machine uses an average of 80% of its resources. Server virtualization and storage virtualization lead to decreased energy consumption, which automatically includes them in the list of green technologies and Eco-friendly technologies [8].

One of the undisputed advantages of virtualization is that it makes migration much easier. When virtualization is used and there is need to deploy a server on a different machine, the only thing to do is to copy the installation of the virtualized environment to another computer.

II. VIRTUALIZATION TECHNOLOGY

A. Virtualization

Virtualization is the creation of a virtual version of something, like an operating system, a server, a storage device or network resources. Divided the hard drive into different partitions is some kind of virtualization, so it isn't so unknown term. So virtualization in its core is involved with adding "layers of abstraction" to the IT environment. These layers decouple one layer from another [2].

These are examples of virtualization technology that is used:

- **Hardware virtualization** adds a layer between individual operating systems and the physical hardware to which they are installed. This layer allows more than one operating system to interact with the same hardware.
- **Operating system virtualization** (container-based virtualization) adds its layer on the top of existing operating system. Result is that multiple operating system instances that use the same source drain less resource overhead.
- **Network virtualization** it is very similar to hardware virtualization and it's a method where all available resources in a network are combined by splitting up the available bandwidth into channels, each is independent from the others, and each can be assigned (or reassigned) to a particular server or device in real time. The idea is that with network virtualization we can break one physical network into multiple virtual networks or we can say that it covers the complexity of the network by dividing it into manageable parts, much like dividing the hard drive into different partitions. It also makes easier to manage files on hard drive.
- **Storage virtualization** adds its layer atop storage hardware. It is the pooling of physical storage from

multiple network storage devices into what appears to be a single storage device that is managed from a central console. Storage virtualization is commonly used in storage area networks. It enables functionalities like combining the storage capacity of multiple storage devices to install a single, logical drive to hardware

- **Server virtualization** is the masking of server resources (including the number and identity of individual physical servers, processors, and operating systems) from server users. The intention is to spare the user from having to understand and manage complicated details of server resources while increasing resource sharing and utilization and maintaining the capacity to expand later [15].

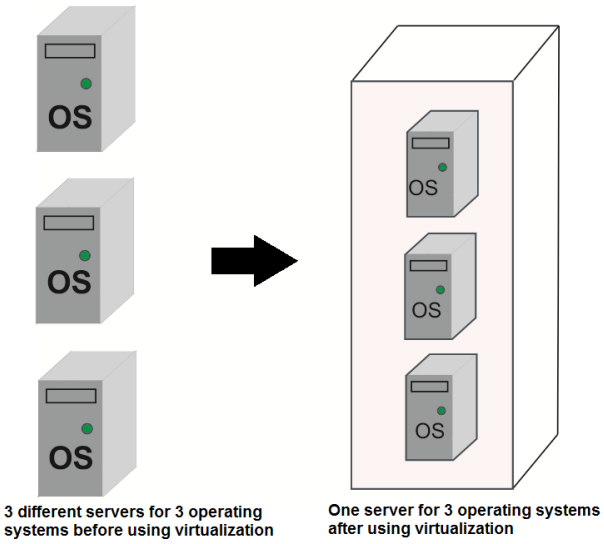


Fig. 1 Example of server virtualization

B. *Benefits of virtualization*

There are many benefits to merge the number of servers in your environment by taking advantage of the many different server virtualization products (i.e. VMware and Microsoft Virtual PC,) on the market [1]. These include:

- Lower number of physical servers - you can reduce costs, needed for hardware resources, because of a lower number of physical servers. For example, five servers are virtualized and they‘re running on one host PC [13]
- By implementing a server consolidation strategy, you can increase the space utilization efficiency in your data center
- When you have each application within its own "virtual server" you can prevent one application from impacting another application when upgrades or changes are made
- You can develop a standard virtual server build that can be easily duplicated which will speed up server deployment
- You can deploy multiple operating system technologies on a single hardware platform (i.e. Windows Server 2003, Linux, Windows 2000, etc)

Virtualization has many benefits. As we said, if the applications use the same server for behavior, they can invade each other, for example because of updating or one of the applications CPU (Central processing Unit) usage is at hundred percent, etc. The risk is greatly reduced by dividing the applications onto different virtual computers. Virtualization gives flexibility, so the operating system does not depend on particular piece of hardware. There are no changes needed for, for example moving the virtual computer. It ensures high scalability of systems. It is not necessary to build a 'final solution' because systems can be expanded on the fly, without prejudice to operate services. By using virtualization technology, it’s very easy to backup the systems and it’s great to simulate technically difficult solutions without need to own or ensure additional hardware

C. *Virtualization in HMI/SCADA*

The basic idea behind virtualization in this context is to work on logical hardware in a bit of a sandbox. Another nice feature is working from images (snapshots) instead of entire hard drives and machines. Imagine building your HMI (Human Machine Interface) exactly how you want, then taking a snapshot. With virtualization, you can run multiple instances of this. SCADA (Supervisory control and data acquisition) installation is an image file that can be run on any computer. Maybe you want to consolidate hardware, or maybe you want a similar environment for development. The concept of "create once, use many" applies here.

If you have an aging SCADA system running on a no-longer-supported hardware or operating system platform, then creating a virtualized environment that lets you continue to support that system software may be a viable strategy. But be aware that virtualized peripherals can also be an issue. Not basic things like Ethernet ports and file systems, but anything exotic (like synchronous serial ports) may not be supportable in a virtual emulation [14].

It can be a big help in supporting legacy HMI/SCADA technologies. It's really useful for programs that are difficult to configure. We could set up a virtualized "production" and "testing" environment on the same computer. But this is insignificant because each installation is better to be separate, and each can support the entire network on a desktop PC [10].

D. *Security in virtualized systems*

Security has a big profit from virtualization and on the

other side virtualization has a lot to lose if it has no security controls. Basically, it is security that has the physical layer abstracted. One easy example is the ability to take a single physical firewall and partition it into multiple virtual firewalls to serve different applications.

The reason, why security and virtualization are discussed a lot, is that server virtualization is moving beyond the development environment and into production. In a production setting, a lot of great ideas that seemed great in development are running into deficiency by the security team.

Some of the old concepts have to go, within the virtualized environment, for example, IP addresses do not identify servers because servers can be redeployed on-the-fly to a different subnet. So your "IP X.X.X.X can send packets to IP Y.Y.Y.Y" access control design is no longer relevant or helpful. Things that were on IP X.X.X.X have moved to a different subnet or data center. Dynamically allocated virtual servers need dynamically allocated virtual security [20].

The SCADA and process control security issues are:

- **Spoofing countermeasures:** To prevent masquerading attacks and to maintain confidentiality and data integrity for PLC and sensor data
- **Identification and authorization:** For both users and data, "to make sure the data is authentic" between devices, sensors, PLCs, controllers and up the manufacturing hierarchy, including human users
- **Logging and auditing:** To provide a forensic capability if something goes wrong, with time and date stamps
- **Encryption:** Voluntary encryption for sensitive or private information, where necessary.
- **Default security:** Products need to come secure from the vendor "out of the box" with security turned on by default
- **Physical security:** To maintain the integrity of the system
- **Policies and procedures:** To provide for secure management practices [17]

III. VIRTUALIZATION FOR EDUCATION AND RESEARCH

A. *Goals and analysis*

As virtualization technology progressed, we decided to use them in laboratory at the Department of Cybernetics and Artificial Intelligence at Technical University in Kosice. A lot of different servers are in use at the department in research and educational process. The most important are Wonderware Application Server, Wonderware Historian Server, Microsoft SQL Server 2005 and OPC (Object Linking and Embedding for Process Control) servers for real models in laboratory.

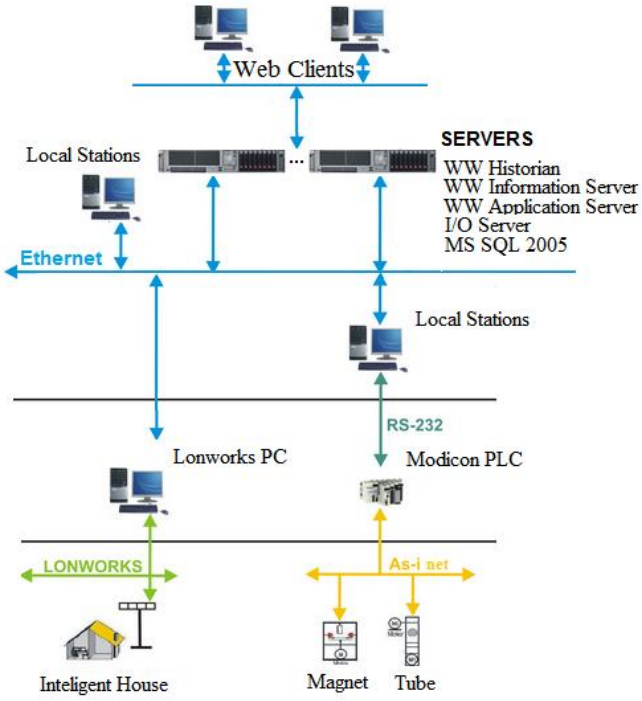


Fig. 2 Servers and their connection used in laboratory

So how we can see, there are at least 5 servers in our laboratory, so the maintenance of them is quite expensive and the hardware already is slowly but surely getting older. So we decided that within these innovations, we will install these servers into virtual machines, which all of them will be on one physical host computer. There are few real controlled models in laboratory (intelligent house, tube, magnet), that use these servers, and also a few virtual-simulated models. All of these models are for research, presentation and educational purpose.

This decision was clearly from the perspective of what virtualization offers. The ability for creating the duplicate servers can be used for all installed servers. One is for development and for deployment for specific applications. The second is for runtime and also for demonstration and promotional purposes. This feature was used for Wonderware Application Server. The Application Server is one of the main products of Wonderware Software Company It offers new possibilities with the new ArchestrA infrastructure usage, which is available in real time in an industrial environment. Industrial Application Server is built on the idea of a modular approach and brings the directness and expandability for users. The basis of this product is an object-oriented technology and programming based on templates. Management of the Wonderware Application Server covers requirements for industrial automation and information applications and

different types of HMI visualization applications over wide supervisory SCADA applications to systems for managing and analyzing the operation and performance of production (MES-Manufacturing Execution Systems).

B. *Realization*

As virtualization tool we use MS Virtual PC 2007, in which we've created individual virtual machines. On two of them have been installed MS Windows Server 2003 standard edition, which has a full support in Microsoft Virtual PC 2007. On the third virtual PC was installed the Microsoft Windows XP Professional operating system. Of course, the count of created virtual machines is limited by host computer hardware [1]. For example, if we want to install and use five virtual machines, which each need 1 GB of RAM (Random Access Memory) and 20 GB of hard drive for virtual hard drive and the host computer has only 4GB of RAM and 60GB of disk space, it's impossible to run all of virtual machines at once.

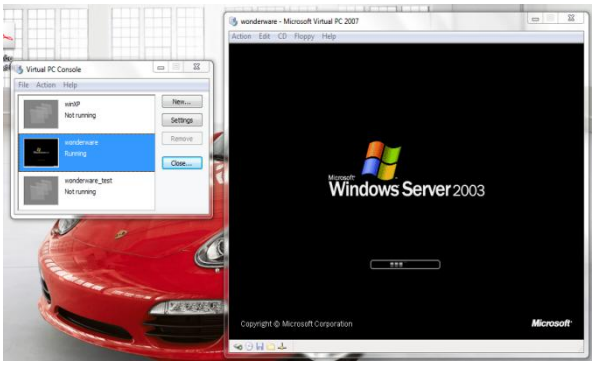


Fig. 3 Virtual PC 2007

Virtual PC uses virtual machine technology to run two or more operating systems and their related applications at the same time (Fig 3.). This tool offers enough possibilities to setup a virtual PC (Fig 4.). There are abilities like setting the size of Memory allocated to a virtual PC, size of the virtual disk and the count of virtual disks, ability to configure I/O (Input/Output) ports and setting the count and types of network adapters or in other words, abilities for networking. Other emulated hardware is DMA (Direct Memory Access) controller, IDE/ATA (Integrated Drive Electronics/Advanced Technology Attachment) controller and keyboard controller.

Virtual PC uses the host PC operating system to interact with external devices such as CD-ROM, floppy, keyboard, mouse or physical display.

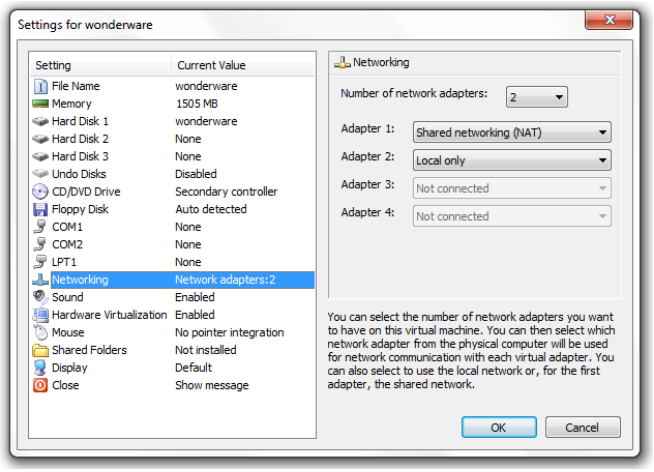


Fig. 4 Virtual PC settings

In the virtual machine where the Microsoft Windows server 2003 standard edition operating system was installed is allocated 1GB of memory and virtual disk of size of 17 gigabytes. There can be allocated multiple virtual disks for individual virtual machines with the fact that we may use existing virtual drives, or create a new. While creating of virtual disk we need to consider what type of virtual disk will be used. Virtual PC supports dynamically expanding virtual hard disks, Fixed-size virtual hard disks and Linked virtual hard disks. We used fixed size, which, as it was mentioned, has the size of 17 gigabytes, which is fully sufficient [2].

Virtual PC tool also supports the further possibilities of virtual disks, as differencing virtual hard disks, which allows multiple users and multiple virtual computers use the same parent virtual hard drive at the same time. Another option is to undo disks. This is very important feature, because undo disks allow users to delete any changes they make to the virtual hard disk during a session (a session lasts from the time the virtual machine is started until it is closed). Virtual PC stores these changes in a separate temporary file, and then at the end of the session, users can save, commit, or delete the changes. Undo disks allow users to start with the exact same virtual disk every time.

The communication of virtual machines was solved by assignment of network adapter through which the host computer is accessing the network with internet access (WAN). In the networking features in Virtual PC settings we assigned to the virtual machine a network card Broadcom Netlink Gigabit Ethernet. Through this option the virtual station appears like another physical computer in the network, and it is automatically assigned with IP address (Host IP: 147.232. 61.8, Virtual Machine, 1 IP: 147.232.61.9, Virtual Machine 2, IP: 147.232.61.10, Virtual Machine 3 IP: 147.232.61.11). So in the fact, it's possible to connect with virtual machine from a random client computer from network. The third machine has only local networking. It means, that the virtual machine communicates with other

running virtual machines on the local network. No traffic transmits over the wire to other computers and no traffic is exchanged with the host operating system [4].

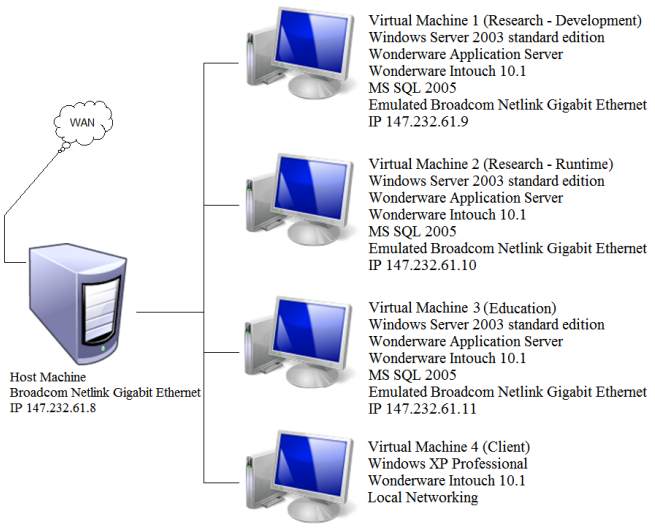


Fig. 5 Virtualization of Wonderware Application Server

Now when all is working fine, it's necessary to install the required software. The software that was installed is Microsoft SQL 2005, Wonderware Intouch 10.1 and Wonderware Application server with ArchestrA IDE (Integrated Development Environment). After installing the software and after adjustment of all features, like setting shared folders and network accounts, it was created a Galaxy (the database where are stored all configuration data, animation objects, variables of animation objects, which belong to projects of one galaxy) called "L509", to which was no longer a problem to connect from any PC on the network. Of course to an integrated development environment IDE can access multiple developers at once.

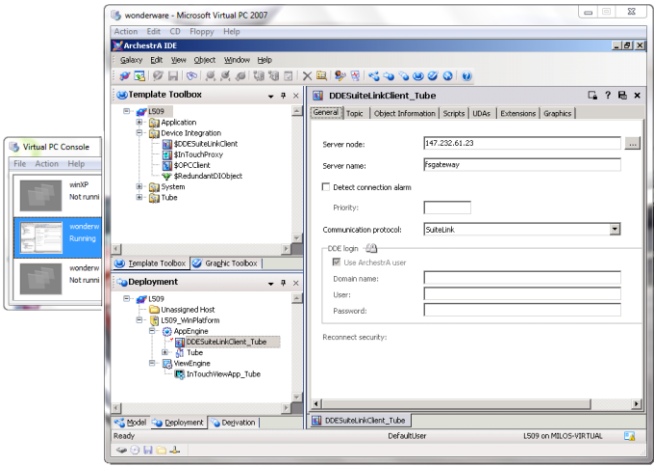


Fig. 6 Galaxy (ArchestrA IDE) running from Virtual PC

C. Evaluation

The biggest benefit for our laboratory is that we can work on development, in the runtime for demonstrations and teach on different platforms independently. Another benefit is that we, as control and informatics specialists can build, operate and maintain virtualized solutions for HMI/SCADA. Of course, the maintenance costs are reduced. Virtualization technology itself offers great ability to backup the systems and great scalability. In general, there are few disadvantages. The biggest one is that we need a performance server that will meet the requirements for our application.

Virtualization technology is very useful in educational process. For example, we have students who are in groups. Students in class have their physical computers with installed virtual machine where is installed operating system and, for example, Wonderware Intouch. The great addition is that students have administrator rights in OS on virtual machine, so they don't need administrator rights for host OS. So they will not bring any damage to host OS and after log-of, virtual machine will start with the exact same virtual disk every time and with no changes, so in fact after log-off the virtual machine will restore its condition.

Generally this technology brings a lot of benefits and most of them are very interesting for department. Few servers at the department are old and maintenance is, in comparison with the possibility to virtualize them and all run at once on a single host PC, expensive.

IV. Conclusion

Virtualization technology has now become a part of department and laboratories and in future several servers will be virtualized. The technology described in this paper brought new opportunities for research and educational process, for working in laboratories that are virtualized. Students have administrator privileges on the virtual OS and are not restricted when they work on projects. Virtualization technology is a big addition in supporting HMI/SCADA technology and applications, e.g. it means that if we have a legacy technology or application running on no-longer supported hardware or software a virtualized environment lets you to continue to support that system software. We have designed and implemented a new hardware, network and software architecture in the laboratory using virtualization benefits, which was applied in a complex hierarchical information-control system with emphasis on control level HMI/SCADA.

Acknowledgment

This publication is supported by grant KEGA 037-011 TUKE-4/2010 and VEGA - 1/0617/08. This work was concluded in our laboratory with the software and licenses provided by Factory Suite Educational Demo Consignment from Pantek (CS) s.r.o.

REFERENCES

[1] R. Carswell, H. Webb, "Guide to Microsoft Virtual PC 2007 and Virtual Server 2005",2008, pp. 462.

[2] D. Rule, R. Dittner,"Best Damn Server Virtualization book period", 2007, pp. 931.

[3] M. Babiuch," The Usage of the New Technologies at the Education at the Department of Control Systems and Instrumentation", 2006, pp. 7-12. ISSN 1210-0471. ISBN 80-248-1211-8.

[4] D. Popescu, Q. Meng, A. A. Ionete,"Remote vs. Simulated, Virtual or Real-Time Automation Laboratory" in *International Conference on Automation and Logistics Shenyang*, China, August 2009.

[5] J. Honeycutt, "Microsoft Virtual PC Technical Overview", 2007, pp. 26.

[6] P. Božek," Complex Control and Metrology Security of Automatized Trial System", *Manufacturing Engineering*, number 2, volume IV, 2005, p. 31-35, Prešov, ISSN 1335-7972.

[7] M. Dobriceanu, A. Bitoleanu, M. Popescu, S. Enache, E. Subtirelu," SCADA system for monitoring water supply networks.", *WSEAS Transactions on Ssystems* 7(10): 1070-1079, 2008. ISSN 1109-2734.

[8] M. Franeková," Safety and security profiles of industry networks used in safety-critical applications", *Transport problems*. Volume 3 Issue 4 Part 1, pp. 25-32, Gliwice, 2008, Poland, ISSN 1896-0596.

[9] G. Hields, "The shortcut guide to Virtualization and Service Automation", 2009, pp. 67.

[10] L. Landryová, I. Zolotová," Challenges and Software Aspects of Remote Labs for Engineering Education." in *Proceedings of 8 the International Conference on Information Technology Based Higher Education and Training*. Kumamoto, Japan, available online at URL: http://ithet07.coe.kumamoto-u.ac.jp, July 10-13.2007, pp 1-4.

[11] L. van Dijk. (2008, October 28) IT Computing. Available: http://www.anandtech.com/print/2653

[12] http://global.wonderware.com/EN/Pages/WonderwareSystemPlatform.aspx

[13] http://www.itworld.com/nls_windowsserver050411

[14] T. Tomshaw. (2009, June 23). Available: http://scadaperspective.com/pipermail/scada_scadaperspective.com/2009-June/001597.html

[15] http://searchservervirtualization.techtarget.com/sDefinition/0,,sid94_gci499539,00.html

[16] http://www.windowsecurity.com/articles/Security-Virtualization.html

[17] M. Willoughby. (November 18). Available: http://www.computerworld.com/s/article/97606/New_security_standards_to_strengthen_SCADA

[18] A. Samoilenko. (2007, October 9). Available: http://ixbtlabs.com/articles2/cm/virtualization-vpc-vserv-page1.html

[19] http://virtualizationconversation.com/2009/02/26/the-future-of-virtualization/

[20] http://www.microsoft.com/virtualization/en/us/products-security.aspx

Altered Fingerprints Analysis Based on Orientation Field Reliability

Adina Petrovici, Corneliu Lazar, *Member, IEEE*

***Abstract* — Fingerprint recognition is one of the most commonly used biometric technology. Even if fingerprint temporarily change (cuts, bruises) it reappears after the finger heals. Criminals started to be aware of this, and try to escape the identification systems applying methods from ingenious to very cruel. It is possible to remove, alter or even fake fingerprints (made of glue, latex, silicone), by burning the fingertip skin (fire, acid, other corrosive material), by using plastic surgery (changing the skin completely, causing change in pattern – portions of skin are removed from a finger and grafted back in different positions, like rotation or 'Z' cuts, transplantations of an area from other parts of the body like other fingers, palms, toes, soles). This paper presents a new method for altered fingerprints analysis based on fingerprint orientation field reliability. The map of the orientation field reliability has peaks in the singular point locations. These peaks are used to analyze altered fingerprints because, due to alteration, more peaks as singular points appear with lower amplitudes.**

I. Introduction

BIOMETRIC recognition (or simply biometrics) refers to the use of distinctive anatomical (e.g., fingerprints, face, iris) and behavioral (e.g., speech) characteristics, called biometric identifiers or traits or characteristics for automatically recognizing individuals.

Every human being possesses fingers (with the exception of hand-related disability) and hence fingerprints. Fingerprints are very distinctive and they are permanent; even if they temporarily change slightly due to cuts and bruises on the skin, the fingerprint reappears after the finger heals.

Offenders, being well aware of this, have been coming up with ways to escape identification by that means. Erasing left over fingerprints, using gloves, fingerprint forgery are certain examples of methods tried by them, over the years.

Failing to prevent themselves, they moved to an extent of mutilating their finger skin pattern, to remain unidentified, by burning the fingertip skin using fire, acid or other corrosive material, cutting and removing certain areas of finger skin, slashing the skin and then stitching the two parts together, changing the skin completely by means of plastic or cosmetic surgery, chewing the skin off, and so on.

In general, obliterating fingerprints is still known to be very uncommon. In contrast to this, if one takes a look at the web world, it would be realized that it is not very far when such things will become common place very soon.

This paper is based upon obliteration of finger ridge patterns highlighting the reasons why offenders go to an extent of performing such act. The goal is to understand the problem of altered fingerprints and to design solutions that can be used to detect these images. The method proposed is based on analyzing the ridge orientation field, to compare the differences between natural fingerprints and distorted ones, by calculating the orientation field reliability of the enhanced image.

Three types of degradations of the ridge patterns may be distinguish: *obliteration*, *distortion* and *imitation*. Obliterated fingerprints refer to the fingerprints resulting from abrading, cutting, burning, applying strong chemicals or transplanting smooth skin. Friction ridge patterns can be turned into distorted (unnatural) ridge patterns by plastic surgery. The problem of identifying altered fingerprints was introduced by Feng, Jain and Ross in [1]. Imitation of the fingerprint is also made by plastic surgery but this involves the transplantation of a large-area friction skin from other parts of the body such as other fingers, palms, soles, toes.

Unlike the problem of fingerprint alteration, the use of fake fingerprints has received increased attention in the literature. Methods of life detection have been proposed based on physiologic characteristics (pulse, temperature, odor, dielectric resistance) or behavioral characteristics (deformation of the skin, diffusion of sweat). Antonelli, Cappelli, Maio, and Maltoni proposed an approach based on the analysis of human skin elasticity in [2]. Distortion codes, that encode informations obtained in the feature extraction stage, are analyzed to determine the nature of the finger [2]. A method based on Radon transform was proposed by Imamverdiev, Kerimova, and Mussaev in [3]. It includes a fuzzy classifier of *c*-means on the basis of statistical characteristics of transformed images that calculates the required threshold to differentiate between true and false fingers. Baldisserra, Franco, Maio, and Maltoni proposed an approach to secure fingerprint scanners against the presentation of fake fingers in [4]. An odor sensor (electronic nose) is used to sample the odor signal and an ad-hoc algorithm allows to discriminate the finger skin odor from that of other materials such as latex, silicone [4].

Altered fingerprints are considered as biometric obfuscation, which is defined as a deliberate attempt by a person to mask his identity from a biometric system. A study regarding finger prints obfuscation problems is presented in

Manuscript received September 1, 2010. This work was supported in part by the EURODOC - "Doctoral Scholarships for research performance at European level" project, financed by the European Social Fund and Romanian Government.

A. Petrovici and C. Lazar are with the Automatic Control and Applied Informatics Department, Technical University "Gh. Asachi" of Iasi, Romania. (E-mails: adina.tampau@yahoo.com; clazar@ac.tuiasi.ro)

[1]. In the same paper, an automatic detection algorithm of altered fingerprints is given. The algorithm is based on detecting the singular points using Poincaré index from the orientation field of a fingerprint. Then, the singular points are removed in order to obtain a continuous orientation field from which a feature vector is extracted and used as input to a support vector classifier for distinguishing between natural and altered fingerprints.

This paper presents a new method for altered finger prints analysis based also on the orientation field of fingerprints. Using an orientation field, singular points are detected following an approach from [6] that uses fingerprint orientation field reliability. The orientation field reliability map has peaks in the singular point locations. These peaks are used to analyze altered fingerprints because due to alteration, more peaks as singular points appear with lower amplitudes. The experimental results on images from FVC2004, DB1, show that locating the position of singular point may be useful in detecting altered fingerprints.

II. Ridge Orientation Field Estimation

The orientation field image is a Level 1 feature that represents the angle $\theta_{i,j}$, shown in Fig. 1, that the fingerprint ridges form with the horizontal axis, crossing through an arbitrary small neighborhood centered at (i, j).

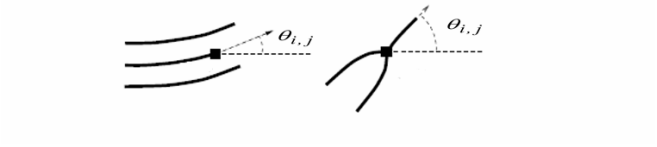


Fig. 1 Ridge ending and ridge bifurcation orientation angle

The binary image obtained after image enhancement is transformed into grayscale image using the euclidian distance transform [1].

In order to mark the foreground and background images, a blockwise (8x8 pixels) binary image is created based on the distance transform image. A block of 8x8 pixels is set as foreground if at least 80% of its pixels have values smaller than a threshold (set to 10.). For each block in the foreground, the orientation field is estimated applying a gradient-based method [5], based on the distance transform image.

A single orientation is assigned for each non-overlapping block of size *wxw* (16x16 pixels) that corresponds to the dominant orientation of the block.

The gradient G at the point (i, j) is computed as a 2 dimensional vector with the components $G_x(i, j)$ and $G_y(i, j)$ the horizontal and the vertical gradients, with respect to the *x* and *y* directions. Simple gradient operators are used such as a Sobel mask (3x3).

The local orientation of each block centered at pixel (i, j) is estimated using the following equations:

$$V_x(i,j) = \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} 2G_x(u,v)G_y(u,v),$$

$$V_y(i,j) = \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} (G_x^2(u,v) - G_y^2(u,v)), \quad (1)$$

$$\theta(i,j) = \frac{1}{2}\tan^{-1}\left(\frac{Vy(i,j)}{Vx(i,j)}\right),$$

where $\theta(i, j)$ is the least square estimate of the local ridge orientation at the block centered at pixel (i, j).

In order to adjust the incorrect local ridge orientation, due to the presence of noise, a low-pass filtering can be used, since local ridge orientation varies slowly in a local neighborhood where singular points appear. The orientation image needs to be converted into a *continuous vector field*, as follows:

$$\Phi_x(i,j) = \cos(2\theta(i,j)),$$
$$\Phi_y(i,j) = \sin(2\theta(i,j)), \quad (2)$$

where $\Phi_x(i, j)$ and $\Phi_y(i, j)$ are components of the vector field. With the resulting vector field, the low-pass filtering can be performed as follows:

$$\Phi_x'(i,j) = \sum_{u=-\frac{w_\Phi}{2}}^{\frac{w_\Phi}{2}} \sum_{v=-\frac{w_\Phi}{2}}^{\frac{w_\Phi}{2}} W(u,v)\Phi_x(i-uw, j-vw),$$

$$\Phi_y'(i,j) = \sum_{u=-\frac{w_\Phi}{2}}^{\frac{w_\Phi}{2}} \sum_{v=-\frac{w_\Phi}{2}}^{\frac{w_\Phi}{2}} W(u,v)\Phi_y(i-uw, j-vw), \quad (3)$$

where W is a 2-dimensional low-pass filter with unit integral and $w_\Phi x w_\Phi$ specifies the size of the filter (default size 5x5).

After the smoothing operation is performed at the block level, the local ridge orientation at (i, j) is computed using

$$O(i,j) = \frac{1}{2}\tan^{-1}\left(\frac{\Phi_y'(i,j)}{\Phi_y'(i,j)}\right), \quad (4)$$

where $O(i, j)$ represents the orientation image.

III. Algorithm for Altered Fingerprint Analysis

The algorithm is implemented based on analyzing the ridge orientation field, in order to compute the orientation field reliability [6]. The reliability of the orientation field describes the consistency of the local orientations in a

neighborhood along the dominant orientation; it is used to locate the unique singular point constantly for all types of fingerprints.

A pre-processing stage of fingerprint enhancement is performed in order to improve the contrast between ridges and valleys and reduce noises in the fingerprint images. Two methods are adopted for image enhancement stage: the first one is Histogram Equalization and the next one is Fourier Transform. A locally adaptive binarization method is performed to binarize the fingerprint image.

Fingerprint segmentation is implemented, to decide which part of the image belongs to the foreground and which part belongs to the background. The singular points will be located more accurately if the localization operates only on the foreground of the fingerprint image.

Based on the distance transform image, the method presented in section II is implemented to obtain the orientation field and the orientation field reliability. The least mean square method of orientation based on the gradients is most widely used to compute the dominant orientation of an image block because of its high efficiency and resolution. Since the gradient phase angle is the orientation with maximum grey value change, it is orthogonal to the local ridge orientation of each pixel.

The continuous vector field is obtained in order to perform the low-pass filtering. The Gaussian filter is applied, having the size of $w_{\Phi}xw_{\Phi}$ set to 5x5. The local ridge orientation is obtained. The method measures the strength of the peaks computing first:

$$\begin{aligned}&\min\_inertia(i,j) = G_{yy}(i,j) + G_{xx}(i,j) - \\ &\Phi'_x(i,j)G_{xx}(i,j) - G_{yy}(i,j) - \\ &\Phi'_y(i,j)G_{xy}(i,j))/2, \quad , \\ &\max\_inertia(i,j) = G_{yy}(i,j) + G_{xx}(i,j) - \\ &\min\_inertia(i,j),\end{aligned} \tag{5}$$

where:

$$\begin{aligned}G_{xx}(i,j) &= \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} G_x(u,v)G_x(u,v), \\ G_{yy}(i,j) &= \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} G_y(u,v)G_y(u,v), \\ G_{xy}(i,j) &= \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} G_x(u,v)G_y(u,v).\end{aligned} \tag{6}$$

Having *min_inertia* and *max_inertia* from (5), the reliability R is given by:

$$R(i,j) = 1 - \frac{\min\_inertia(i,j)}{\max\_inertia(i,j)}. \tag{7}$$

Based on the method described for field orientation estimation and the computation of the reliability R, the following algorithm is proposed, for altered fingerprint analysis:

Stage 1: Segmentation

For each block of size 8x8 pixels belonging to the distance transform image

If 80% of the pixels belonging to the block have grayscale value > threshold (set to 10)

Set pixels (i,j) to 1

Else

Set pixels (i,j) to 0

End if

End for

Stage 2: Orientation field before filtering and continuous vector field

For each pixel (i,j) in the distance transform image compute the horizontal and vertical gradients $G_x(i,j)$ and $G_y(i,j)$, using Sobel mask (3x3)

For each block of size wxw (16x16) centered at pixel (i,j)

If the block belongs to the foreground

Compute $V_x(i,j)$, $V_y(i,j)$

Compute $G_{xx}(i,j)$, $G_{yy}(i,j)$, $G_{xy}(i,j)$

Compute $\theta(i,j) = \tan^{-1}\left(2(V_y, V_x)/2\right)$

Compute components of the continuous vector field $\Phi_x(i,j)$ and $\Phi_y(i,j)$

End if

End for

Stage 3: Orientation field and reliability orientation image

For each block of size wxw (16x16) centered at pixel (i,j)

If the block belongs to the foreground

Compute $\Phi'_x(i,j) = sum(sum(W(i,j)\Phi_x(i-uw, j-vw)))$

Compute $\Phi'_y(i,j) = sum(sum(W(u,v)\Phi_y(i-uw, j-vw)))$

Compute $O(i,j) = a\tan 2(\Phi'_y(i,j), \Phi'_x(i,j))/2$

Compute $\min\_inertia(i,j)$ and $\max\_inertia(i,j)$

Compute $R(i,j)$

End if

End for

The algorithm for altered fingerprint analysis has been implemented in MATLAB.

By observing the plot in the three-dimensional space of

the reliability R obtained from the proposed method, we can identify high peak amplitudes that can be used to detect natural fingerprints and more smaller peak amplitudes that can be used to detect altered fingerprints.

IV. Experimental Results

The proposed algorithm for fingerprint analysis based on the estimation of orientation field and the computation of the reliability was tested on natural fingerprint images from FVC2004, DB1, captured with optical sensor, with size 640x480, at a resolution of 500dpi. Three different fingerprint images from two major categories (arch and loop) are presented in this section.

The natural fingerprints are compared with the altered fingerprints, by observing the values of the reliability singular points. In this paper, the singular point is defined as the point with maximum curvature on the convex ridge. For natural fingerprints, the reliability orientation image generally has one sharp peak, while in altered fingerprints more peaks are detected with smaller values. Starting from this observation, the altered fingerprint analysis can be done using the density and the amplitude of the peaks of the singular points.

Due the lack of altered fingerprints images, the experimental results for the distorted fingerprints are obtained based on synthetically altered images.

Two types of alterations are simulated:

(i) 'Z' cut (obtained by making a 'Z' shaped cut on the fingertip, lifting and switching the two triangles, and stitching them back);

(ii) central rotation (obtained by cutting a circular region and rotated by different degrees).

Initially, the arch type of a natural fingerprint and the corresponding altered forms obtained by central rotation and 'Z' cut together with their orientation fields are considered in Fig. 2. The orientation field reliability map and the peak of the singular points for an arch type fingerprint are represented in the Fig. 3. The peak from the natural fingerprint in Fig 3(a) has a high value of 2644. The two altered fingerprint images, have more peaks, but with small values between 641 for the highest positive peak and -461 for the negative peak.

The second analysis was done considering the loop type

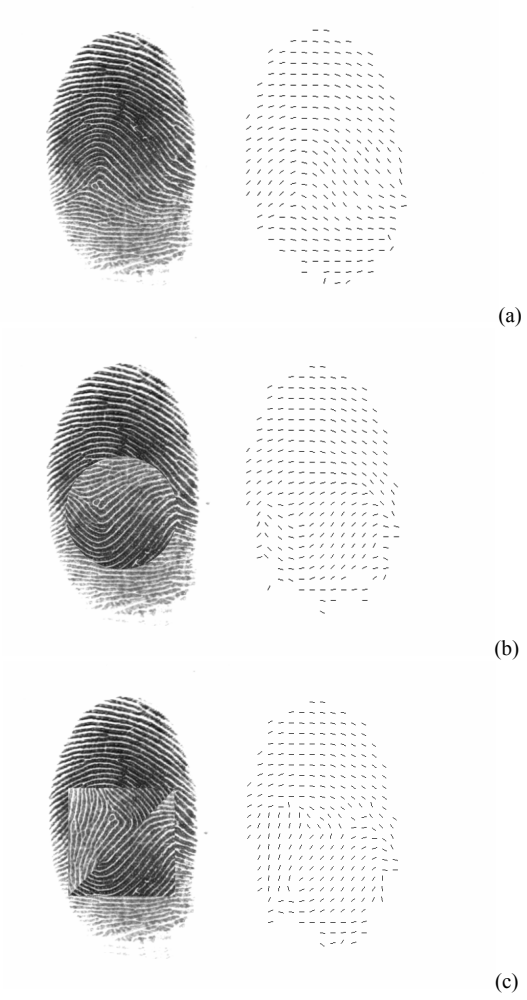


Fig. 2 (a) Arch type natural fingerprint and its orientation field estimation; (b) Altered fingerprint from (a) by central rotation and its orientation field estimation; (c) Altered fingerprint from (a) by 'Z' cut and its orientation field estimation.

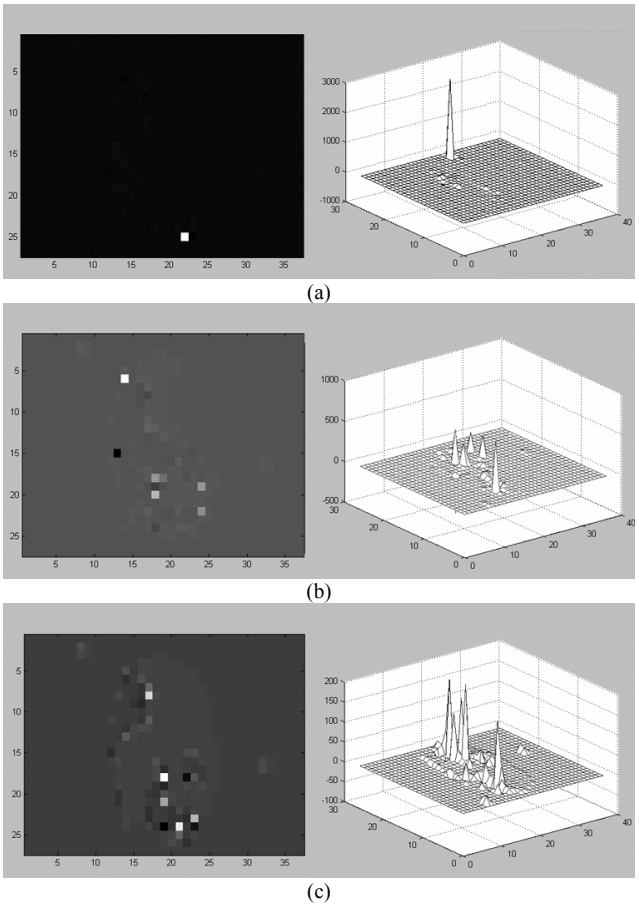


Fig. 3 Orientation field reliability map and peaks indicating the singular points for the arch type fingerprint in (a) natural fingerprint, (b) altered fingerprint by central rotation and (c) altered fingerprint by 'Z' cut.

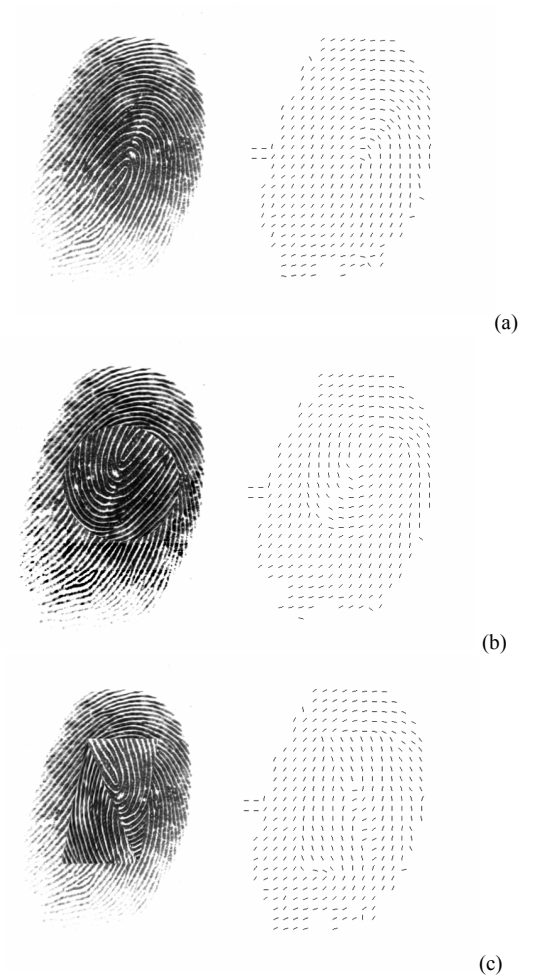


Fig. 4 Loop type natural fingerprint and its orientation field estimation; (b) Altered fingerprint from (a) by central rotation and its orientation field estimation; (c) Altered fingerprint from (a) by 'Z' cut and its orientation field estimation.

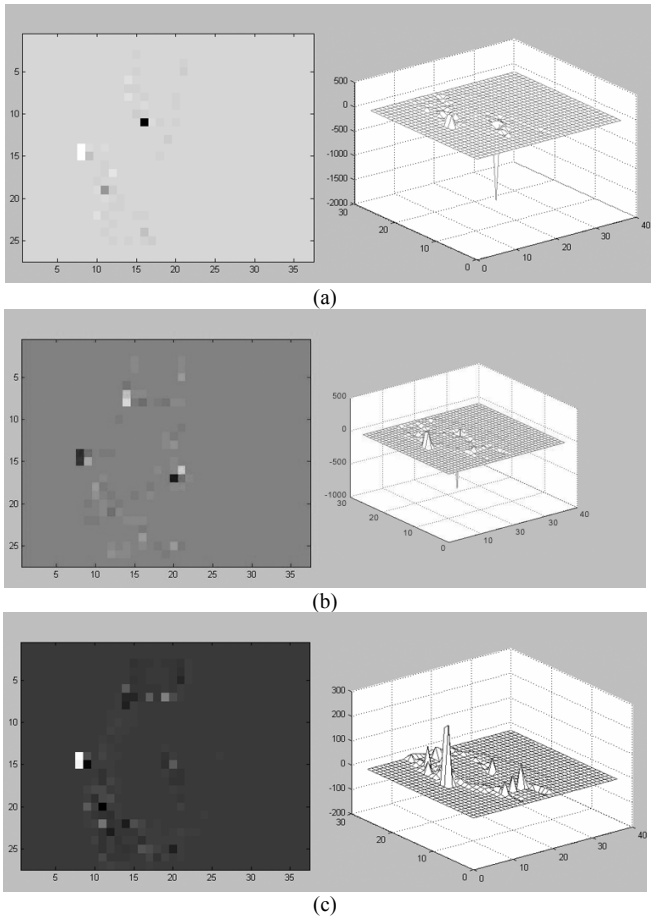


Fig. 5 Orientation field reliability map and peaks indicating the singular points for the loop type fingerprint in (a) natural fingerprint, (b) altered fingerprint by central rotation and (c) altered fingerprint by 'Z' cut.

natural fingerprint and its altered forms obtained by central rotation and 'Z' cut from Fig. 4. In the same figure the orientation fields of the above fingerprints are given.

For the loop type natural fingerprint represented in Fig. 4(a), a negative peak with a high value of –1558 is found, represented in Fig. 5(a). For the image from Fig. 4(b), the peaks found have also small values between 249 and -836, like the results obtained with the arch type fingerprint images. The peaks from the image in Fig. 4(c) vary between 247 and -105, represented in Fig. 5(c).

The next fingerprint of type double loop and the orientation field are represented in Fig. 6 together with the two types of alterations (i) and (ii). The double loop type natural fingerprint shown in Fig. 6(a) has one peak with high negative value of -3175, represented in Fig. 7(a). The values of the peaks from the two altered fingerprints varies from 590 for the distortions introduced in Fig. 7(b) and -647 for the negative peak and 407 for the positive peak in Fig. 7(c), respectively.

The orientation field reliability maps represented in Fig. 3, Fig. 5 and Fig. 7 represent the localization of the singular points in the two-dimensional space. The gray level is determined by R and is proportional with the strength of the peaks.

In [1], it was pointed out that using a feature vector termed as curvature histogram, extracted from the continuous vector field, as an input to train LIBSVM, an integrated software for support vector classification, leads to results showing that 92% altered fingerprints can be correctly detected.

In this work, singular points are detected with high amplitudes for natural fingerprints, and relatively small amplitudes for altered fingerprints, that can be used for classifying algorithms, in order to give a completely comparative study.

Testing the efficiency of the orientation field reliability, the experimental results indicate that the reliability R has strong information that can be used for future research. The singular value decomposition of R leads to obtaining essential features for discrimination and has good stability.

The results obtained in altered fingerprints analysis using orientation field reliability are persuasive and could be employed for automatic detection of biometric obfuscation.

The proposed algorithm is tested on synthetically altered images, in the absence of a real altered fingerprints database.

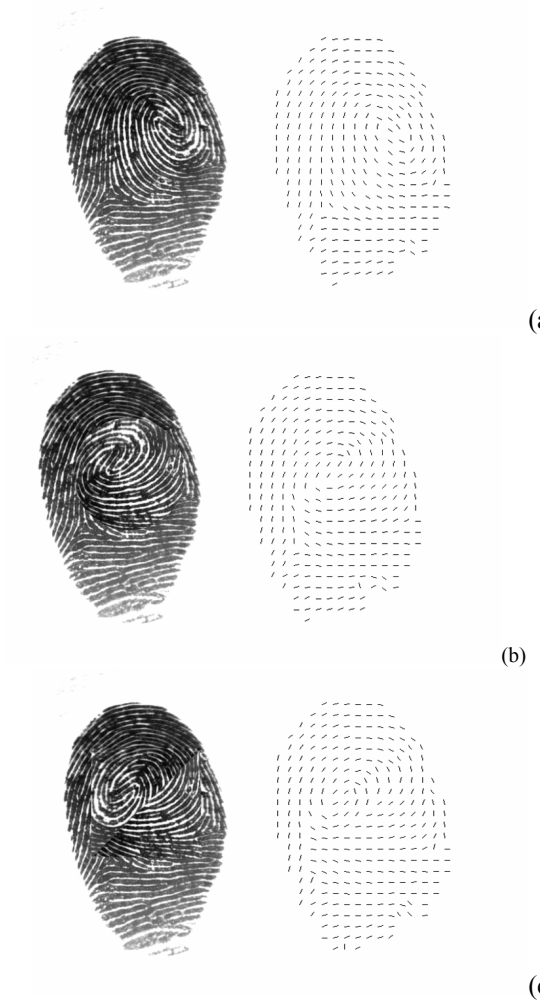


Fig. 6 Double loop type natural fingerprint and its orientation field estimation; (b) Altered fingerprint from (a) by central rotation and its orientation field estimation; (c) Altered fingerprint from (a) by 'Z' cut and its orientation field estimation.

V. Conclusion

This paper presented a method to analyze natural fingerprints and distorted fingerprints, based on the field orientation reliability. Using an orientation field, singular points are detected based on the fingerprint orientation field reliability. The orientation field reliability map has peaks in the singular point locations and these peaks are used to analyze altered fingerprints. Due to alteration, more peaks as singular points appear with lower amplitudes.

The experimental results demonstrate that the proposed algorithm can provide important information in order to automatically detect altered fingerprints.

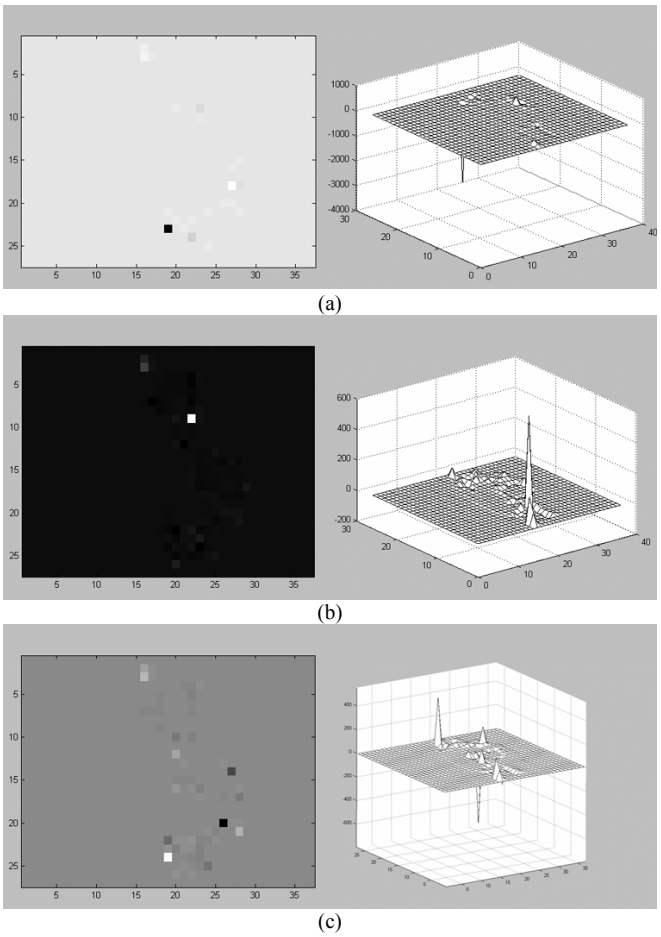


Fig. 7 Orientation field reliability map and peaks indicating the singular points for the double loop type fingerprint in (a) natural fingerprint, (b) altered fingerprint by central rotation and (c) altered fingerprint by 'Z' cut.

References

[1] Jianjiang Feng, Anil K. Jain, Arun Ross, "Fingerprint Alteration", *MSU Technical report*, MSU-CSE-09-30, Dec. 2009.

[2] A. Antonelli, R. Cappelli, D. Maio, and D. Maltoni, "Fake Finger Detection by Skin Distortion Analysis", *IEEE Trans. Information Forensics and Security*, vol. 1, no. 3, pp. 360-373, 2006.

[3] Ya. Imamverdiev, L. Kerimova, and Ya. Mussaev, "Method of Detection of Real Fingerprints on the Basis of the Radon Transform" *ISSN 0146-4116, Automatic Control and Computer Sciences, 2009,* Vol. 43, No. 5, pp. 270–275. © Allerton Press, Inc., 2009.

[4] Denis Baldisserra, Annalisa Franco, Dario Maio and Davide Maltoni, "Fake Fingerprint Detection by Odor Analysis", in *Proceedings International Conference on Biometric Authentication (ICBA06)*, Hong Kong, January 2006.

[5] L. Hong, Y. Wan, and A. K. Jain, "Fingerprint Image Enhancement: Algorithm and Performance Evaluation", *IEEE Trans. Patterrn Analysis and Machine Intelligence*, Vol. 20, no. 8, pp. 777-789, 1998.

[6] Mohammed Sayim Khalil, Dzulkifli Muhammad, Muhammad Khurram Khan, and Khaled Alghathbar, "Singular points detection using fingerprint orientation field reliability", *International Journal of Physical Sciences*, Vol. 5(4), pp. 352-357, April 2010.

Adding a Social Dimension to a Learning Style-based Adaptive Educational System

Elvira Popescu, *Member, IEEE*

***Abstract*—Providing social and adaptive learning environments is a desideratum of today's e-learning systems. In this paper, we make a first step towards this goal, by adding a social dimension to a learning style-based adaptive educational system (called WELSA). First, an overview of WELSA 1.0 is presented, focusing on its learner modeling method and adaptation provisioning techniques. Next, a mashup-based solution is presented for the integration of a set of Web 2.0 tools into the platform (blog, social software tool, microblogging tool). The necessary extensions for each system component are also introduced, leading towards WELSA 2.0.**

I. Introduction

IN the world of pervasive Internet, learners are also evolving: the so-called "*digital natives*" (persons born and raised after the development of digital technologies) [21], want to be in constant communication with their peers, they expect an individualized instruction and a personalized learning environment, which automatically adapts to their learning preferences. Therefore the need to offer intelligent e-learning platforms, where students can learn in a personalized way, by interacting and collaborating with their teachers and peers.

A first step towards attaining this goal was the creation of WELSA platform (**W**eb-based **E**ducational system with **L**earning **S**tyle **A**daptation) [19]. WELSA is aimed at providing courses adapted to each student's learning style, one of the individual characteristics that play an important role in the learning process, according to educational psychologists [16]. More specifically, learning style represents the individual manner in which a person approaches a learning task, the learning strategies activated in order to fulfill that task. For example, some learners prefer graphical representations and remember best what they see, others prefer audio materials and remember best what they hear, while others prefer text and remember best what they read. There are students who like to be presented first with the definitions followed by examples, while others prefer abstract concepts to be first illustrated by a concrete, practical example. Similarly, some students learn easier when confronted with hands-on experiences, while others prefer traditional lectures and need time to think things through. Some students prefer to work in groups, others learn better alone. These are just a few examples of the many different preferences related to perception modality, processing and organizing information, reasoning, social aspects etc., all of which can be included in the learning style concept [15].

In the last few years, Web 2.0 tools (also known as "social software tools", e.g., *blog, wiki*, *social bookmarking systems*, *media sharing tools*) gained a lot of attention and started to be used in educational settings [1, 10, 11], with encouraging results with respect to student satisfaction, knowledge gain and/or learning efficiency. This is motivated by the fact that the principles Web 2.0 is based on (user-centered, participative architecture, openness, interaction, social networks, collaboration) are in line with modern educational theories such as socio-constructivism [24]. According to it, knowledge cannot be transmitted but has to be constructed by the individual, by means of collaborative efforts of groups of learners [23].

Blogs for example can be seen as a means for students to publish their own ideas, essays and homework and as a space where they can reflect on their learning process (i.e., a kind of "learning diary"). Furthermore, posting comments to blog articles represents a means of social interaction, as well as an opportunity to provide critical and constructive feedback. Also, blogs help create a sense of community among students with similar interests ("educational blogosphere"). Similarly, social bookmarking tools can be used for storing and sharing links to resources of interest for the course (i.e., a kind of "personal knowledge management tool"). Students can share bookmarks they have discovered with their peers and also tag and rate the collected resources. More examples of pedagogical scenarios and practical guidelines for the use of Web 2.0 technologies to support teaching and learning can be found in [10].

In this context, it is only natural to try to merge the advantages of personalization already provided by WELSA with those promised by Web 2.0 tools. We therefore decided to add a social dimension to WELSA, by integrating a set of Web 2.0 tools with demonstrated pedagogical value into the platform. From a technical point of view, the solution that we propose is based on mashups [12], ensuring a lightweight architecture, with loosely-coupled components.

This work was supported by the strategic grant POSDRU/89/1.5/S/61968, Project ID 61968 (2009), co-financed by the European Social Fund within the Sectorial Operational Program Human Resources Development 2007 – 2013.

E. Popescu is with the Software Engineering Department, University of Craiova, Romania (e-mail: popescu_elvira@software.ucv.ro).

The rest of the paper is structured as follows: in section 2 we present an overview of the initial version of WELSA (WELSA 1.0); the extensions that we propose are described in section 3 (WELSA 2.0); finally, in section 4 we draw some conclusions and point towards future research directions.

II. WELSA 1.0

WELSA is an intelligent learning environment, which dynamically adapts the courses to the learning style of each student. The process implies two stages:

1. identifying the learning style of the student (modeling stage)
2. applying the corresponding adaptation rules (adaptation stage).

Please note that a detailed presentation of WELSA 1.0 can be found in [19]. In what follows, we give a brief overview of the system, outlining its main functionalities and underlying principles.

A. Learner Modeling

An important step towards an accurate learner diagnosis is the selection of an appropriate taxonomy of learning styles. Based on an extensive examination of the learning style models proposed in the literature, we created our own model, called ULSM (Unified Learning Style Model), which integrates the most representative characteristics from traditional models. Thus, ULSM includes preferences related to: perceptual modality, way of processing and organizing information, motivational and social aspects (e.g., *Visual / Verbal, Abstract / Concrete, Serial / Holistic, Active experimentation / Reflective observation, Individual work / Team work, Intrinsic motivation / Extrinsic motivation*). A detailed description of the ULSM model, as well as its advantages over traditional models in e-learning settings is included in [16].

Based on our own experimental studies [14], as well as other similar reports [9], we consider students' behavior in the system (i.e., time spent on each type of learning resources, order of accessing the resources, level of involvement with the communication tools) to be an accurate indicator of their learning preferences. We therefore decided to use an implicit modeling method, based on the automatic monitoring and analysis of students' actions in the system (as opposed to an explicit modeling method, based on students' answers to dedicated psychological questionnaires).

The modeling process unfolds as follows: first, the logged student actions are preprocessed and aggregated to yield the behavioral patterns. Next, the reliability levels of these patterns are calculated as well (i.e., the larger the number of available relevant actions, the more reliable the resulted pattern). Subsequently, the WELSA Analysis tool computes the ULSM preference values, using modeling rules based on the pattern values, their reliability levels and their weights. It should be noted that these rules also take into account the specificities of each course: the pattern thresholds as well as the importance of each pattern may vary with the structure and subject of the course. Therefore, the teachers should have the possibility to adjust the predefined values to correspond to the particularities of her/his course or even to eliminate some of the patterns, which are not relevant for that course. This is why the WELSA Analysis tool has a configuration option, which allows the teacher to modify the weight and threshold values.

B. Adaptation Provisioning

Once the students' learning preferences are identified by the WELSA Analysis tool, the next step is to associate adaptation actions that are best suited for each preference. More specifically, we decided to use adaptive sorting and adaptive annotation techniques. Thus, the learning objects (LOs) are placed in the course page in the order which is most appropriate for each learner; additionally, a "traffic light" technique was used to differentiate between recommended LOs (with a highlighted green title), standard LOs (with a black title) and not recommended LOs (with a dimmed light grey title). It should be mentioned however that the learning path suggested by the system is not compulsory: it is simply a recommendation that the student may choose to follow or not. We consider that offering control to students, instead of strictly guiding them, is a more flexible and rewarding pedagogic approach [17].

From a technical point of view, it should be mentioned that WELSA does not store the individualized course web pages, but instead generates them on the fly, each time an HTTP request is received by the server. The adaptation servlet queries the learner model database, in order to find the ULSM preferences of the current student. Based on these preferences, the servlet applies the corresponding adaptation rules and generates the new HTML page, by automatically composing it from the selected and ordered LOs, each with its own status (highlighted, dimmed or standard).

Figure 1 gives an overall view of WELSA system, illustrating the interactions with the two main actors (the student and the teacher), as well as the process workflow. As can be seen in the figure, WELSA is composed of three main modules:

- an authoring tool for the teachers, allowing them to create courses conforming to the internal WELSA format (XML-based representation)
- a data analysis tool, which is responsible for: i) interpreting the behavior of the students and consequently building and updating the learner model, based on the

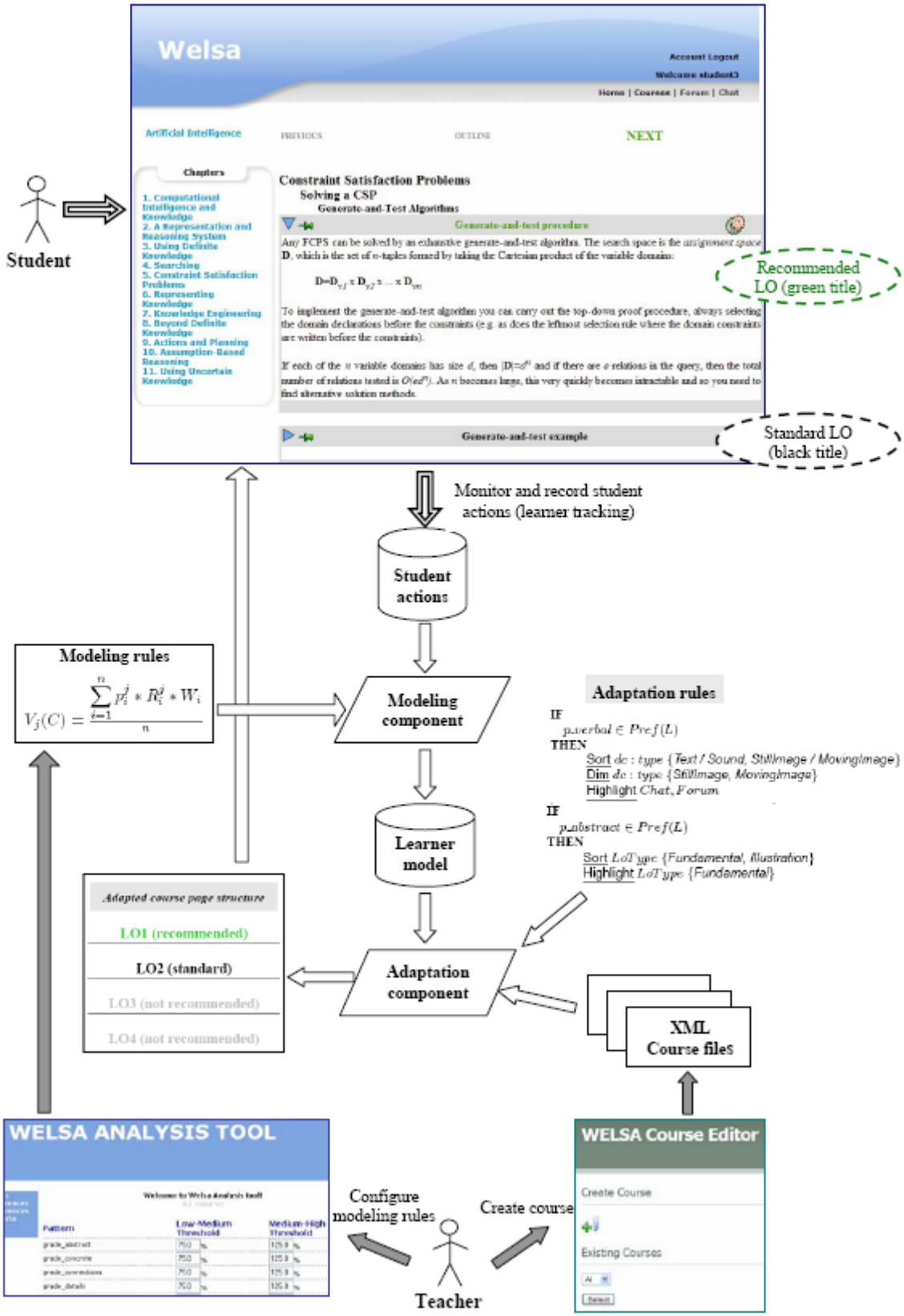


Figure 1. WELSA 1.0 overall architecture

built-in modeling rules; ii) providing configuration options for the teachers, who can set certain parameters of the modeling process, so that it fits the particularities of their own course; iii) providing various aggregated information about the learners

- a course player (basic learning management system) for the students, enhanced with two special capabilities: i) learner tracking functionality (monitoring the student interaction with the system); ii) adaptation functionality (incorporating adaptation logic and offering individualized course pages).

As far as the implementation is concerned, Java-based and XML technologies are employed for all WELSA components. Apache Tomcat 6.0 is used as HTTP web server and servlet container and MySQL 5.0 is used as DBMS.

C. System Validation

According to the layered evaluation approach proposed in [4], WELSA was validated experimentally both from the learner modeling and the adaptation provisioning point of view. The results of the former experiment are presented in [15], while the results of the latter are reported in [17].

Furthermore, we performed also a global evaluation of the WELSA system. After following the course sessions, the 71 students participating in the experiment were asked to assess various aspects of their learning experience with WELSA, on a 1 to 10 scale (e.g., course content, presentation, platform interface, navigation options, communication and collaboration tools, the course as a whole). Very good marks were assigned to most of the features, with only one feature (the communication tools) receiving lower ratings; this can be explained by the fact that only basic communication functionalities were offered to the students (i.e., forum and chat). All in all, we can conclude that students had a very positive learning experience with WELSA. These findings are also reflected in the readiness of the students to adopt WELSA system for large scale use, with 87.50% of them willing to do so and only 6.25% reluctant [19].

Based on the results of the experiment (i.e., students' suggestions regarding the use of more advanced communication and collaboration tools), as well as on the increasing importance of Web 2.0 applications in education (as pointed out in section 1), we decided to add a social dimension to our system, leading towards *WELSA 2.0*. More specifically, we decided to extend the communication and collaboration functionalities (which only consist in a forum and a chat in WELSA 1.0) with the new generation of Web 2.0 tools (blog, social bookmarking tool, microblogging tool etc.). The next section gives an overview of the platform extensions performed to this end.

III. TOWARDS WELSA 2.0

A. Architectural Issues

The integration of the Web 2.0 components can be done by means of *mashups* (i.e., combining data and/or functionalities from two or more external sources to create a new Web application). Accessing data and functionalities can be done by several methods:

- APIs (Application Programming Interface) based on REST (Representational State Transfer)
- RSS (Really Simple Syndication) feed integration
- Screen scraping [12].

Paper [2] contains a review of mashup applications in various domains; examples of e-learning applications include: [8], [25].

The first step towards the creation of WELSA 2.0 was to select the most suitable Web 2.0 tools to be integrated in the system, which meet two requirements:

- have a demonstrated pedagogical value (according to case studies reported in the literature)
- offer technical support for integration (well documented and maintained APIs, RSS feeds etc.).

We therefore decided to add the following tools in WELSA: blog (Blogger [3]), social bookmarking tool (Delicious [7]), and microblogging tool (Twitter [22]). Naturally, the range of social media instruments could be subsequently extended.

Thus, students will be able to use these tools (that most of them are already familiar with in informal contexts outside school) in a semi-formal framework, in the WELSA e-learning platform. The tools are added as widgets in the platform interface (i.e., *aggregation mashups*); all student actions (such as: *post_blog_message*, *add_bookmark*, *post_tweet* etc.) can be retrieved from the tools (by means of APIs or RSS) and recorded in the platform's database, for further processing (i.e., *integration mashups*).

Figure 2 shows a schematic architecture of WELSA 2.0, highlighting the newly added or extended components (grey shaded areas):

- an *additional learner tracking component*, which uses the APIs or RSS feeds provided by Blogger, Delicious and Twitter respectively, to retrieve learner actions and store them in the WELSA database
- the *Student actions database*, covering a wider range of student actions (e.g., *login_blog, login_twitter, login_delicious, post_blog_message, post_blog_comment, post_tweet, add_bookmark* etc.)
- the extended *Modeling and Adaptation rules*, accommodating the social and collaborative preferences of the students.

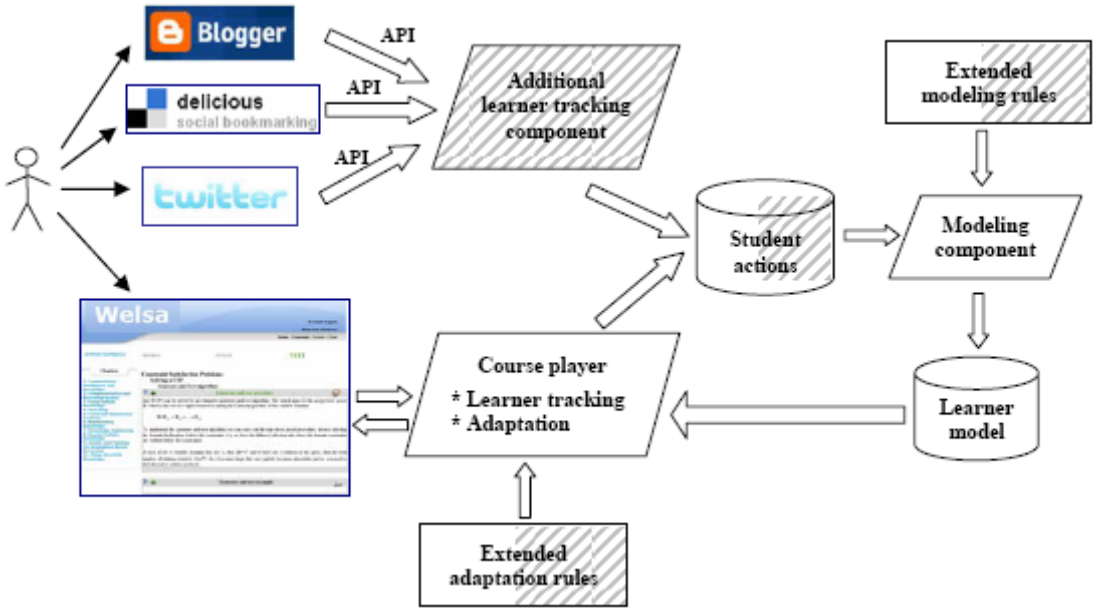


Figure 2. WELSA 2.0 schematic architecture (grey shaded areas represent extensions from WELSA 1.0)

B. *Extending the Modeling Component*

As already noted, with the introduction of the three new tools, the range of student actions monitored and recorded by the system is extended, which leads to an increase in the number (and variety) of behavioral patterns that can be computed. WELSA 1.0 already takes into account over 100 behavioral indicators, referring to:

- Educational resources (i.e., learning objects - LOs) that compose the course: time spent on each LO, number of accesses to an LO, number of skipped LOs, results obtained to evaluation tests, order of visiting the LOs. For each LO we have access to its metadata file, including information regarding the instructional role (e.g., 'Definition', 'Example', 'Exercise', 'Interactivity', 'Illustration' etc.), the media type (e.g., 'Text', 'Sound', 'Image', 'Video'), the level of abstractness and formality etc.
- Navigation choices: either linear, by means of the "Next" and "Previous" buttons or nonlinear, by means of the course Outline.
- Communication tools - a synchronous one (chat) and an asynchronous one (forum): time, number of visits, number of messages.

Once we introduce the Web 2.0 tools, the range of behavioral patterns is extended by adding social and collaboration indicators: number of blog posts, number of blog comments, number of tweets, number of bookmarks added, time spent on the blog, frequency of accesses to the social bookmarking tool etc. Consequently, the modeling rules have to be refined to take into consideration the new behavioral indicators. A first step in this respect would be to perform an experimental study with the students, monitoring and recording all their interactions with WELSA 2.0. Next, statistical analysis should be applied on the data to identify correlations between the newly introduced behavioral indicators and students' ULSM preferences, following the approach proposed in [14]; these correlations will represent the basis for the new modeling rules. Based on our previous findings [15], we expect that a higher number of available patterns will also lead to a higher accuracy of the learning style diagnosis.

Just as in case of WELSA 1.0, teachers will have the possibility to adjust the new modeling rules to conform to the specificities of the course at hand (e.g., a course whose assignments rely heavily on the use of blog will place a high weight on the *number_of_blog_posts* behavioral indicator; a course not requiring the use of microblogging will eliminate all Twitter-related behavioral indicators from the modeling rules etc.).

Furthermore, since the social and collaborative behavioral indicators are stored in WELSA database together with the rest of the students' behavioral patterns, the data analysis tool will be able to automatically include them in the aggregated data that it offers to the teachers for visualization (e.g., number of each action type per student, time spent using each tool, length of blog contributions etc.). This information can be used by the teachers for comparisons, grading, statistical purposes. Obviously, only quantitative data can be provided by the system; as far as qualitative analysis is concerned, this has to be performed by the teachers (by manually inspecting the actual content

contributed by the students in the Web 2.0 tools). Parts of this task could also be automated (e.g., natural language processing, content analysis for Web 2.0 [5]) but this is outside the scope of our system.

C. Extending the Adaptation Component

As mentioned in the previous section, WELSA 1.0 supports only navigation and presentation level adaptation. In WELSA 2.0, the range of adaptation actions could be extended to incorporate also collaboration level adaptation. Two extension directions are envisaged:

- First of all, recommendations can be given for group formation based on students' learning styles. Similar works in this area include: [6], [13].
- Secondly, recommendations can be offered regarding the most suitable Web 2.0 tool for each student and each task. A first step in this respect is reported in [20], which identified the effects of cognitive style on user acceptance of blogs and podcasts.

It is important to mention that expanding the range of adaptation actions will not imply an increase in the teacher's workload; the adaptation is done automatically by the system, based on the built-in rules. Of course, just as in case of WELSA 1.0, teachers will have the possibility to fine-tune the newly added adaptation rules (e.g., define specific constraints for group formation).

IV. Conclusions

We started this paper with an overview of WELSA 1.0 educational system, focusing on its learner modeling method and adaptation provisioning techniques. We then showed how a social dimension can be added to WELSA 1.0, by integrating a set of Web 2.0 tools (blog, social bookmarking tool, microblogging tool).

As future work, we plan to complete the prototype implementation of WELSA 2.0 and perform experiments with students to validate the new platform (i.e., assess the accuracy of the modeling method as well as the efficiency and effectiveness of the adaptation actions on the learning process). The encouraging results obtained with WELSA 1.0 (as reported in [17]), corroborated with our students' positive attitude towards the use of Web 2.0 tools in education (as reported in [18]), lead us to believe that the new WELSA 2.0 system will prove beneficial to the learning process.

References

[1] P. Anderson, *What is Web 2.0? Ideas, technologies and implications for education*, technical report, JISC, 2007, available at: www.jisc.ac.uk/media/documents/techwatch/tsw0701b.pdf

[2] B. Beemer and D. Gregg, "Mashups: A Literature Review and Classification Framework", *Future Internet*, 1(1), 2009, pp. 59-87.

[3] Blogger – http://www.blogger.com, last accessed on May 10, 2010.

[4] P. Brusilovsky, C. Karagiannidis, and D. Sampson, "Layered Evaluation of Adaptive Learning Systems", *International Journal for Continuing Engineering Education and Life-Long Learning*, 14(4/5), 2004, pp. 402–421.

[5] CAW 2.0 - Content Analysis for the Web 2.0 - http://caw2.barcelonamedia.org/, last accessed on May 10, 2010.

[6] C. Christodoulopoulos and K. Papanikolaou, "A Group Formation Tool in an E-Learning Context," *Proc. ICTAI 2007*, pp. 117-123.

[7] Delicious - http://delicious.com/, last accessed on May 10, 2010.

[8] H. Drachsler, D. Pecceu, T. Arts, E. Hutten, L. Rutledge, P. Van Rosmalen et al., "ReMashed-Recommendations for Mash-Up Personal Learning Environments", *Proc. EC-TEL 2009*, LNCS 5794, pp. 788-793.

[9] S. Graf, *Adaptivity in Learning Management Systems Focussing on Learning Styles*, PhD Thesis, Vienna University of Technology, Austria, 2007.

[10] K. Grodecka, F. Wild, and B. Kieslinger, *How to use social software in higher education*, iCamp handbook, 2009, available at: http://www.icamp.eu/wp-content/uploads/2009/01/icamp-handbook-web.pdf

[11] M. Homola and Z. Kubincova, "Taking Advantage of Web 2.0 in Organized Education (A Survey)", *Proc. ICL 2009*, pp. 741-752.

[12] E. Ort, S. Brydon, and M. Basler, "Mashup Styles", 2007, http://java.sun.com/developer/technicalArticles/J2EE/mashup_1

[13] P. Paredes, A. Ortigosa, and P. Rodriguez, "TOGETHER: an Authoring Tool for Group Formation based on Learning Styles", *Proc. A3H (7th International Workshop on Authoring of Adaptive and Adaptable Hypermedia at EC-TEL 2009)*.

[14] E. Popescu, "Learning Styles and Behavioral Differences in Web-based Learning Settings", *Proc. ICALT 2009*, IEEE Computer Society Press, pp. 446-450.

[15] E. Popescu, "Diagnosing Students' Learning Style in an Educational Hypermedia System", *Cognitive and Emotional Processes in Web-based Education: Integrating Human Factors and Personalization*, Advances in Web-Based Learning Book Series, IGI Global, 2009, pp. 187-208.

[16] E. Popescu, "A Unified Learning Style Model for Technology-Enhanced Learning: What, Why and How?", *International Journal of Distance Education Technologies (IJDET),* 8(3), IGI Global, 2010, pp. 65-81.

[17] E. Popescu, "Adaptation Provisioning with respect to Learning Styles in a Web-Based Educational System: An Experimental Study", *Journal of Computer Assisted Learning*, 26(4), Wiley, 2010, pp. 243-257.

[18] E. Popescu, "Students' Acceptance of Web 2.0 Technologies in Higher Education: Findings from a Survey in a Romanian University", *Proc. DEXA 2010 Workshops*, IEEE Computer Society Press, pp. 92-96.

[19] E. Popescu, C. Badica, and L. Moraret, "Accommodating Learning Styles in an Adaptive Educational System", *Informatica*, 34(3), 2010 (in press).

[20] N. Saeed, Y. Yang, and S. Sinnappan, "Effects of Cognitive Style on User Acceptance of Blogs and Podcasts", *Proc. ICALT 2009*, IEEE Computer Society Press, pp. 293-297.

[21] G. Small and G. Vorgan, *iBrain: Surviving the Technological Alteration of the Mondern Mind*, HarperColins, 2008.

[22] Twitter - http://twitter.com/, last accessed on May 10, 2010.

[23] C. Ullrich, K. Borau, H. Luo, X. Tan, L. Shen, and R. Shen, "Why Web 2.0 is good for learning and for research: principles and prototypes", *Proc. WWW 2008*, ACM Press, pp. 705-714.

[24] L. S. Vygotsky, *Mind in society*, Harvard University Press, Cambridge, MA, 1978.

[25] F. Wild, *Mash-Up Personal Learning Environments*, iCamp project deliverables, 2009 - available at: http://www.icamp.eu/wp-content/uploads/2009/01/d34_icamp_final.pdf.

XCPI - A Measurement and Calibration Software Tool for Networked and Embedded Control Systems

Mihai Postolache, Ciprian Spiridon

***Abstract*— This work is presenting an implementation of a software tool for calibration of and data acquisition from standalone and networked embedded control systems. The proposed low-cost and still powerful tool XCP Interpreter (XCPI) is aimed to assist the developer of embedded systems from the early stage of coding until the later ones of testing, validating and parameterization of the final product.**

I. Introduction

THE development of nowadays real-time networked embedded systems, consisting of several Electronic Control Units (ECUs) interconnected through different standard or proprietary industrial networks is a real challenge. Their complex interactions between nodes, while running in parallel make difficult for developer to write, debug, test and validate applications for such control systems. Higher level tools and communication protocols are required to assist the system designer at every stage of development.

For example, the automotive industry has developed professional and expensive tools (as shown in Fig. 1) for calibration and measurement of Controller Area Networks - CAN (CANape - Vector Informatik, XCP modules for LabView - National Instruments). The underlying communication protocol used is Universal Measurement and Calibration Protocol (XCP) [5], which is an extension of CAN Calibration Protocol (CCP) [4] provided by ASAM (Association for Standardisation of Automation and Measuring Systems), earlier known as ASAP (Arbeitskreis zur Standardisierung von Applikations-systemen), a consortium of European manufacturers of automation, test and development systems for the automotive industry as well as manufacturers of electronic control units.

However, CCP and XCP protocols are intended for but not limited to automotive systems design. Due to their specific facilities and simple architecture, these application-level protocols can be used in any non-automotive application relaying on networked and embedded systems.

The next sections of this paper are to describe a simple yet powerful tool which will be referred in the following as the XCP Interpreter (XCPI). This software tool is based on the public version of basic XCP provided by Vector Informatik that may be easily customized for any particular ECU.

Manuscript received May 15, 2010. This work was supported by The National Centre for Program Management from Romania under the research grant SICONA – 12100/2008.

Authors are with The "Gheorghe Asachi" Technical University of Iasi, Prof.dr.doc. Dimitrie Mangeron Street, No. 27, 700050 IASI, phone: +40-232-278680/1322; fax: +40-232-230751; e-mail: mpostol@ ac.tuiasi.ro).

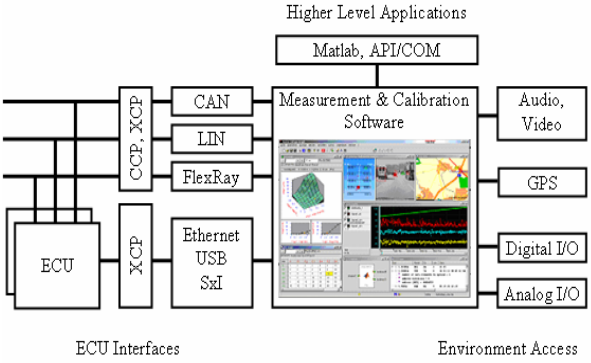


Fig. 1. Framework of calibration and measurement system used in the automotive industry

Related research work on the subject was done by Ao *et all* in [1] where a user-friendly calibration system based on CCP is described, with the function of measurement data acquisition and parameter calibration of an hybrid electric vehicle. Also, in [2] Yang S.W. *et all* express a multi-node calibration system for multi-ECU based on CCP protocol, and Yang He *et all* developed an XCP based distributed calibration system [3].

II. Calibration and Measurements in Networked and Embedded Systems

Usually, embedded systems are manufactured with the calibration process in mind. For example, the result of the software development process of ECU consists of algorithms embedded in firmware whose behavior might be attuned using different parameters later on. Those parameters are given by the manufacturer default values at the end of the ECU development process. It is obvious that measurements are made since the early hardware and software development stages in order for the ECU to meet specific design requirements.

What the activity of calibration and measurement means starts usually after the fabrication process of ECU has been completed. Here is where the application engineer or the integrator is getting involved with the fine-tuning of the parameters of an ECU introduced in a more complex system in order to meet overall system requirements. So, the application engineer receives specifications form standards and regulations and tries to obtain an optimal

behavior of the system as whole by adjusting specific parameters of the component ECU, which is the calibration process.

But calibration is a closed loop optimization process so it needs feedback, as in Fig. 2. Changing parameters must be accompanied by measuring the effects produced on the controlled system. This means not only acquiring data from sensors and actuators using conventional meters, but also reading the values of other internal variables of ECU which is not possible unless it is provided with particular measurement interfaces.

The interfaces are meant to allow the application engineer to adjust parameter values and read back internal variables stored in local memory of any ECU in real time. All this must be done without having the ECU physically disconnected from the network or even with no significant

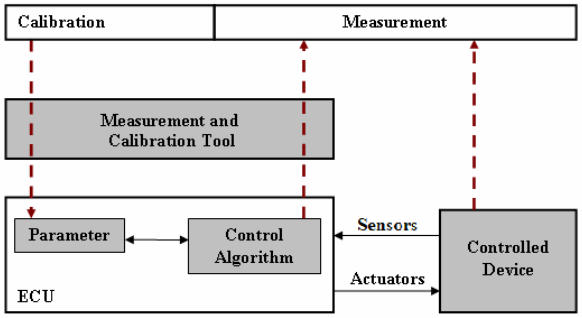


Fig. 2. Calibration of ECU parameters and measurement of physical and virtual quantities in a closed loop optimization process

interference on the normal operation of it.

Such requirements for the calibration and measurements interface for the ECU components of a networked embedded system cannot be reached unless by using the same communication protocols at the lower levels of the network stack. CAN Calibration Protocol can be used to build such interfaces, but for CAN-based networks only.

On the other hand, a universal calibration and measurement interface is desirable, which is intended to be independent of the lower levels of the network stack. XCP is an extension of CCP in this direction, having a common higher layer over CAN, LIN, FlexRay, SxI, USB, Ethernet or other lower level protocols used in embedded networks.

The calibration process is considered finished when optimal values of the parameters have been established and therefore written in a non-volatile memory area of the ECU.

The type and the number of calibration parameters are application specific. When complex functions are accomplished, an ECU may use even thousands of parameters or more and the internal variables to be monitored are of the same magnitude. This is at least one reason to consider the calibration as a very complex procedure, a complexity that the tools used in this process have to deal with.

In Fig. 3 is illustrated the architecture of a calibration

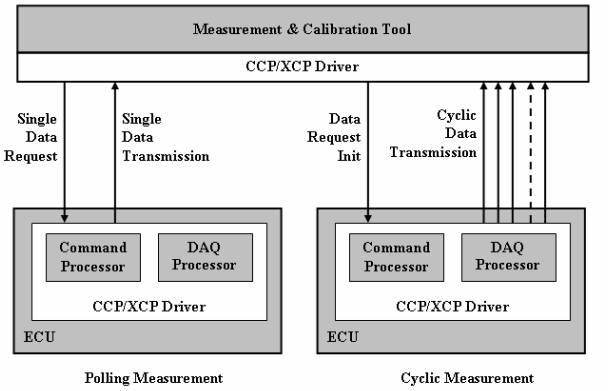


Fig. 3. Architecture of a calibration and measurement tool for networked and embedded systems

and measurement tool for networked and embedded systems. It is a master slave configuration, with the master making asynchronous polling or cyclic measurements of the parameters or internal variables of the slave ECU. Each slave ECU runs a CCP/XCP driver that receives the requests from the master and executes them by replying with the requested one-shot or cyclic data. The next section will have an insight of the family of protocols CCP and XCP which the software tool described subsequently is based on.

III. The CCP and XCP Family of Calibration and Measurement Protocols

A. Overview of the CAN Calibration Protocol (CCP)

CAN is a joint development of Robert Bosch GmbH and Intel Corporation used in many high-end automotive control systems as well as in industrial control systems. The CAN Communication Protocol used for calibration and data acquisition is a master-slave type communication as

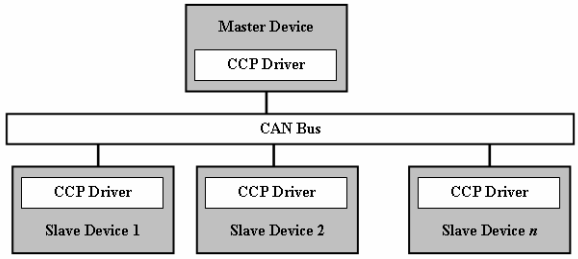


Fig. 4. The master – slave architecture of CCP on top of CAN

the 7th -layer as defined by the OSI Reference Model on top of CAN, as shown in Fig. 4.

The master device (host) is a calibration tool or a diagnostic / monitoring tool or a measurement system initiating the data transfer on the CAN by sending

commands to the slave devices. The CCP implementation supports commands for generic control with primitive memory transfers and for data acquisition. These two function sets of the communication protocol are independent and may run asynchronously, depending on their implementation in the slave controller.

CCP has been designed to handle the restrictions and demands of both small 8-bits microcontrollers and embedded systems with high performance. No extra hardware has to be connected to the ECU as the CCP driver is fully implemented in the software. A simple implementation of CCP only needs a small part of the available RAM, ROM and CPU time for execution.

A simple implementation of CCP only need two CAN message identifiers, which can be set as low priority that does not disturb the application required network traffic. If CCP is to be used from an ordinary PC as master, the same

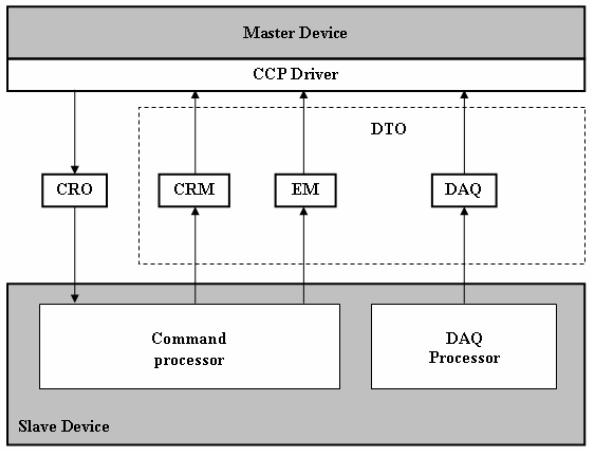


Fig. 5. CCP specific CRO and DTO messages

simple and low cost CAN interface which is used in microcontrollers can be used.

CCP uses generic commands, which are not node specific, to perform different functions in a slave node. A logical connection between master and slave has to be set up using individual node addresses before any commands can be sent. Once a connection established, every message from the master is followed by a reply message from the slave containing data or error codes. A connection persists until the master decides to connect to another slave or until the master sends a disconnect command.

The two types of CAN messages needed for communication between master and the selected slave are Command Receive Object (CRO) and Data Transmission Object (DTO), one for each direction (Fig. 5).

CRO messages are sent from master to slave and contain control commands. DTO messages are sent from slave to master. When a slave has received a CRO message it performs the given instructions and then answers with a DTO message containing a Command Return Message (CRM). The CRM code tells the master if the corresponding control command has been performed as required or not.

The CAN identifiers used for the CRO and DTO messages are defined by the slave device and used by a configuration file (A2L file, defined by the ASAM MCD 2MC/ASAP2 standard) in order to configure the master. The configuration file may also contain information about

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
CRO	CMD	CTR	Data	Data	Data	Data	Data	Data
DTO	CMD	ERR	CTR	Data	Data	Data	Data	Data

Fig. 6. The general format of CRO and DTO messages

the slave memory organization, which is useful for data acquisition and calibration.

CRO messages (Fig. 6) are sent from master to slave and contain instructions. The first byte is a command code (CMD) which describes the purpose of the message. The second byte is a command counter (CTR) and is used for keeping track of the communication. The command counter is also expected to be sent in return in the DTO message from the slave. Bytes 2-7 are reserved for data parameters depending on the command code. A message is always 8 bytes long even when not all are used.

The DTO message (Fig. 6) is sent by the slave as reply to a received CRO message and it is also used for data acquisition. The first byte in the message is the Packet ID (PID). The value of the PID describes the message type. It can be a CRM-DTO (reply message to a received CRO), an Event Message (EM-DTO) if the DTO reports internal slave status changes or it can be a Data Acquisition

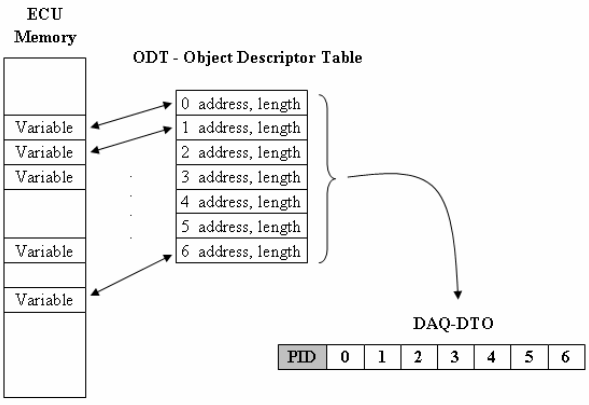


Fig. 7. ODT list organization for data acquisition

Message (DAQ-DTO), used for cyclic data acquisition from the slave device.

The master device can initialize data acquisition from a slave device, which in return sends the defined data in DAQ-DTOs. Up to 7 internal variables of ECU are

assigned to a list called Object descriptor Table, as in Fig 7.

Each entry in the ODT list contains at least the address of a variable in the memory of ECU and its length in bytes. Each ODT list has an ID (0-253) and its value stands for PID in a DAQ-DTO message. Thus, using a DAQ-DTO message a slave device can send to the master 7 bytes of internal data.

Typically several ODTs are defined for data acquisition in a slave ECU (Fig. 8). CCP allows the setup of a number of DAQ lists consisting of multiple ODTs, which may be simultaneously active. The sampling and transmission of

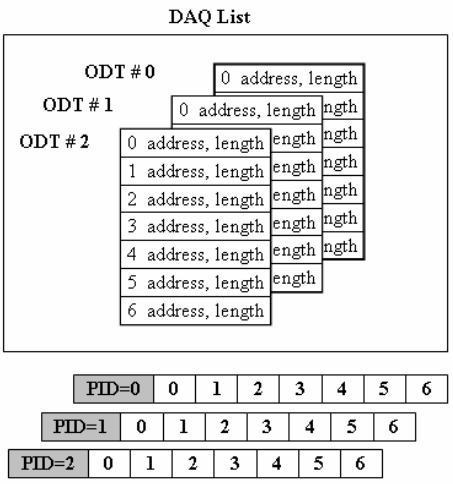


Fig. 8. DAQ list with 3 ODT lists and the 3 DAQ-DTOs

the DTOs of each DAQ list is triggered by individual events in the ECU.

Specific commands may be sent by the master using CRO messages in order to set up ODT lists and DAQ lists and initialize the transmission of DAQ-DTO messages.

B. New Features of XCP Protocol

The XCP protocol is an improved and generalized version of CCP v.2.1, which can be used not only in CAN-based networks but in any other automotive or industrial control networks. The XCP Protocol Layer is independent of several XCP Transport Layers on top of which is built such as XCP on CAN, XCP on Ethernet, XCP on SxI etc.

The XCP protocol extends the main features of CCP described above with new functions such as synchronous stimulation, a cyclic procedure similar to synchronous data acquisition with the mention that the data transfer are initiated by the master and have as destination the selected slave. Also, the overall performance is improved by power-up data transfer, self-configuration of ECUs and measurement modules, time stamping, higher efficiency and throughput. Even with all these enhancements and extensions, the ECU memory footprint and resource consumption remain small, similar to those used by CCP. Typical XCP on CAN implementation is 2 to 3 Kbytes of

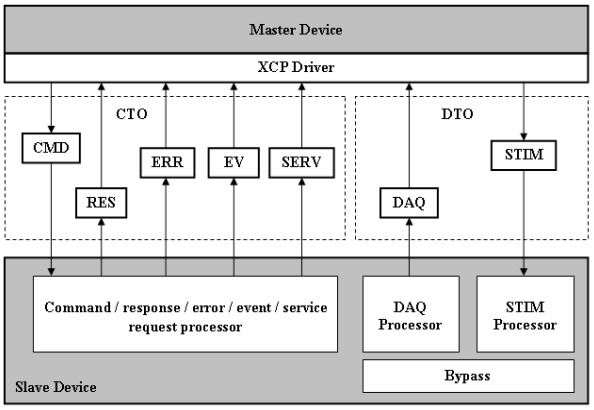


Fig. 9. XCP communication flow between master and slave devices

ROM and at least 100 bytes of RAM, depending on the implemented options and the amount of data acquired.

In Fig. 9 the communication flow between the master device and the selected slave device is shown. One can observe that Command Transfer Object (CTO) messages replace all the CRO together with CRM-DTO and EV-DTO messages defined as of CCP. DAQ messages as of CCP together with the newer STIM messages are considered Data Transfer Object (DTO) messages as of the XCP protocol.

Just like with CCP, only two CAN identifiers will be used with XCP on CAN, one for each of the two types of XCP messages, CTO and DTO respectively.

IV. THE XCP INTERPRETER

The application program was developed using Microsoft C# and requires .NET Framework 2.0 being installed on the PC station running the Microsoft Windows XP operating system. A CANcaseXL interface via USB from Vektor Informatik was used to physically connect the master device to the CAN bus as well as the XL-Driver–Library API for .NET for application programming. There are three layers on which the application is built of. First, a friendly user interface layer provide the user with menus and control boxes appropriate for configuration, calibration and measurements of various parameters of ECUs.

The second layer of the application is implementing the XCP driver for the master device running the XCP Interpreter. This layer handles any CTO and DTO messages required for calibration and data acquisition according to its functionality as of XCP basic version of implementation. Here are the commands implemented for connecting to or disconnecting from a slave device as well as read and write the selected slave ECU memory.

The lower layer of the application is the CAN Transport Layer as defined by the ASAM XCP Part3 – Transport Layer Specification XCP on CAN v.1.0.0 and make use of the Vector Informatik CAN API library which provide the

upper levels two CAN channels: one for transmission of CTO messages and the other for receiving DTO messages.

The application program is able to load initial configuration files (ASCII files) containing information about predefined calibration and data acquisition session or

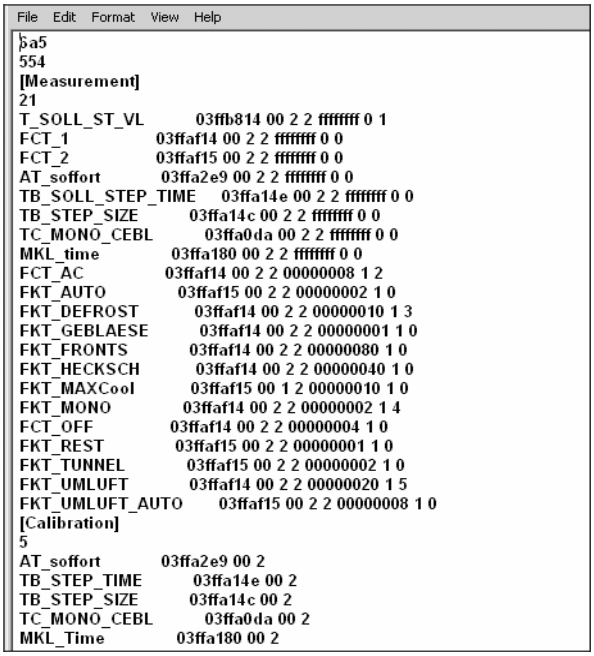


Fig. 10. XCP session ASCII configuration file

previously saved sessions. The identifiers used for the two CAN communication channels are part of this configuration file as well as a detailed description of the calibration parameters and measurement internal variables of ECU. An example of configuration file is shown in Fig.

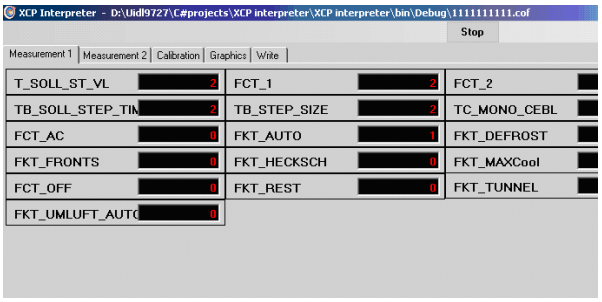


Fig. 11. Part of measurement application window

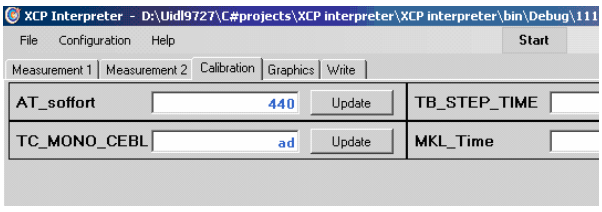


Fig. 12. Part of calibration application window

10.

The configuration file also indicates the number of parameters monitored in the measurement window, in the calibration window and in the graphical view window as

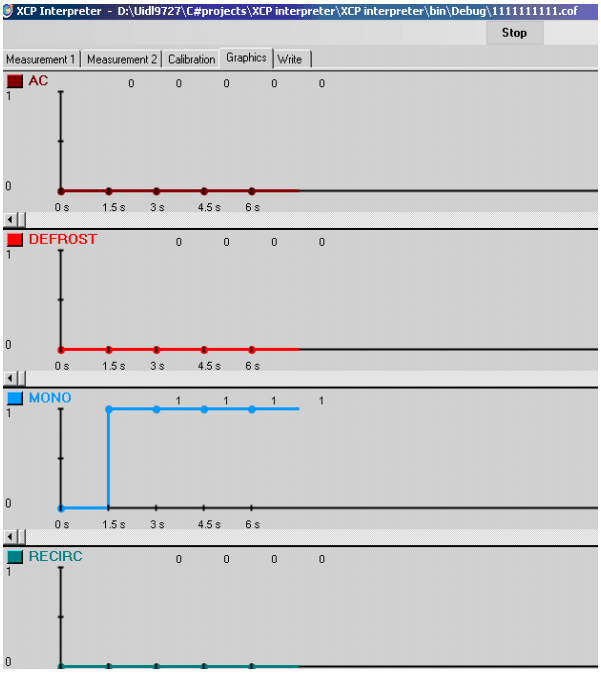


Fig. 13. Sample of graphical view application window

well as other identification data for each parameter (name, address, address extension, bit mask, data type, value, upper and lower bounds etc.).

At most 160 internal values of ECU can be measured and displayed in decimal or hexadecimal in the two measurement windows and till 60 parameters can be calibrated using the calibration window shown in Fig. 12.

A maximum number of five internal values may be selected to be displayed in the graphical view window as is shown in Fig.13. The last tabbed window allows the user to view a text description of the messages sent and received but the XCP driver of the master station.

The File menu is for configuration management (with standard *new*, *open*, *save* and *close* as basic operations), while the Configuration menu may be used to *add*, *edit* or *remove* parameters or internal values together with their attributes within any of the three representation windows described above.

V. Conclusions

The XCP Interpreter application for CAN-based networks described in this paper is a simple yet powerful tool that shall be of real help in assisting the developer or the integrator engineer for in-network debugging and calibration of Embedded Control Units in automotive

applications or in any other CAN-based industrial network application.

Its flexibility and reduced implementation costs make from XCP Interpreter a good alternative to the much more expensive professional tools in this field available on the market. Future work will consider a complete implementation of the basic functionality according to XCP specifications, with the addition of other professional features.

The use of a new module to automatically extract configuration information directly from the *map* or *elf* binary application files of the slave devices instead of the actual use of ASCII configuration files represents another way to improve and simplify the user interface. Also, the user interface may be enhanced by adding new features for numerical as well as graphical data representation and conversion to engineering units.

REFERENCES

[1] GQ Ao, H. Zhong, JX Qiang, L. Yang, B. Zhuo, "The development of a new communication protocol and calibration system for hybrid electric vehicle" in *Proceedings of the 2006 IEEE International Conference on Vehicular Electronics and Safety*, Shanghai, 2006, pp. 465-469

[2] S.W. Yang , L. Yang, and B. Zhuo, "Developing a multi-node calibration system for CAN bus based vehicle", in *Proceedings of the 2006 IEEE International Conference on Vehicular Electronics and Safety*, Shanghai, 2006, pp.199-203

[3] H. Yang and Xiaomin S, "An XCP Based Distributed Calibration System" in *Advances in Sofware Engineering,* Edts. D. Ślęzak, T. Kim, A. Kiumi, T. Jiang, J. Verner and S. Abrahão, Volume 59, pp. 9-15, Springer Berlin Heidelberg, 2009

[4] *CCP – CAN Calibration Protocol v.2.1*, *ASAP standard*, 1999

[5] *XCP v.1.0 – The Universal Measurement and calibration Protocol Family*, *ASAM standard*, 2003.

Robust Maximum Principle and all around

Aleksander S. Poznyak*

In this lecture we present a new version of Maximum Principle recently designed especially for the construction of optimal control strategies for the class of uncertain systems given by a system of ordinary differential equations with unknown parameters from a given set (finite or compact) that corresponds to different scenarios of possible dynamics. Such sort of problems, dealing with finite uncertain sets, are very common, for example, for Reliability Theory where some of sensors or actuators can fail changing completely the structure of a system to be controlled (each of possible structures can be associated with one of the fixed parameter values). The problem under consideration belongs to the class of optimization problems of min-max type. The proof is based on the Tent Method (suggested by V.G.Boltyanski in 1975) which in details is also discussed. We show that in general case the original problem can be converted to the analysis of non-solid convex cones that leads to inapplicability of the method of Dubovitzkii and Milyutin (1965) for deriving the corresponding necessary conditions of optimality whenever the Tend Method remains still working. The present lecture is both a refinement and an extension of the author earlier publications and consists of three complementary parts.

Part 1 deals with a Motivation related to a min-max static optimization problem. The following simple single-dimensional optimization problem is considered

$$\min_{u\in\mathbb{R}}\max_{\alpha\in\mathscr{A}} h^{\alpha}(u)$$

where $h^{\alpha}:\mathbb{R}\mapsto\mathbb{R}$ is a differentiable *strictly convex* function and

$$\mathscr{A}=\{\alpha_1\equiv 1,\alpha_2\equiv 2,\ldots,\alpha_N\equiv N\}$$

is a simple finite set containing only N possible parameter values, that is

$$\min_{u\in\mathbb{R}}\max\{h^1(u),h^2(u),\ldots,h^N(u)\}$$

It is shown that

- The joint Hamiltonian $H(u,\lambda)$ of the initial optimization problem is equal to the integral of the individual Hamiltonians $H_i(u,\lambda_i),(i=1,\ldots,N)$ calculated over the given compact $\mathscr{A}$;

- In the optimal point u^* all loss functions $h^{\alpha}(u^*)$, corresponding to active indices for which $\lambda^*_{\alpha}>0$, are equal.

The main question arising here is: "*Do these two principle properties*, formulated in the Propositions above for finite-dimensional min-max problems, *remain to be valid for infinite-dimensional case*, formulated in a Banach space for a min-max optimal control problem?" The answer is:

YES! they do!

The detailed justification of this positive answer forms the main contribution of this lecture.

Part 2 represents a robust version of Maximum Principle dealing with construction of the min-max control strategies for the class of uncertain systems given by an ordinary differential equation with unknown parameters from a given compact set. A finite collection of parameters corresponds to different sceneries of possible dynamics. The proof is based on the Tent Method justified in the previous part of the book. The min-max Linear Quadratic (LQ) Control Problem is considered in the details. It is shown that the design of the min-max optimal controller in this case may be reduced to a finite-dimensional optimization problem given at the corresponding simplex set containing the weight parameters to be found. The robust LQ optimal control may be interpreted as a mixture (with the optimal weights) of the controls which are optimal for each fixed parameter value. A robust time-optimality is also considered (as a particular case of Lagrange problem). Usually, the Robust Maximum Principle appears only as a necessary condition for robust optimality. But the specific character of linear time-optimization problem permits us to obtain more profound results. In particular, in this case the Robust Maximum Principle appears as the *necessary and sufficient condition*. Moreover, for the linear robust time-optimality it is possible to establish

*The author is with Department of Automatic Control, CINVESTAV-IPN, Mexico DF, Mexico. apoznyak@ctrl.cinvestav.mx

some additional results: *existence and uniqueness* of robust controls, *piecewise constancy* of robust controls for polyhedral resource set, the Fel'dbaums type estimate for *the number of intervals of constancy* (or "switching"). The comparison of the optimal controllers, designed by the Maximum Principle and Dynamic Programming for LQ-problems, is done. The application of the obtained results to Multimodel Sliding mode Control and Multimodel Differential Games are presented.

Our main purpose is to obtain the min-max control arising whenever the state of a system at time $t \in [0,T]$ as described by a vector

$$x(t) = (x_1(t), \ldots, x_n(t))^T \in \mathbb{R}^n$$

evolves accordingly to a prescribed law given usually in the form of a first-order vector ordinary differential equation

$$\dot{x}(t) = f^{\alpha}(x(t), u(t), t)$$

under the assignment of a vector valued control function

$$u(t) = (u_1(t), \ldots, u_r(t))^T \in \mathbb{R}^r$$

which is the control that may run over a given control region $U \subset \mathbb{R}^r$, and α is a parameter that may run over a given parametric set $\mathscr{A}$. On the right-hand side

$$f^{\alpha}(x,u,t) = (f_1^{\alpha}(x,u,t), \ldots, f_n^{\alpha}(x,u,t))^T \in \mathbb{R}^n$$

we impose the usual restrictions: *continuity* with respect to the collection of the arguments x, u, measurability on t and *differentiability* (or Lipschitz condition) with respect to x. Here we will assume that admissible $u(t)$ may be only piecewise continuous at each time interval from $[0,T]$ (T is permitted to be vary). Controls which have the same values except at common points of discontinuity will be considered as identical.

Part 3 deals with designing the *Robust Maximum Principle for Stochastic Systems* given by stochastic differential equations (with the It integral implementation) and subject to terminal constraints.

The main goal of this part is to illustrate the possibilities of the MP approach for a class of min-max control problems for uncertain systems given by a system of linear stochastic differential equations with a *controlled drift and diffusion terms* and unknown parameters within a given finite and, in general, compact uncertainty set supplied by a given measure. The problems for finite uncertain sets are very common, for example, for Reliability Theory where some of sensors or actuators can fail changing completely the structure of a system to be controlled (each of possible structures can be associated with one of the fixed parameter values). If in the deterministic case the adjoint equations are backward ordinary differential equations and represent, in some sense, the same forward equation but in reverse time, in the stochastic case such interpretation is not applicable because any time reversal may destroy the non - anticipativeness of the stochastic solutions, that is, any obtained robust control should be independent of the future.The proof the Robust Maximum Principle is also based on the use of the Tent Method but with the special technique specific for stochastic calculus. The Hamiltonian function used for these constructions is equal to the Lebesgue integral over the given uncertainty set of the standard stochastic Hamiltonians corresponding to a fixed value of the uncertain parameter. Two illustrative examples, dealing with production planning and reinsurance-dividend management, conclude this part.

Most of the material, presented in this lecture, has been class-tested in the "Steklov" Mathematical Institute (Moscow, 1962-1980), in the Institute of Control Sciences (Moscow, 1978-1993), in the Mathematical Investigation Center of Mexico (CIMAT, Guanojuato, 1995-2006) and in the Center of Investigation and Advance Education of IPN (CINVESTAV, Mexico, 1993-2009). Some studies, dealing with multimodel sliding-mode control and multi-model differential games, present the main results of Ph.D.-thesis of students defended during the last years.

Tensor Product Models for Automotive Applications

Radu-Emil Precup, *Senior Member, IEEE*, Claudia-Adina Dragoş, Stefan Preitl, *Senior Member, IEEE*, Mircea-Bogdan Rădac, and Emil M. Petriu, *Fellow, IEEE*

***Abstract*—This paper offers original Tensor Product (TP) models for two controlled plants in automotive applications, a magnetically actuated mass-spring-damper system, and a drive line system consisting of an internal combustion engine, a continuously variable transmission, a final reduction gear, a flexible drive shaft and a load. The modeling starts with the Linear Parameter-Varying (LPV) models which are derived from the first principle mathematical models of the controlled plants. The LPV models are then transformed into the TP models on the basis of the TP Tool. Digital simulation results are included to validate the new models.**

I. Introduction

THE tensor product (TP)-based model transformation of dynamic systems transforms the models defined over bounded domains into polytopic or Takagi-Sugeno fuzzy models [1]–[3]. The TP-based model transformation offers a convenient way to transform the popular Linear Parameter-Varying (LPV) dynamic models into parameter-varying weighted combinations of Linear Time-Invariant (LTI) systems. Optimization constraints are taken into consideration in the LPV model transformation.

The main advantages of the TP-based model transformation of LPV models into convex combinations of LTI models are the possibility to allow the application of the Linear Matrix Inequality (LMI) and Parallel Distributed Compensation (PDC) frameworks to the obtained affine models [2], [5]–[8]. TP models and several controllers for the TORA system are suggested in [9], [10]. TP-based controls of a prototypical aeroelastic wing section and a two-dimensional aeroelastic system are analyzed in [11] and [12], respectively. Applications to laboratory gantry crane control systems and parallel-type double inverted pendulums are given in [13], [14]. The combination of LMI and PDC with TP models for the state feedback control design of the translational oscillations is treated in [15]. TP friction models are investigated in [16]. Recent TP-based modeling and control applications include the electrical drives, the control of heavy vehicles, the temperature control and the stabilization of the 3-DOF RC helicopters [17]–[20].

This paper considers the controlled plants in two automotive applications, a magnetically actuated mass-spring-damper (MAMSD) system [21] and a drive line system consisting of an internal combustion engine, a Continuously Variable Transmission (CVT), a Final Reduction Gear (FRG), a flexible drive shaft and a load [22]. Both applications can be viewed as controlled plants in automotive and mechatronics benchmarks [23], [24]. Original TP models are suggested for the two applications.

A unified modeling approach is conducted for both applications. It starts with the derivation of the LPV models from the first principle mathematical models of the controlled plants. The TP Tool [25] is next used to transform the LPV models into the TP ones.

The paper is structured as follows. The TP model for the MAMSD system is derived in the next section. Section III is focused on the derivation of the TP model for the drive line system. Digital simulation results are presented in Section II and Section III to validate the new TP models. The conclusions are highlighted in Section IV.

II. Tensor Product Model for Magnetically Actuated Mass-Spring-Damper System

The state-space mathematical model of the MAMSD system as controlled plant to be used in an electromagnetic actuator is [21], [26]–[29]

$$\begin{aligned}
\dot{x}_1 &= x_2,\\
\dot{x}_2 &= -\frac{k}{m}x_1 - \frac{c}{m}x_2 + \frac{k_a}{m(k_b+d-x_1)}x_3^2,\\
\dot{x}_3 &= \frac{R}{2k_a}x_1x_3 - \frac{R(k_b+d)}{2k_a}x_3 - \frac{1}{2k_a}x_1V +\\
&\quad + \frac{(k_b+d)}{2k_a}V + \frac{1}{(k_b+d-x_1)}x_2x_3,\\
y &= x_1,
\end{aligned} \tag{1}$$

where: $x_1\,(\mathrm{m}) = x$ – the mass position, $x_2\,(\mathrm{m/s}) = \dot{x}$ – the mass speed, $x_3\,(\mathrm{V\cdot s}) = \lambda$ – the magnetic flux, $V\,(\mathrm{V})$ – the control signal, $y\,(\mathrm{m})$ – the controlled output, $k\,(\mathrm{N/m})$ – the stiffness of the spring, $c\,(\mathrm{N\cdot s/m})$ – the coefficient of the damper, $R\,(\Omega)$ – the resistance of the resistive circuit,

Manuscript received April 10, 2010. This work was supported in part by the CNMP and CNCSIS of Romania.

R.-E. Precup, C.-A. Dragoş, S. Preitl and M.-B. Rădac are with the "Politehnica" University of Timisoara, Department of Automation and Applied Informatics, Bd. V. Parvan 2, RO-300223 Timisoara, Romania (phone: +40 2564032 -29, -30, -26; fax: +40 2564032 -14; e-mail: radu.precup@aut.upt.ro, claudia.dragos@aut.upt.ro, spreitl@aut.upt.ro, mircea.radac@aut.upt.ro).

E. M. Petriu is with the University of Ottawa, School of Information Technology and Engineering, 800 King Edward, Ottawa, ON, K1N 6N5 Canada (e-mail: petriu@site.uottawa.ca).

subject to magnetic flux variation, according to Faraday's law, d (m) – the distance between the contact position and the spring neutral position, and k_a (V^2s^2/N), k_b (m) – the constants in the relation between the magnetic flux and the current [21]. The state vector corresponding to (1) is

$$\mathbf{x} = [x_1 = x \quad x_2 = \dot{x} \quad x_3 = \lambda]^T \in R^3 . \tag{2}$$

Accepting the bounded parameter vector $\mathbf{p}$ which is the scalar λ in this case,

$$\mathbf{p} = \lambda \in R , \tag{3}$$

the model (1) is expressed as the LPV state-space model

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}(\mathbf{p})\mathbf{x} + \mathbf{B}(\mathbf{p})\mathbf{u}, \\ y &= \mathbf{C}(\mathbf{p})\mathbf{x} + \mathbf{D}(\mathbf{p})\mathbf{u}, \end{aligned} \tag{4}$$

where $\mathbf{u} = V \in R$, and the expressions of the matrices are

$$\mathbf{A}(\mathbf{p}) = \begin{bmatrix} 0 & 1 & 0 \\ -k/m & -c/m & \lambda/(4k_a m) \\ \lambda R/(2k_a) & 0 & -R(k_b + d)/(2k_a) \end{bmatrix}, \tag{5}$$

$$\mathbf{B}(\mathbf{p}) = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \mathbf{C}(\mathbf{p}) = \mathbf{C} = \mathbf{I}_3, \mathbf{D}(\mathbf{p}) = \mathbf{D} = [0],$$

with $\mathbf{I}_3$ – the third order identity matrix.

Introducing the parameter-varying system matrix

$$\mathbf{S}(\mathbf{p}) = \begin{pmatrix} \mathbf{A}(\mathbf{p}) & \mathbf{B}(\mathbf{p}) \\ \mathbf{C}(\mathbf{p}) & \mathbf{D}(\mathbf{p}) \end{pmatrix} \in R^{6\times 4}, \tag{6}$$

the model (4) becomes

$$\begin{pmatrix} \dot{\mathbf{x}} \\ y \end{pmatrix} = \mathbf{S}(\mathbf{p}) \begin{pmatrix} \mathbf{x} \\ \mathbf{u} \end{pmatrix}. \tag{7}$$

However the matrices $\mathbf{C}$ and $\mathbf{D}$ are constant with respect to $\mathbf{p}$ such that the following simpler expression of the matrix $\mathbf{S}(\mathbf{p})$ in (6) is obtained:

$$\mathbf{S}(\mathbf{p}) = (\mathbf{A}(\mathbf{p}) \quad \mathbf{B}(\mathbf{p}) \in R^{3\times 4} . \tag{8}$$

and the state-space mathematical model (1) is then expressed as the LPV state-space model

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{S}(\mathbf{p}) \begin{pmatrix} \mathbf{x} \\ \mathbf{u} \end{pmatrix}, \\ y &= \mathbf{C}\,\mathbf{x} + \mathbf{D}\,\mathbf{u} . \end{aligned} \tag{9}$$

The goal of the TP-based model transformation is to transform the LPV state-space model of the controlled plant in (9) into the following parameter-varying combination of LTI system matrices $\mathbf{S}_i = [\mathbf{A}_i \quad \mathbf{B}_i]$, also called vertex systems:

$$\begin{aligned} \dot{\mathbf{x}} &= S \overset{N}{\underset{n=1}{\otimes}} \mathbf{w}_n(\mathbf{p}_n) \begin{pmatrix} \mathbf{x} \\ \mathbf{u} \end{pmatrix} = \\ &= \sum_{i=1}^{I} w_{1,i}(\lambda) \mathbf{S}_i \begin{pmatrix} \mathbf{x} \\ \mathbf{u} \end{pmatrix}, \\ y &= \mathbf{C}\,\mathbf{x} + \mathbf{D}\,\mathbf{u} , \end{aligned} \tag{10}$$

where the row matrix $\mathbf{w}_n(\mathbf{p}_n)$ contains a bounded variable and its continuous weighting functions, N is the tensor's dimension, and S is the (N+2)-dimensional coefficient tensor. The TP model (10) is convex because the weighting functions fulfill certain conditions.

The TP Tool [25] is used to transform the TP model (10) into several polytopic forms depending on the number of singular values and the number of shapes of weighting functions. The following polytopic form results (for $I = 2$ in (10)) in terms of keeping the maximum singular values and of using the normal weighting functions (Fig. 1):

$$\begin{aligned} \dot{\mathbf{x}} &= \sum_{i=1}^{2} w_{1,i}(\lambda)(\mathbf{A}_i \mathbf{x} + \mathbf{B}_i \mathbf{u}), \\ y &= \mathbf{C}\,\mathbf{x} + \mathbf{D}\,\mathbf{u} . \end{aligned} \tag{11}$$

Two digital simulation scenarios were applied to validate the TP model (10). The first scenario deals with the 0.475 step modification of the control signal, and the digital simulation results are presented in Fig. 2. The second scenario concerns the digital simulation results for the behavior of the TP model (11) with respect to the variable step modification of the control signal according to Fig. 3 (a), and the results are presented in Fig. 3 (b).

For the sake of simplicity only a sample of digital simulation results is presented here. The sample illustrates the variations of the controlled output (mass position) versus time in both simulation scenarios.

The small differences between the results in Fig. 2 and Fig. 3 validate the TP model (10) because the maximum error between the original state-space model (1) and the TP model (10) is $7.3775 \cdot 10^{-11}$, and the root mean square error is $4.3122 \cdot 10^{-11}$. Therefore the TP model (10) has enough accuracy to model well the controlled plant of the MAMSD system.

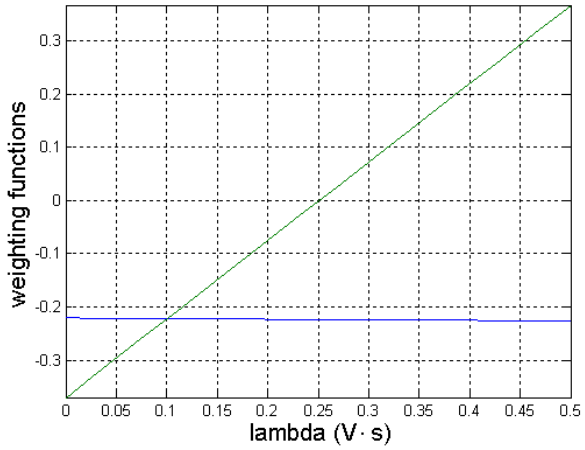


Fig. 1. Weighting functions of the TP model (10).

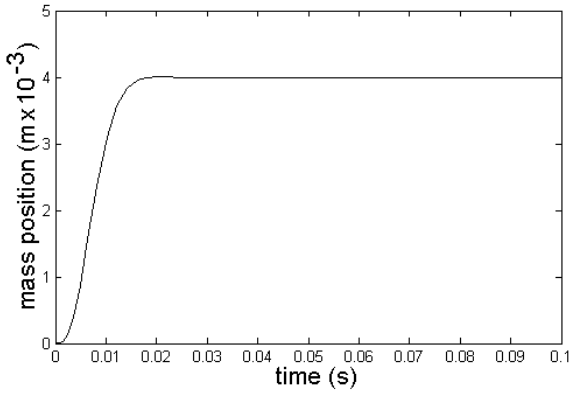


Fig. 2. Mass position versus time for the TP model (10) in the first digital simulation scenario.

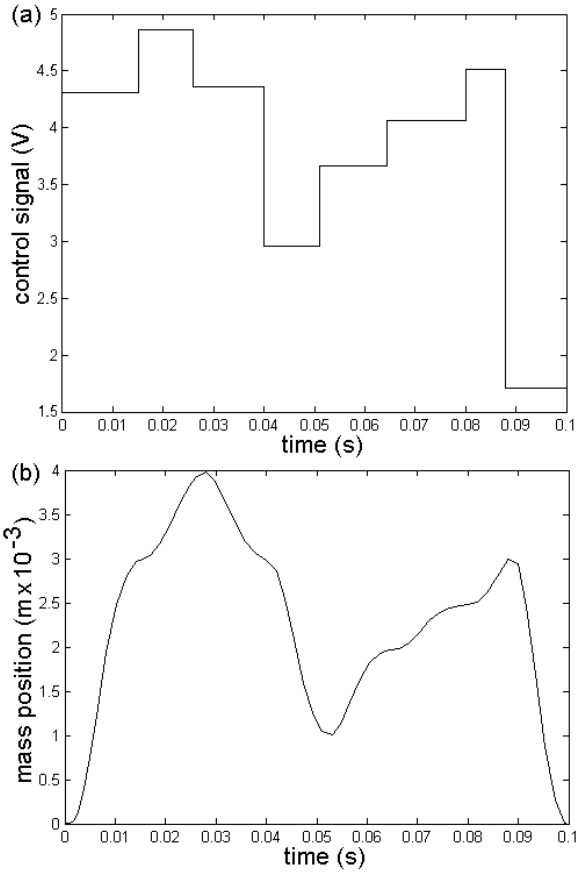


Fig. 3. Control signal versus time (a) and mass position versus time (b) for the TP model (10) in the second digital simulation scenario.

III. Tensor Product Model for Drive Line System

The first principle mathematical model of the drive line system consisting of an internal combustion engine, a CVT, an FRG, a flexible drive shaft and a load is [22], [30]

$$
\begin{aligned}
&T_{eng} - T_1 = J_{eng}\dot{\omega}_{eng},\ T_{eng} = \Gamma(\omega_{eng}),\\
&\omega_2 = i_{CVT}\omega_{eng},\ T_1 = T_2 i_{CVT}/\eta_{CVT},\\
&\omega_3 = i_{FRG}\omega_2,\ T_2 = T_3 i_{FRG}/\eta_{FRG},\\
&T_3 = T_k + T_b,\ T_k = k\int_0^t(\omega_3(\tau)-\omega_W(\tau))d\tau,\\
&T_b = b(\omega_3-\omega_W),\ T_3 - T_{Roll} - T_{Drag} - T_{Dist} = J_{veh}\dot{\omega}_W,
\end{aligned}
\tag{12}
$$

where: $T_{eng}\,(\mathrm{N\cdot m})$ – the engine torque, $\omega_{eng}\,(\mathrm{rad/s})$ – the angular velocity of the engine, $\Gamma(\omega_{eng})$ – the nonlinear engine map with an example afferent to an internal combustion engine given in [22], $J_{eng}\,(\mathrm{kg\cdot m^2})$ – the moment of inertia of the engine, $T_1\,(\mathrm{N\cdot m})$ and $T_2\,(\mathrm{N\cdot m})$ – the engine-CVT and CVT-FRG torques, respectively, i_{CVT} – the kinematic CVT ratio, η_{CVT} – the CVT load-efficiency, $\omega_2\,(\mathrm{rad/s})$ and $\omega_3\,(\mathrm{rad/s})$ – the angular velocities of the CVT and of the FRG, respectively, i_{FRG} – the kinematic FRG ratio, η_{FRG} – the FRG load-efficiency, $T_3\,(\mathrm{N\cdot m})$ – the drive shaft torque, $T_{Roll}\,(\mathrm{N\cdot m})$ and $T_{Drag}\,(\mathrm{N\cdot m})$ – the load torques due to rolling resistance and aerodynamic drag, respectively, $T_{Dist}\,(\mathrm{N\cdot m})$ – the algebraic sum of torques due to hill climbing and all other disturbances, $J_{veh}\,(\mathrm{kg\cdot m^2})$ – the equivalent vehicle moment of inertia, $\omega_W\,(\mathrm{rad/s})$ – the angular velocity of the wheels, and $T_k\,(\mathrm{N\cdot m})$, $T_b\,(\mathrm{N\cdot m})$, $k\,(\mathrm{N\cdot m/rad})$ and $b\,(\mathrm{N\cdot m\cdot s/rad})$ – the parameters which describe the dynamics of the spring specific to the FRG.

The model (12) is expressed as the LPV state-space model

$$
\begin{aligned}
\dot{\mathbf{x}} &= \mathbf{A}(\mathbf{p})\mathbf{x} + \mathbf{B}(\mathbf{p})\mathbf{u},\\
\mathbf{y} &= \mathbf{C}(\mathbf{p})\mathbf{x} + \mathbf{D}(\mathbf{p})\mathbf{u},
\end{aligned}
\tag{13}
$$

where the state vector and the controlled output vector are $\mathbf{x}$ and $\mathbf{y}$, respectively:

$$
\mathbf{x} = [x_1 = \omega_{eng} \quad x_2 = T_k \quad x_3 = \omega_W]^T \in R^3, \tag{14}
$$

$$\mathbf{y} = [y_1 = T_3 \quad y_2 = \omega_W]^T \in R^2, \tag{15}$$

and the control input (in the vector **u**) is usually the variable i_{CVT}.

The parameter vector in (13) is

$$\mathbf{p} = [p_1 = i_{CVT} \quad p_2 = m \quad p_3 = \omega_W]^T \in R^3, \tag{16}$$

where the parameter m results from the linrearization of the nonlinear engine map $\Gamma(\omega_{eng})$:

$$\Gamma(\omega_{eng}) = \Gamma_0 + m\,\omega_{eng}. \tag{17}$$

The relation (17) leads to the following control input vector which ensures the transformation of (12) into (13):

$$\mathbf{u} = [u_1 = i_{CVT} \quad u_2 = \Gamma_0 \quad u_3 = T_{Roll} + T_{Dist}]^T \in R^3. \tag{18}$$

Therefore the matrices introduced in the model (13) are then expressed as

$$\begin{aligned}
&\mathbf{A}(\mathbf{p}) = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ -k \cdot i_{FRG} p_1 & 0 & -k \\ b \cdot i_{FRG} p_1 / J_{veh} & 1/J_{veh} & a_{33} \end{bmatrix}, \\
&a_{11} = [p_2 - b \cdot i_{FRG}^2 p_1^2 / (\eta_{CVT} \eta_{FRG})] / J_{eng}, \\
&a_{12} = -i_{FRG} p_1 / (\eta_{CVT} \eta_{FRG} J_{eng}), \\
&a_{13} = b \cdot i_{FRG} p_1 / (\eta_{CVT} \eta_{FRG} J_{eng}), \\
&a_{33} = -b / J_{veh} - c\, p_3 / J_{veh}, \\
&\mathbf{B}(\mathbf{p}) = \begin{bmatrix} 0 & 1/J_{eng} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1/J_{veh} \end{bmatrix}, \\
&\mathbf{C}(\mathbf{p}) = \begin{bmatrix} b \cdot i_{FRG} p_1 & 1 & -b \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{D}(\mathbf{p}) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.
\end{aligned} \tag{19}$$

The parameter-varying system matrix

$$\mathbf{S}(\mathbf{p}) = \begin{pmatrix} \mathbf{A}(\mathbf{p}) & \mathbf{B}(\mathbf{p}) \\ \mathbf{C}(\mathbf{p}) & \mathbf{D}(\mathbf{p}) \end{pmatrix} \in R^{5\times 6}, \tag{20}$$

transforms the model (13) into

$$\begin{pmatrix} \dot{\mathbf{x}} \\ \mathbf{y} \end{pmatrix} = \mathbf{S}(\mathbf{p}) \begin{pmatrix} \mathbf{x} \\ \mathbf{u} \end{pmatrix}. \tag{21}$$

The LPV state-space model of the controlled plant given in (21) is next transformed into the TP model

$$\begin{pmatrix} \dot{\mathbf{x}} \\ \mathbf{y} \end{pmatrix} = S \otimes_{n=1}^{N} \mathbf{w}_n^T(\mathbf{p}_n) \begin{pmatrix} \mathbf{x} \\ \mathbf{u} \end{pmatrix}, \tag{22}$$

with the continuous weighting functions $\mathbf{w}_n^T = [w_1 \quad w_2 \quad w_3] \in R^3$. The application of the TP Tool leads to the polytopic form

$$\begin{aligned}
&\begin{pmatrix} \dot{\mathbf{x}} \\ \mathbf{y} \end{pmatrix} = \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} w_{1,i}(p_1) w_{2,j}(p_2) w_{3,k}(p_3) \mathbf{S}_{i,j,k} \times \\
&\times \begin{pmatrix} \mathbf{x} \\ \mathbf{u} \end{pmatrix}, \\
&\mathbf{S}_{i,j,k} = \begin{pmatrix} \mathbf{A}_{i,j,k} & \mathbf{B}_{i,j,k} \\ \mathbf{C}_{i,j,k} & \mathbf{D}_{i,j,k} \end{pmatrix},
\end{aligned} \tag{23}$$

with the normal weighting functions (Fig. 4) obtained for the maximum singular values, $I = 3$, $J = 2$ and $K = 2$.

The digitally simulated behavior of the TP model (24) of the controlled plant with respect to the 0.0232 step modification of i_{CVT} is presented in Fig. 5. The maximum error between the first principle model in (12) and the TP model in (22) is $1.199 \cdot 10^{-11}$, and the root mean square error is $6.0512 \cdot 10^{-12}$, so this new TP model dedicated to the drive line system as controlled plant is validated.

The TP model expressed in (22) has enough accuracy to model well the controlled plant of the drive line system. Both outputs $y_1 = T_3$ and $y_2 = \omega_W$ can be used as controlled outputs in the control systems.

IV. Conclusion

The paper has suggested original TP models for two automotive applications. The digital simulation results validate the models.

The relevance of the TP models presented and simulated in this paper is that they allow the modification of the parameter varying convex combination according to the designer's option. The type of the convex combination considerably influences the further LMI design and resulting control performance.

The polytopic models derived in this paper can be used further because they enable the convenient design of the controllers for two plants. The design can be based on the manipulation of the convex hulls beside the manipulation of the LMIs. Future research will be focused on the combination of several control solutions using other models and system structures [31]–[34] while implementing low-cost automation solutions assisted by analyses [35]–[39].

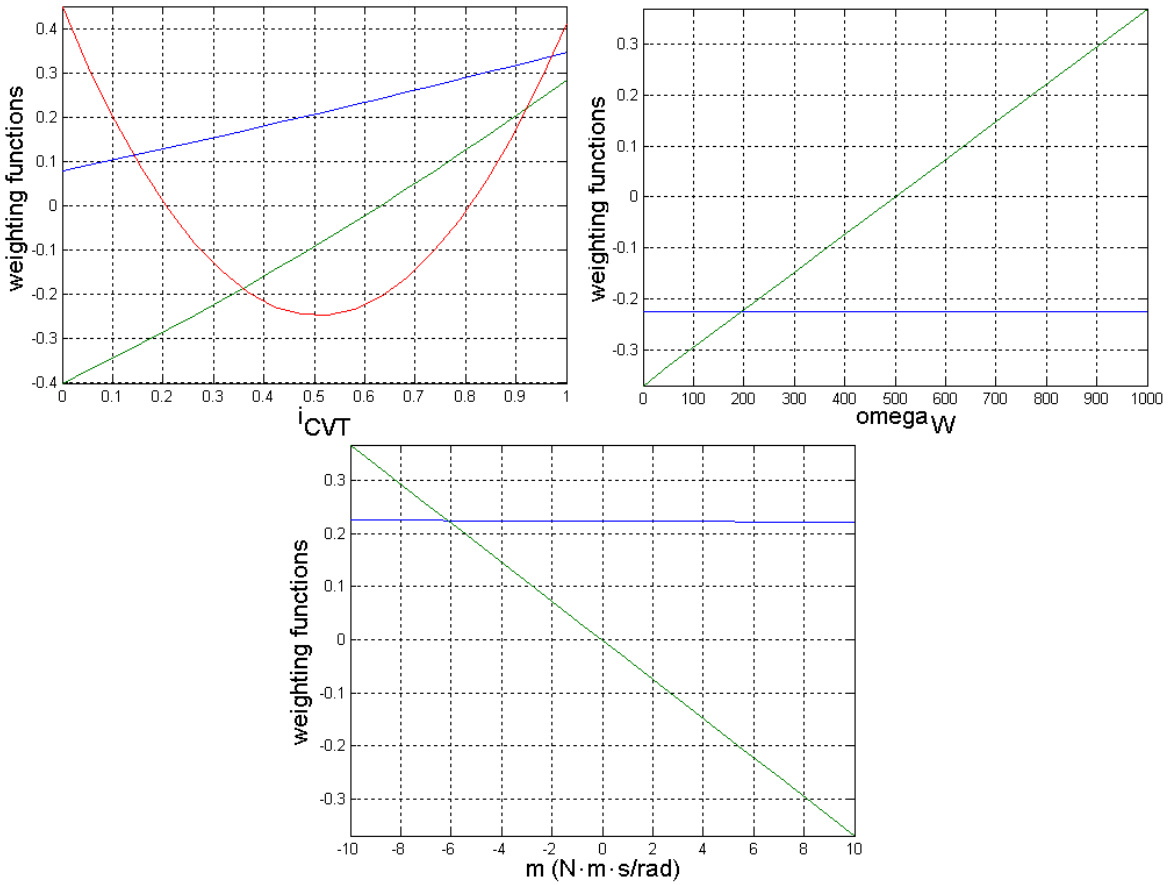


Fig. 4. Weighting functions of the TP model (22).

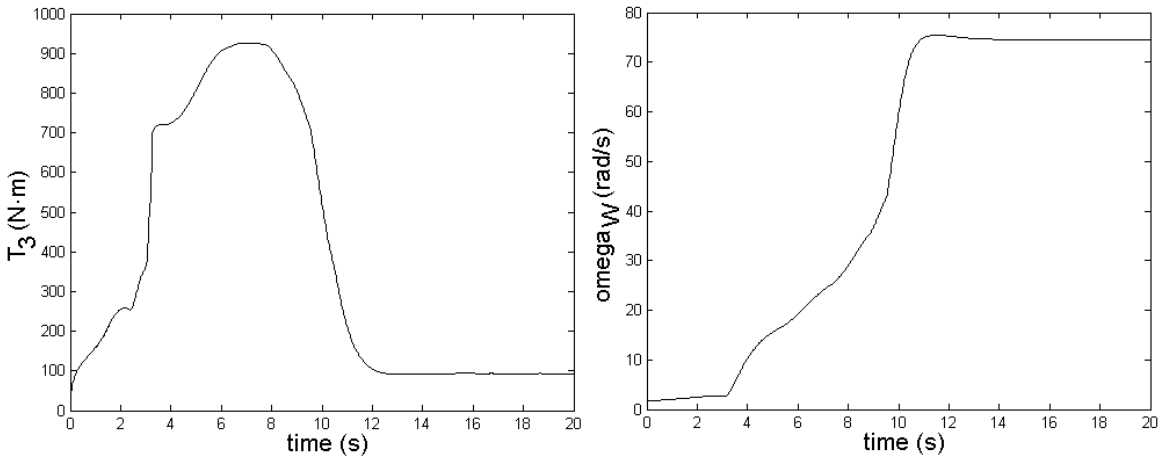


Fig. 5. Digital simulation results for the TP model (22) with respect to the 0.0232 step modification of i_{CVT}.

ACKNOWLEDGMENT

This work was supported by the CNMP and CNCSIS of Romania. This work was supported by the ETOCOM project (TÁMOP-4.2.2-08/1/KMR-2008-0007) through the Hungarian National Development Agency in the framework of Social Renewal Operative Program supported by EU and co-financed by the European Social Fund. This work was partially supported by the strategic grant POSDRU 6/1.5/S/13 (2008) of the Ministry of Labor, Family and Social Protection, Romania, co-financed by the European Social Fund – Investing in People.

REFERENCES

[1] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, pp. 1253–1278, March-May 2000.

[2] P. Baranyi, "TP model transformation as a way to LMI based controller design," *IEEE Trans. Ind. Electron.*, vol. 51, pp. 387–400, April 2004.

[3] S. Nagy, P. Baranyi, and Z. Petres, "Centralized tensor product model form," in *Proc. 6th Int. Symp. Appl. Machine Intelligence and Informatics (SAMI 2008)*, Herlany, Slovakia, 2008, pp. 189–193.

[4] A. Matszangosz, S. Nagy, and P. Baranyi, "Numerical convex hull manipulation under tensor product transformation based control design," in *Proc. IEEE Int. Conf. Computational Cybernetics (ICCC 2008)*, Stara Lesna, Slovakia, 2008, pp. 169–172.

[5] P. Baranyi, D. Tikk, Y. Yam, and R. J. Patton, "From differential equations to PDC controller design via numerical transformation," *Comp. Ind.*, vol. 51, pp. 281–297, Aug. 2003.

[6] C. Ariño and A. Sala, "Relaxed LMI conditions for closed-loop fuzzy systems with tensor-product structure," *Eng. Appl. Artif. Intell.*, vol. 20, pp. 1036–1046, Dec. 2007.

[7] A. Sala and C. Ariño, "Asymptotically necessary and sufficient conditions for stability and performance in fuzzy control: Applications of Polya's theorem," *Fuzzy Sets Syst.*, vol. 158, pp. 2671–2686, Dec. 2007.

[8] A. Sala, "On the conservativeness of fuzzy and fuzzy-polynomial control of nonlinear systems," *Annu. Rev. Control*, vol. 33, pp. 48–58, April 2009.

[9] Z. Petres, B. Resko, and P. Baranyi, "Reference signal control of the TORA system: A TP model transformation based approach," in *Proc. 2004 IEEE Int. Conf. Fuzzy Systems (FUZZ-IEEE 2004)*, Budapest, Hungary, 2004, pp. 1081–1086.

[10] G. Hancke and Á. Szeghegyi, "Nonlinear control via TP model transformation: the TORA system example," in *Proc. 2nd Slovakian-Hungarian Joint Symp. Applied Machine Intelligence (SAMI 2004)*, Herlany, Slovakia, 2004, pp. 333–340.

[11] P. Baranyi, "Tensor-product model-based control of two-dimensional aeroelastic system," *J. Guidance Control Dyn.*, vol. 29, pp. 391–400, May-June 2005.

[12] P. Baranyi, Z. Petres, P. L. Varkonyi, P. Korondi, and Y. Yam, "Determination of different polytopic models of the prototypical aeroelastic wing section by TP model transformation," *J. Adv. Comput. Intell. Intell. Inform.*, vol. 10, pp. 486–493, July 2006.

[13] F. Kolonic, A. Poljugan, and I. Petrovic, "Tensor product model transformation-based controller design for gantry crane control system - An application approach," *Acta Polytechnica Hungarica*, vol. 3, pp. 95–112, Dec. 2006.

[14] S. Nagy, Z. Petres, and P. Baranyi, "TP model transformation based controller design for the parallel-type double inverted pendulum," in *Proc. IEEE Intl. Conf. Fuzzy Systems (FUZZ-IEEE 2008)*, Hong Kong, 2008, pp. 1374–1380.

[15] Z. Petres, P. Baranyi, P. Korondi, and H. Hashimoto, "Trajectory tracking by TP model transformation: case study of a benchmark problem," *IEEE Trans. Ind. Electron.*, vol. 54, pp. 1654–1663, June 2007.

[16] Y. Kunii, B. Solvang, G. Sziebig, and P. Korondi, "Tensor product transformation based friction model," in *Proc. 11th Int. Conf. Intelligent Engineering Systems (INES 2007)*, Budapest, Hungary, 2007, pp. 259–264.

[17] P. Korondi, "Sector sliding mode design based on tensor product model transformation," in *Proc. 11th Int. Conf. Intelligent Engineering Systems (INES 2007)*, Budapest, Hungary, 2007, pp. 253–258.

[18] Z. Petres, S. Nagy, P. Gaspar, and P. Baranyi, "H_∞ gain-scheduling based control of the heavy vehicle model, a TP model transformation based control," in *Proc. IEEE Intl. Conf. Fuzzy Systems (FUZZ-IEEE 2008)*, Hong Kong, 2008, pp. 1542–1547.

[19] R.-E. Precup, S. Preitl, I.-B. Ursache, P. A. Clep, P. Baranyi, and J. K. Tar, "On the combination of tensor product and fuzzy models," in *Proc. IEEE Int. Conf. Automation, Quality and Testing, Robotics (AQTR 2008)*, Cluj-Napoca, Romania, 2008, vol. 2, pp. 48–53.

[20] P. Baranyi, P. Korondi, and K. Tanaka, "Parallel distributed compensation based stabilization of a 3-DOF RC helicopter: A tensor product transformation based approach," *J. Adv. Comput. Intell. Intell. Inform.*, vol. 13, pp. 25–34, Jan. 2009.

[21] S. Di Cairano, A. Bemporad, I. V. Kolmanovsky, and D. Hrovat, "Model predictive control of magnetically actuated mass spring dampers for automotive applications," *Int. J. Control*, vol. 80, pp. 1701–1716, Nov. 2007.

[22] M. Mussaeus, "Control issues of hybrid and conventional drive lines," M.Sc. thesis, Dept. Mech. Eng., Section Syst. Control, Eindhoven Univ. of Technology, Eindhoven, The Netherlands, 1997.

[23] U. Kiencke and L. Nielsen, *Automotive Control Systems for Engine, Driveline and Vehicle*, 2nd ed. Berlin, Heidelberg, New York: Springer-Verlag, 2005.

[24] R. Isermann, *Mechatronic Systems: Fundamentals*. Berlin, Heidelberg, New York: Springer-Verlag, 2005.

[25] S. Nagy, Z. Petres, and P. Baranyi, "TP Tool – A Matlab toolbox for TP model transformation," in *Proc. 8th Int. Symp. Hungarian Researchers on Computational Intelligence and Informatics (CINTI 2007)*, Budapest, Hungary, 2007, pp. 483–495.

[26] C.-A. Dragoş, S. Preitl, and R.-E. Precup, "Low cost Takagi-Sugeno fuzzy controller for an electromagnetic actuator," *Sci. Bull. "Politehnica" Univ. Timişoara, Romania, Trans. Automatic Control Comp. Sci.*, vol. 54(68), pp. 87–92, June 2009.

[27] C.-A. Dragoş, "Study concerning model based predictive control," PhD Report 1, "Politehnica" Univ. Timişoara, Timişoara, Romania, 2009 (in Romanian).

[28] C.-A. Dragoş, "Study concerning the modeling of nonlinear processes and control solutions," PhD Report 2, "Politehnica" Univ. Timişoara, Timişoara, Romania, 2009 (in Romanian).

[29] C. Lazăr et al., "Real-time informatics technologies for embedded-system-control of power-train in automotive design and applications," Research Report 1 of the SICONA CNMP Grant, "Gh. Asachi" Tech. Univ. Iaşi, Iaşi, Romania, 2009 (in Romanian).

[30] C. Lazăr et al., "Real-time informatics technologies for embedded-system-control of power-train in automotive design and applications," Research Report 2.1 of the SICONA CNMP Grant, "Gh. Asachi" Tech. Univ. Iaşi, Iaşi, Romania, 2009 (in Romanian).

[31] R. Dobrescu, V. E. Oltean, and M. Dobrescu, "Simulation models and Zeno path avoidance in a class of piecewise linear biochemical processes," *Control Eng. Appl. Inf.*, vol. 10, pp. 33–39, March 2008.

[32] A. E. Bălău, C. F. Căruntu, D. I. Pătraşcu, C. Lazăr, M. H. Matcovschi, and O. Păstrăvanu, "Modelling of a pressure reducing valve actuator for automotive applications," in *Proc. 3rd IEEE Multi-conf. Systems and Control (MSC 2009)*, Saint Petersburg, Russia, 2009, pp.1356–1361.

[33] D. I. Pătraşcu, A. E. Bălău, C. F. Căruntu, C. Lazăr, M. H. Matcovschi, and O. Păstrăvanu, "Modelling of a solenoid valve actuator for automotive control systems," in *Proc. 17th Intl. Conf. Control Systems and Computer Science (CSCS-17)*, Bucharest, Romania, 2009, pp. 541–546.

[34] V. E. Oltean, "On qualitative behaviours of a class of piecewise-linear control systems (Part II: A case study)," *Rev. Roum. Sci. Techn. - Électrotechn. et Énerg.*, vol. 54, pp. 205–212, June 2009.

[35] I. Škrjanc, S. Blažič, S. Oblak, and J. Richalet, "An approach to predictive control of multivariable time-delayed plant: Stability and design issues," *ISA Trans.*, vol. 43, pp. 585–595, Oct. 2004.

[36] Z. C. Johanyák and S. Kovács, "Fuzzy rule interpolation based on polar cuts," in *Computational Intelligence, Theory and Applications*, B. Reusch, Ed. Berlin, Heidelberg, New York: Springer-Verlag, 2006, pp. 499–511.

[37] L. Kovács and B. Paláncz, "Glucose-insulin control of type1 diabetic patients in H_2/H_∞ space via computer algebra," *Lect. Notes Comput. Sci.*, vol. 4545, pp. 95–109, Aug. 2007.

[38] J. Vaščák, "Fuzzy cognitive maps in path Planning," *Acta Technica Jaurinensis, Series Intelligentia Computatorica*, vol. 1, pp. 467–479, Dec. 2008.

[39] I. Harmati and K. Skrzypczyk, "Robot team coordination for target tracking using fuzzy logic controller in game theoretic framework," *Robot. Auton. Syst.*, vol. 57, pp. 75–86, Jan. 2009.

Path following with collision avoidance and velocity constraints for multi-agent group formations

Ionela Prodan[1], Sorin Olaru[1], Cristina Stoica[1], Silviu-Iulian Niculescu[2]

***Abstract*— This paper deals with avoidance constraints while following an optimal trajectory for a group of agents operating in open space. The basic idea is to use the Model Predictive Control (MPC) technique to solve a real time optimization problem over a finite time horizon. Following a specified trajectory, the agents move in the same direction and eventually they end up in a particular formation. Combining the optimization-based control study with the ability of MPC to handle convex and non-convex constraints allows a thorough analysis of the motion control of a group of agents with linear dynamics subject to state constraints. Avoidance constraints are also added to the optimization problem when the agents are operating in an environment with obstacles. A flat trajectory is planned in the physical open space allowing the agents to maneuver successfully in such a dynamic environment and to reach a common objective.**

I. INTRODUCTION

Questions about achieving a formation of a group of agents and how to ensure that all the agents avoid collision, both inside the group and with the obstacles around them arise when dealing with multi-agent systems [1]. The goal of this paper is to present a control approach for a set of subsystems having independent dynamics while achieving a common objective. The problem is relevant in many applications involving the control of cooperative systems [2], [3], [4]. Among the applications, we cite the characterization of pedestrian behavior in the crowd [5], [6], [7], [8]. Such a characterization is essential for evaluating the safety of the social infrastructures. An important property of these cooperating systems is that the group behavior is not imposed in a hierarchical manner by one of the agents, this behavior results from the local interaction between the agents and their neighbors. For instance, every pedestrian in a crowd knows where the other pedestrians in its neighborhood are heading, but it does not know the average heading of all pedestrians in the crowd.

In this paper we consider the Model Predictive Control (MPC), a widely used technique in control community due to its ability to handle control and state constraints, while offering good performance specifications [9], [10], [11], [12] [13], [14]. The particularity will be the use of a group of agents in a predictive control context, which enables the inclusion of state constraints, both for collision avoidance between the agents and for the velocity of each agent. Several related studies can be found in the literature as for example those for the design of vehicle formation through the use of MPC detailed in [15]. Other approaches can be found in aerospace applications, where MPC is applied to spacecraft formation keeping [16], but no avoidance constraints are considered. The collision avoidance between the agents is known to be a difficult problem, since certain constraints require the use of auxiliary binary variables in the modeling and optimization aspects of the control algorithms. In particular, in [14], the authors considered such an approach based on the use of auxiliary binary variables together with MPC, with connections to the developments in hybrid control systems. In [1], the collision and obstacle avoidance are included in the trajectory planning of spacecraft vehicles, but the velocity constraints are not taken into account.

Our paper considers a two-dimensional environment for the group of agents, with supplementary non-convex velocity constraints.

An important contribution to the previous work is the decrease of the complexity of the control design problem. This is obtained by reducing the number of auxiliary binary variables used to reformulate the non-convex state constraints in a linear form.

The path following problem formulated in a non-convex constrained predictive control framework is described from the standard centralized point of view as a collection of receding horizon mixed integer optimization problem. Using the decentralized predicted control actions, the agents move in the same direction following a specified trajectory. The imposed state constraints will enforce a certain safety distance, eventually, the agents ending up in a particular formation. The specified trajectory of the group of agents can be generated using the differential flatness formalism along the lines in [17], [18]. In [18], the use of MPC is combined with differential flatness formalism for trajectory generation of nonlinear systems. In conclusion, the goal of our paper is to achieve an agent group formation only by imposing constraints on the position and the velocity of the agents, while they follow a specified path.

The remaining paper is organized as follows. Section II introduces the agent dynamics and the reference trajectory generation mechanism. The constrained predictive control problem is then summarized in. Section III which deals with the linear reformulation of the state constraints for the real-time optimization problem. Section IV presents the MPC problem, where the generated trajectory is used by the group of agents for prediction in a centralized approach. Based on the information received from the MPC formulation, the

[1]SUPELEC Systems Sciences (E3S) - Automatic Control Department, 3 rue Joliot Curie, 91192, Gif sur Yvette cedex, France, {ionela.prodan,sorin.olaru,cristina.stoica}@supelec.fr

[2]Laboratory of Signals and Systems, SUPELEC - CNRS, 3 rue Joliot Curie, 91192, Gif sur Yvette, France, {ionela.prodan,silviu.niculescu}@lss.supelec.fr

avoidance and velocity constraints are taken into account, leading the agents to follow the reference trajectory in a formation which depends on the geometry of the constraints. In Section V, illustrative simulation results are presented. Finally, some conclusions are drawn in Section V.

Throughout the paper, the following notations are used. An intersection of finitely many halfspaces, a polytope denoted as P will be used to describe a safety region for an agent.

(x, y) - position coordinates of an agent
(v_x, v_y) - velocity coordinates of an agent
ξ - agent state
N_a - number of agents
i - the i-th agent
$P(\xi^i)$ - polyhedral safety region of the i-th agent
$\overline{P}(\xi^i)$ - the complement of the polyhedral safety region $P(\xi^i)$
N - prediction horizon
V_N - cost function
δ^c - binary variables $\{0, 1\}$

II. Problem Formulation

In this section, based on the model of individual agents, the principles of a prediction based optimization problem is stated such that the group converge to a fix formation. Imposing the specified state constraints the agents will preserve a safety distance in-between, thus allowing collision avoidance both inside the group and with obstacles. Non-convex velocity constraints can be considered in the same formulation.

A. Model description

Let us consider a linear system (vehicle, pedestrian or agent in a general form) whose dynamics is modeled by the following equation:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{v_x} \\ \dot{v_y} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{\mu}{m} & 0 \\ 0 & 0 & 0 & -\frac{\mu}{m} \end{bmatrix} \begin{bmatrix} x \\ y \\ v_x \\ v_y \end{bmatrix} + + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{1}{m} & 0 \\ 0 & \frac{1}{m} \end{bmatrix} \begin{bmatrix} u_x \\ u_y \end{bmatrix}, \quad (1)$$

where (x, y) are the position coordinates, (v_x, v_y) the velocity coordinates of the agent in a two-dimensional representation. The agent mass is denoted by m, and μ is the damping factor. By associating the index i to the i-th agent the following model is obtained:

$$\dot{\xi}^i(t) = A_c\xi^i(t) + B_c u^i(t),\ i = 1 : N_a, \quad (2)$$

with the corresponding state and input vectors:
$\xi^i = [x\ y\ v_x\ y_y]^T$, $u^i = [u_x\ u_y]^T$, and N_a the number of agents. A corresponding discrete-time model for the equations (2) is constructed upon a chosen sampling period T_e by considering the time instants $t_k = kT_e$:

$$\xi^i_{k+1} = A{\xi^i}_k + Bu^i_k, \quad k \in \mathbb{N}, \quad i = 1 : N_a, \quad (3)$$

where $\xi_k = \xi_{t_k}$, $u_k = u_{t_k}$. The pair (A, B) is given by:

$$A = e^{A_c T_e}, \quad B = \int_0^{T_e} e^{A_c(T_e-\theta)} B_c d\theta$$

For the collision avoidance problem let us consider a convex set, a polytope (in the state space) that describes a safety region around an agent i and also safety limits for the velocity of an agent i:

$$P(\xi^i) = \{\xi \in \mathbb{R}^{n_\xi} : H(\xi - \xi^i) \leq K\} \quad (4)$$

where $H \in \mathbb{R}^{n_c \times n_\xi}$, $K \in \mathbb{R}^{n_c \times 1}$, n_ξ is the state dimension and n_c is the number of halfspaces. The position of the agent i represents the center of the region defined by the projection of $P(\xi^i)$ on the position subspace of the state space. The feasible region in the space of solutions is the complement of the safety region, denoted by $\overline{P}(\xi^i)$, which can be described as a union of regions that cover all the space except the polytope $P(\xi^i)$. The velocity constraints imposed to the agent i represent for example safety limits, such as a minimum maneuvering velocity near an obstacle or another agent. Another example considers a spacecraft formation flying, where each agent has to keep its velocity grater than a specified value, even if the formation follows a trajectory with a relative velocity inferior to some pre-imposed bounds for each spacecraft.

For sake of completness, next we summarize the problem of generating a reference trajectory for the linear system (1), along the line in [17].

B. Trajectory generation

The idea is to find a trajectory $(\xi(t), u(t))$ that steers our model (1) from an initial state x_0 to a final state x_f, over a fixed time interval $[t_0, t_f]$. Using (see [17] for practical details of the construction) the system is parameterized in terms of a finite set of variables $z(t)$ and a finite number of their derivatives:

$$\begin{aligned} \xi(t) &= \xi(z(t), \dot{z}(t), \cdots, z^{(q)}(t)), \\ u(t) &= u(z(t), \dot{z}(t), \cdots, z^{(q)}(t)), \end{aligned} \quad (5)$$

where $z(t) = \Upsilon(\xi(t), u(t), \dot{u}(t), \cdots, u^{(q)}(t))$ is called the flat output. In order to generate the reference trajectory we use the class of polynomial functions. Using the parametrization (5) and imposing boundary constraints for the evolution of the differentially flat systems [18] one can generate a reference trajectory $z^{ref}(t)$ by the resolution of a linear system of equalities. Therefore the corresponding reference state and input for the system (1) are obtained by replacing the reference flat output $z^{ref}(t)$ with $t \in [t_0, t_f]$ in (5):

$$\begin{aligned} \xi^{ref}(t) &= \xi(z^{ref}(t), \dot{z}^{ref}(t)), \\ u^{ref}(t) &= u(\dot{z}^{ref}(t), \ddot{z}^{ref}(t)), \end{aligned} \quad (6)$$

where $t \in [t_0, t_f]$.

The flat trajectory can also be generated to enforce obstacle avoidance at the path planning stage. The idea is illustrated in figure (1). In this framework the obstacles can be modeled in terms of a convex safety region around each agent, as in (4). Even if the reference trajectory is generated over the entire interval $[t_0, t_f]$, intermediary points can be added along the system trajectory in order to avoid obstacles at a specific time subinterval by the redesign of the flat trajectory.

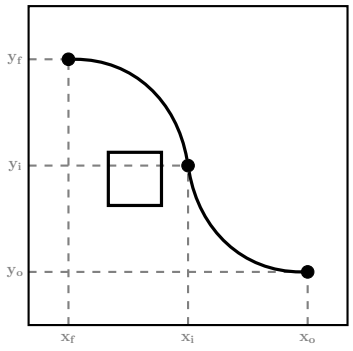


Fig. 1: A flatness trajectory for obstacle avoidance

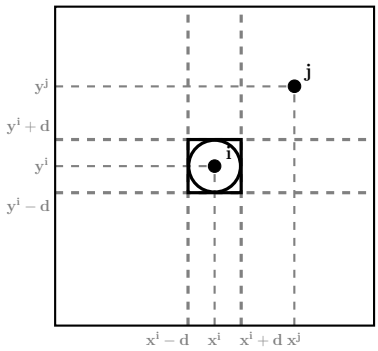


Fig. 2: Approximation of state constraints: the square approximates the regions with linear constraints

Since the reference trajectory is available beforehand, an optimization problem which minimizes the tracking error for the system can be formulated in a predictive control framework. Consequently the agents must follow the reference trajectory from the initial position to the desired position, using the available information over a finite time horizon N in the presence of constraints.

C. Constrained Predictive Control

The aim is to find the N-move control sequence $u^* = \{u_{k|k}, u_{k+1|k}, \cdots, u_{k+N-1|k}\}$ that minimizes the finite horizon quadratic objective function $V_N(\xi_k, u^*)$:

$$V_N = \xi^T_{k+N|k} P \xi_{k+N|k} + \sum_{l=1}^{N-1} \xi^T_{k+l|k} Q \xi_{k+l|k} + \sum_{l=0}^{N-1} u^T_{k+l|k} R u_{k+l|k}, \tag{7}$$

while respecting the constraints imposed by each agent dynamics (3), and the physical limitations

$$\xi^i \in \overline{P}(\xi^i), \quad i = 1 : N_a \tag{8}$$

Here $Q = Q^T \geq 0$, $R > 0$ are the weighting matrices and $P = P^T \geq 0$ defines the terminal cost. A finite horizon trajectory optimization is performed at each sample instant, the first component of the resulting control sequence is effectively applied and the optimization procedure is reiterated using the available measurements based on the receding horizon principle [13].

The constraints (8) describe a non-convex region in the state-space and thus, the MPC problem (7) can not be casted in the classical LP/QP parametric problem formulation. In the following we re-state the constraints in a linear form, by introducing a set of auxiliary binary variables, which have to be considered as decision variables in the new MPC formulation. It is worth mentioning that the problem can be interpreted as a hybrid system control problem [14].

III. Linear constraints reformulation

The constraints for 2 agents are discussed here, the generalization to N_a agents following the same lines. Let us consider the global model of any two different agents $(i, j) \in \mathbb{N}_{[1,N_a]} \times \mathbb{N}_{[1,N_a]}, i \neq j$:

$$\begin{bmatrix} \xi^i_{k+1} \\ \xi^j_{k+1} \end{bmatrix} = \begin{bmatrix} A_i & 0 \\ 0 & A_j \end{bmatrix} \begin{bmatrix} \xi^i_k \\ \xi^j_k \end{bmatrix} + \begin{bmatrix} B_i & 0 \\ 0 & B_j \end{bmatrix} \begin{bmatrix} u^i_k \\ u^j_k \end{bmatrix} \tag{9}$$

From the point of view of the MPC algorithm, the feasible region $\overline{P}(\xi^i)$ is non-convex. In order to reformulate the constraints in a linear form one has to use mixed integer techniques. By introducing n_c additional binary variables, $\delta^c \in \{0, 1\}$ one can write:

$$-H(\xi - \xi^i) \leq -K + M\delta^c, \quad c = 1 : n_c \tag{10}$$

This linear description gives a natural formulation for the constrains verification. If $\delta^c = 1$, the right-hand side of the inequality is negative and elementwise inferior to the left-hand side, for a sufficiently large user-defined scalar $M > 0$. From the optimization point of view the inequality is inactive in this case and trivially satisfied. If $\delta^c = 0$, the c-th inequality is activated. For collision avoidance, it is required that at least one of the halfspaces defining the constraints in (10) to be active which is translated by the additional constraint

$$\sum_{c=1}^{n_c} \delta^c \leq n_c - 1$$

Remark: Introducing a large number of constraints in (4) allows a better approximation of the safety region while increasing complexity of the problem by the increase of the number of binary variables. It is worthwhile to consider simple candidates for the safety region of the agents (hypercubes or simplices).

For simplicity reasons, in the rest of the paper we consider a rectangular shape for the region to be a fair choice in terms of precision and complexity (Fig.2):

$$P_d = \frac{d}{2} \mathcal{B}^\infty(x^i, y^i), \tag{11}$$

where $\mathcal{B}^\infty(x^i, y^i)$ is the ball with norm infinity centered in (x^i, y^i) and d is a constant which defines the size of the box.

Figure (2) illustrates that the avoidance constrains can be

written as:

$$\begin{aligned} x^j \geq x^i + d \quad \text{or} \quad x^j \leq x^i - d \quad \text{or} \\ y^j \geq y^i + d \quad \text{or} \quad y^j \leq y^i - d. \end{aligned} \tag{12}$$

To translate the avoidance constraints as the complement of the safety region (in order to avoid logical operands) in terms of linear constraints one has to introduce in (12) four binary variables δ^c, $c = 1, \ldots, 4$, leading to:

$$\begin{aligned} &x^i - x^j \leq -d + M\delta^1, \quad -x^i + x^j \leq -d + M\delta^2, \\ &y^i - y^j \leq -d + M\delta^3, \quad -y^i + y^j \leq -d + M\delta^4, \\ &\delta^1 + \delta^2 + \delta^3 + \delta^4 \leq 3. \end{aligned} \tag{13}$$

Thus a binary variable is associated to each inequality (12). Consequently, a large number of inequalities in the description of the safety region will enforce the use of a exceeding number of binary variables, which exponentially affects the complexity of the MPC problem.

We point here that a more compact representation which can be obtained using only two binary variables δ^c, $c = 1, 2$:

$$\begin{aligned} x^i - x^j &\leq -d + M(\delta^1 + \delta^2), \\ -x^i + x^j &\leq -d + M(1 - \delta^1 + \delta^2), \\ y^i - y^j &\leq -d + M(1 + \delta^1 - \delta^2), \\ -y^i + y^j &\leq -d + M(2 - \delta^1 - \delta^2). \end{aligned} \tag{14}$$

For any combination of the two binary variables a constraint from (12) will be activated. With respect to the original system (9) a compact form is the following:

$$\underbrace{\left[\begin{array}{cccc|cccc} 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \end{array}\right]}_{C_{ij}^p} \left[\begin{array}{c} \xi_k^i \\ \xi_k^j \end{array}\right] + \\ + \underbrace{\left[\begin{array}{cc} M & M \\ -M & M \\ M & -M \\ -M & -M \end{array}\right]}_{D_{ij}^p} \left[\begin{array}{c} \delta^1 \\ \delta^2 \end{array}\right] \leq \underbrace{\left[\begin{array}{c} -d \\ -d + M \\ -d + M \\ -d + 2M \end{array}\right]}_{\gamma_{ij}^p} \tag{15}$$

These constraints have to be used in the classical MPC framework with the values of $(C_{ij}^p, D_{ij}^p, \gamma_{ij}^p)$ as in (15). The auxiliary binary variables $\delta^c = [\delta^1, \delta^2]$ have to be considered in the optimization problem.

The rectangular region (11) is also considered to define the velocity constraints for an agent i:

$$\begin{aligned} v_x^i \leq -v_m, \quad \text{or} \quad -v_x^i \leq -v_m, \quad \text{or} \\ v_y^i \leq -v_m, \quad \text{or} \quad -v_y^i \leq -v_m, \end{aligned} \tag{16}$$

where the constant $v_m > 0$.

Similarly, the non-convex velocity constraints (17) can be rewritten using binary variables δ^c, $c = 1, 2$ and correspond in general terms to *fixed-obstacle avoidance constraints*:

$$\begin{aligned} v_x^i &\leq -v_m + M(\delta^1 + \delta^2), \\ -v_x^i &\leq -v_m + M(1 - \delta^1 + \delta^2), \\ v_y^i &\leq -v_m + M(1 + \delta^1 - \delta^2, \\ -v_y^i &\leq -v_m + M(2 - \delta^1 - \delta^2), \end{aligned} \tag{17}$$

For each agent, velocity constraints can be added in the classical MPC framework with the values of $(C_{ij}^v, D_{ij}^v, \gamma_{ij}^v)$, defined by (15). Globally, the non-convex state constraints (12), (17) are transposed in a compact form (with δ^p and δ^v the auxiliary binary variables introduced for position and velocity constraints reformulation):

$$\left[\begin{array}{c} C_{ij}^p \\ C_{ij}^v \end{array}\right] \left[\begin{array}{c} \xi^i \\ \xi^j \end{array}\right] + \left[\begin{array}{cc} D_{ij}^p & 0 \\ 0 & D_{ij}^v \end{array}\right] \left[\begin{array}{c} \delta^p \\ \delta^v \end{array}\right] \leq \left[\begin{array}{c} \gamma_{ij}^p \\ \gamma_{ij}^v \end{array}\right]. \tag{18}$$

IV. Optimization-Based Control & Path Following

This section presents the centralized MPC problem, where an optimization is performed to compute the control law for each agent. Based on the global prediction model, a flat trajectory is planned and, based on the information received from the real-time predictive control law, the avoidance constraints are taken into account. This leads the agents to achieve a formation while following the trajectory.

To define the MPC centralized problem, let us consider in a first step the global system defined as:

$$\dot{\tilde{\xi}}(t) = A_{g_c}\tilde{\xi}(t) + B_{g_c}\tilde{u}(t), \tag{19}$$

with the corresponding state and input vectors:
$\tilde{\xi} = [x^1\ y^1\ v_x^1\ v_y^1 | \cdots | x^{N_a}\ y^{N_a}\ v_x^{N_a}\ v_y^{N_a}]^T$,
$\tilde{u} = [u_x^1\ u_y^1 | \cdots | u_x^{N_a}\ u_y^{N_a}]^T$,
and the matrices which describe the model:
$A_{g_c} = diag[A_1, \cdots, A_{N_a}]$, $B_{g_c} = diag[B_1, \cdots, B_{N_a}]$.

The next step is to compare the measured state and input variables with the reference trajectory $(\xi^{ref}(t), u^{ref}(t))$ which satisfies the nominal dynamics:

$$\dot{\tilde{\xi}}^{ref}(t) = A_c\tilde{\xi}^{ref}(t) + B_c\tilde{u}^{ref}(t), \tag{20}$$

$\tilde{\xi}^{ref} = [\xi^{ref_1} | \cdots | \xi^{ref_{N_a}}]^T$, $\tilde{u}^{ref} = [u^{ref_1} | \cdots | u^{ref_{N_a}}]^T$. Then from (19), (20) the global system becomes:

$$\dot{\hat{\xi}}(t) = A_{g_c}\hat{\xi}(t) + B_{g_c}\hat{u}(t), \tag{21}$$

with $\hat{u}(t) = \tilde{u}(t) - \tilde{u}^{ref}(t)$, $\hat{\xi}(t) = \tilde{\xi}(t) - \tilde{\xi}^{ref}(t)$. The corresponding discrete time prediction model is the following:

$$\hat{\xi}_{k+1} = A_g\hat{\xi}_k + B_g\hat{u}_k, \tag{22}$$

with A_g, B_g the discrete form of A_{g_c}, B_{g_c} as described in Section II. Taking into account the constraints (18) and the fact that all the agents must follow the given reference trajectory, the centralized MPC problem is formulated as:

$$V_N(\hat{\xi}_k, \hat{u}_k) = \min_{u_k, \delta_{p_k}, \delta_{v_k}} V_N(\hat{\xi}_k, \hat{u}_k, \delta_{p_k}, \delta_{v_k}), \tag{23}$$

subject to

$$\begin{aligned} &\hat{\xi}_{k+l+1|k} = A_g\hat{\xi}_{k+l|k} + B_g\hat{u}_{k+l|k}, \quad l = 0 : N-1 \\ &\left[\begin{array}{c} C^p \\ C^v \end{array}\right] \hat{\xi}_{k+l|k} + \left[\begin{array}{cc} D^p & 0 \\ 0 & D^v \end{array}\right] \left[\begin{array}{c} \delta_{k+l|k}^p \\ \delta_{k+i|k}^v \end{array}\right] \leq \left[\begin{array}{c} \gamma^p \\ \gamma^v \end{array}\right] \end{aligned} \tag{24}$$

where V_N is the cost function (7) and A_g, B_g, C^p, C^v, D^p, D^v, γ^p, γ^v contain the centralized structural information of the multi-agent system. Therefore the centralized MPC

controller is acting on the global system (22) while offering the control inputs for each agent.

The simulation results show that the agents eventually form a certain structure while they follow the reference trajectory that we planned.

V. Simulation Results

This section proposes three simulation examples in order to better illustrate the proposed techniques. The system dynamics of each agent is given by (1) with the parameters $m = 60$kg, and $\mu = 20$Ns/m.

Example 1: Based on the model proposed in Section II, this example considers the tracking problem for a single agent. The generated trajectory is plotted in blue in Fig.3. The behavior of the agent over a time period of 400s, starting from an initial position $\xi = [15 \ -1 \ 1 \ -5]^T$ is depicted in red in the same figure.

Example 2: This example considers three agents following the same trajectory (in magenta, Fig.4) generated for the first example. The initial states for the agents are: $\xi^1 = [15 \ -1 \ 1 \ -5]^T$, $\xi^2 = [-7 \ 7 \ 5 \ 10]$, $\xi^3 = [8 \ 8 \ -5 \ 2]$. The parameters for the collision avoidance constraints (14) are: $d = 1$, $M = 100$. In Fig.4, the evolution of the agent formation is represented at three different time instants, all the agents are represented as filled circles and the safety region for each agent is represented as a square with $d = 1$. Each square points in the direction of each agent velocity vector. Good tracking performances for the given reference trajectory is obtained with a prediction horizon $N = 2$. In order to avoid the collision and according to the optimization result, the agents are self-organized (and can be assimilated with a flocking behavior) into the formation depicted in Fig.4.

Example 3: This example considers the case of three agents both with collision avoidance (14) and velocity (17) constraints (Fig.6) with $d = 1$, $v_m = 8$, $M > 0$. Fig.5 presents the simulation results for the initial states of each agent: $\xi^1 = [15 \ -1 \ 1 \ -5]^T$, $\xi^2 = [-7 \ 7 \ 5 \ 10]$, $\xi^3 = [8 \ 8 \ -5 \ 2]$. The agents are self-organizing in a triangle formation and the trajectory of its center of gravity is plotted in Fig.6 in magenta. Fig.5.c illustrates good tracking performances of the center of gravity (in blue), for a prediction horizon as low as $N = 2$. Increasing the prediction horizon leads to a better tracking of the reference trajectory. This impose a trade-off between complexity (increasing N) and precision of tracking the given path.

For different initial conditions and tunning parameters of the optimization problem, most simulations show that the agents have a regular motion while following the path in a specific formation. The state constraints are always satisfied. Although for some initial position of the agents, simulations have shown that appears a lack of synchronization while following the path in a line formation. Therefore the agents

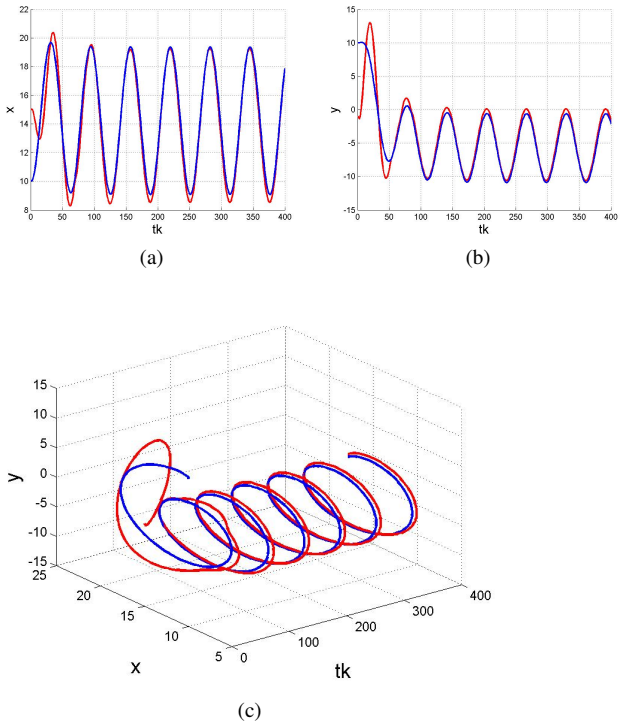


Fig. 3: The reference trajectory and the time evolution of one agent, along the path: (a) X-axis, (b) Y-axis, (c) X,Y-axes

formation proves to be sensitive to the alignment and this indicates from simulations that the triangle formation is less sensitive to numerical error along the real-time optimization. This deserves a detailed analysis when disturbances and noises affect the agents dynamics and represents one of the current research topics.

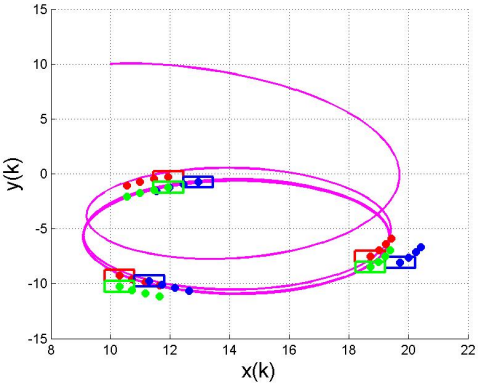
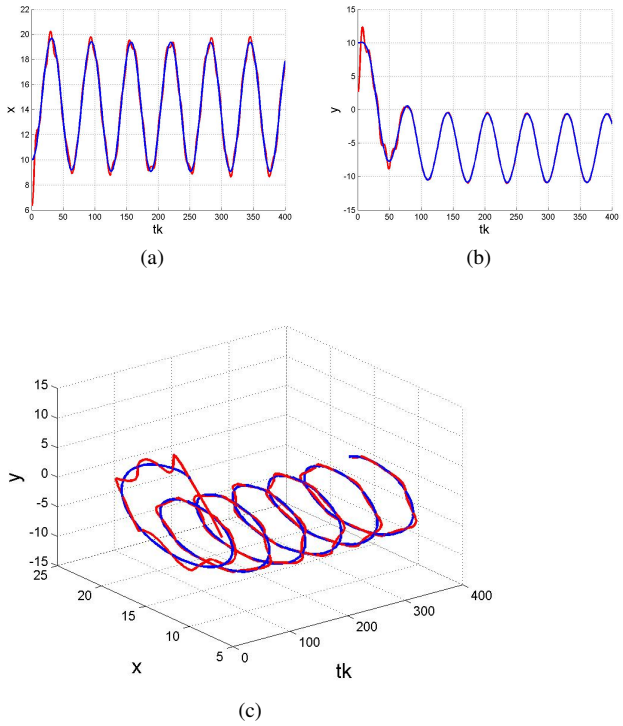


Fig. 4: Behavior of three agents in a triangle formation, with position constraints, at different time instances

VI. Conclusions

A centralized constrained MPC formulation for multiple agents that follow a given path, while satisfying collision avoidance and velocity constraints has been proposed in this



(a) (b) (c)

Fig. 5: The reference trajectory and the time evolution of the center of gravity of the triangle formation of 3 agents, along the path: (a) X-axis, (b) Y-axis, (c) X,Y-axes

paper. In the path following of multi-agent systems problems about collision avoidance, both within the group or with any obstacles always appear. The collision avoidance constraints describe a non-convex region, therefore a set of auxiliary binary variables are introduced in order to translate the non-convex state constraints into linear inequalities. In a similar way, constraints on velocity are also imposed and treated by adding other binary variables. The properties of differentially flat systems were used in order to obtain a reference trajectory for the group of agents. Therefore, based on the results provided by the constrained optimization problem, the agents organize themselves in a specific formation while they follow the given path. The results were presented through some illustrative simulations of several examples.

Depending on the size of the global system, a centralized MPC problem may be too large or may require a large computational effort. Therefore, future work will focus on investigating the case where the global system is decomposed in subsystems, leading to a distributed MPC formulation problem.

VII. Acknowledgements

The research of Ionela Prodan is supported by the EADS Foundation.

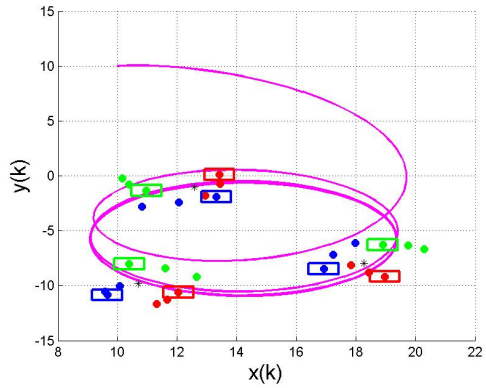


Fig. 6: Behavior of three agents in a triangle formation, with position and velocity constraints, at different time instances

References

[1] A. Richards and J. How, "Model predictive control of vehicle maneuvers with guaranteed completion time and robust feasibility," in *Proceedings of the American Control Conference*, vol. 5, pp. 4034–4040, 2005.

[2] M. Mesbahi and M. Egerstedt, "Graph-theoretic Methods in Multiagent Networks," 2010.

[3] V. Blondel, J. Hendrickx, A. Olshevsky, and J. Tsitsiklis, "Convergence in multiagent coordination, consensus, and flocking," in *44th IEEE Conference on Decision and Control and European Control Conference*, pp. 2996–3000, 2005.

[4] R. Olfati-Saber and R. Murray, "Distributed cooperative control of multiple vehicle formations using structural potential functions," in *IFAC World Congress*, pp. 346–352, 2002.

[5] D. Helbing, I. Farkas, and T. Vicsek, "Simulating dynamical features of escape panic," *Nature*, vol. 407, no. 6803, pp. 487–490, 2000.

[6] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Physical review E*, vol. 51, no. 5, pp. 4282–4286, 1995.

[7] Y. Tajima and T. Nagatani, "Scaling behavior of crowd flow outside a hall," *Physica A: Statistical Mechanics and its Applications*, vol. 292, no. 1-4, pp. 545–554, 2001.

[8] Z. Fang, W. Song, J. Zhang, and H. Wu, "Experiment and modeling of exit-selecting behaviors during a building evacuation," *Physica A: Statistical Mechanics and its Applications*, 2009.

[9] E. Camacho and C. Bordons, *Model predictive control*. Springer Verlag, 2004.

[10] J. Rossiter, "Model based predictive control: a practical approach, 2003."

[11] J. Maciejowski and M. Huzmezan, "Predictive control," *Robust Flight Control*, pp. 125–134.

[12] S. Qin and T. Badgwell, "An overview of industrial model predictive control technology," in *AIChE Symposium Series*, vol. 93, pp. 232–256, Citeseer, 1997.

[13] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, pp. 789–814, 2000.

[14] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, pp. 407–428, 1999.

[15] W. Dunbar and R. Murray, "Model predictive control of coordinated multi-vehicle formations," in *IEEE Conference on Decision and Control*, vol. 4, pp. 4631–4636, 2002.

[16] V. Manikonda, P. Arambel, M. Gopinathan, R. Mehra, F. Hadaegh, S. Inc, and M. Woburn, "A model predictive control-based approach for spacecraft formation keeping and attitude control," in *Proceedings of the American Control Conference*, vol. 6, 1999.

[17] M. Van Nieuwstadt and R. Murray, "Real-time trajectory generation for differentially flat systems," *International Journal of Robust and Nonlinear Control*, vol. 8, no. 11, pp. 995–1020, 1998.

[18] J. De Doná, F. Suryawan, M. Seron, and J. Lévine, "A flatness-based iterative method for reference trajectory generation in constrained NMPC," *Nonlinear Model Predictive Control*, pp. 325–333, Pavia, 2008.

Distributed Control Architecture with Intelligent Servo Drives for Sun Tracking Systems

D. Puiu*, F. Moldoveanu* and D. Floroian*
* *Transilvania* University of Brasov/Automation Department, Brasov, Romania
e-mail: puiudan@unitbv.ro, moldof@unitbv.ro, danf@unitbv.ro

***Abstract*—Nowadays the quick evolutions of manufacturing processes demand more flexible automation systems. These changes imply that the centralized motion control solutions should be replaced by the distributed control architectures based on intelligent servo drives. This paper presents the control of a sun tracking system with two local intelligent servo drives connected on a CAN network to a motion controller which has implemented a solar position algorithm. After the motion controller determines the position of the sun, it coordinates the two drives in order to correctly position the solar panel. This control structure can be extended and a single motion controller can coordinate more sun tracking systems.**

I. INTRODUCTION

The decreasing product and technology life cycles have made fixed automation systems cost prohibitive. As a result of that the automation users need flexible automation systems that can be modified upgraded and reused easily. This trend ensures that automation system have a long term competitive position [3], [8].

The solution for these problems is to replace the centralized control architecture based around a single host controller with a distributed control architecture where the tasks of the central controller are distributed among the local drives. In this way the local intelligent drives requires less computation power and they are cheap. Further more because each local intelligent drive has only a few tasks it is easier to debug it [3], [7].

Even if the intelligence is distributed through the control system, it has to work in real-time. This implies the implementation of a synchronization strategy of the nodes of the industrial communication network [1], [2]. To solve these problems it can be use an industrial communication network that has implemented in its protocol specific mechanisms for synchronizing its nodes such as CAN or PROFIBUS [7].

The sun tracking systems, in general allow solar panels to collect up to 50% more energy than that can be collected using stationary solar panels [4]. The majority of sun tracking systems moves the solar panel on two axis [10].

This paper presents the implementation of distributed control architecture for a two axis sun tracking system. The axes are actuated with two servos, which receive position commands from two local intelligent drives. The distributed control system is based on a CAN network where are connected the two local drives, a motion controller and a host computer.

This architecture reduces the cost of the control system because even if it uses more microcontrollers they are cheaper because they have fewer tasks to do. Further more, the control structure can be in any moment extended or reduced without complex hardware modifications.

II. DISTRIBUTED CONTROL ARCHITECTURES

The first industrial distributed systems are based on OSI standard for network protocols. It resulted a hierarchic architecture which has its components grouped on four layers (see Fig. 1).

Each component from the system receives commands from the devices from the superior layer and sends command to the devices from the inferior layers [3], [5].

The highest layer is the management one wihch ensures a graphic user interface; the downloading and uploading of programs and parameters; the monitoring of the stocks and analysis of the working times [3], [5].

The next layer is the control process one, where the execution times are a critical aspect. The intelligent units from this layer have the following main tasks: the generation of the trajectory and the coordination and the synchronization of the axis [3], [5]. Because the communication time is critical, the network that links this layer with the lower ones must ensure real time communication [6].

The local intelligent drives from the axis control layer receive motion trajectory for the actuator and executes it [3], [5].

This structure has a few disadvantages that make it hard to implement [7]:

- it has several communication protocols with increase the price of the automation and makes it hard to debug;
- from the top of the structure to the bottom and

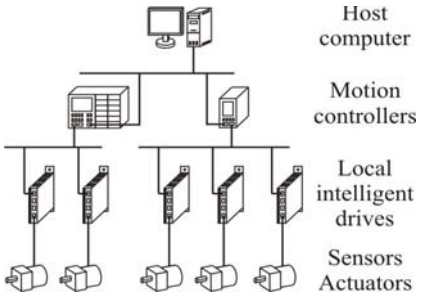


Figure 1. Distributed control system with hierarchic architecture [3].

reverse the data have to pass through several processor and the speed of transfer is decreased;

- the combination between the horizontal traffic with the vertical traffic is hard to be implemented for a real time multi-axis control.

An optimal solution to implement a distributed real time system is to connect, in a network with a single protocol, all the intelligent units from all the layers [7].

III. THE CONTROL STRUCTURE OF THE SUN TRACKING SYSTEM

The intelligence of the control structure is composed of three microcontroller developing boards Dice-Kit from Fujitsu, a Global Positioning System (GPS) module and a host computer. The boards have their own CAN controllers and the computer has a serial to CAN adapter.

Because the distributed control system with hierarchic architecture has a series of disadvantages the control structure of the sun tracking system will have a single CAN network. The structure of the system is presented in figure 2. The two axis of the sun tracking system are actuated with two servos, which have there own position and current control loops. The first servo actuates the azimuth angle and it receives its position reference from the Dice-Kit 2 board. The second servo is for the elevation angle and it is commanded by the Dice-Kit 3 board. These two Dice-Kit boards have the role of local intelligent servo drives.

The motion coordinator, it is also a Dice Kit development board from Fujitsu and it controls the two local intelligent servo drives via the CAN network. In addition to that, it is connected to a GPS module from which it receives the global coordinates of the positioning system and the local hour.

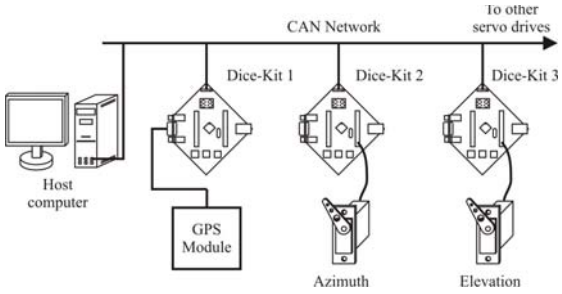


Figure 1. The control structure of the sun tracking system

A. The Motion Coordinator

As is presented in figure 2, the motion coordinator is connected to a GPS module which transmits information using the National Marine Electronics Association (NMEA) standard. The motion coordinator uses the Global Positioning System Fixed Data (GGA) message because it contains the global position of the module and the local hour.

Using the information from the GGA message, the motion coordinator calculates the azimuth and elevation angles. The algorithm used to determine the angles using the position of the module and the local hour was developed the National Renewable Energy Laboratory (NREL) from Colorado, USA and is presented in detail in the paper [9].

The motion coordinator receives at every second a GGA message, from the GPS module, and it recalculates the azimuth and elevation angles. If the motion coordinator needs to move the panel to a certain position it will have to transmit to the fist local intelligent servo drive the azimuth angle, respective elevation angle to the second one, and than in a broadcast message the time in which the motion should take place. The variation of this time determines the variation of the speed of the solar panel.

If the motion coordinator does not receive any message from the GPS module after a minute it sends an error message to the host computer.

B. The Local Intelligent Servo Drives

The two servos, that actuate the sun tracking system, have their own control loops, including the positioning loop, and can position their rotor under an angle of 180. Servos are composed of an electric motor mechanically linked to a potentiometer. Pulse-width modulation (PWM) signals sent to the servo are translated into position commands by electronics inside the servo. When the servo is commanded to rotate, the motor is powered until the potentiometer reaches the value corresponding to the commanded position.

When a local intelligent drive receives a new set of angles for the axis that it controls and the time in which the transition should take place, it will gradually change the reference of its servo in order to generate a linear move for the axis.

Because the frequency of the PWM signal for each motor is 50Hz it means that at every 20ms the local intelligent drive should update the position of the servo. The local intelligent servo drive calculates the immediate command with the following equation:

$$u_k = \frac{\left|u_{ki} - u_{kf}\right| \cdot \tau}{\tau_c} \quad (1)$$

where τ is the time that passed from the beginning of the transition; τ_c is the time in which the robot should effectuate the transition and u_k is the angle of the servo at τ moment. If the driver finishes the transition and it does not receive a new one it will maintain the last command.

The first local intelligent servo drive controls the azimuth angle from 90 to 270 degree (see figure 3). The sun tracking system must be orientated to south when the azimuth angle is 180 degree. The duty cycle of the command PWM signal generated by the local intelligent drive determine the position of the solar panel. When the duty cycle is 10% the solar panel is at 90 degree, respective 20% for 270 degree. After the first local intelligent driver calculates the azimuth angle (u_1) it will determine the equivalent duty cycle with the equation:

$$dc_1 = \frac{90 + u_1}{18}\% \quad (2)$$

where dc_1 is the duty cycle of the PWM signal generated by the first local intelligent drive.

The second local intelligent servo drive controls the elevation angle from 0 to 90 degree. In this case the duty cycle of the command PWM signal must be 15% for 0 degree, respective 10% for 90 degree and result the following equation:

$$dc_1 = \frac{90 + u_1}{18}\% \quad (3)$$

where dc_1 is the duty cycle of the PWM signal generated by the first local intelligent drive.

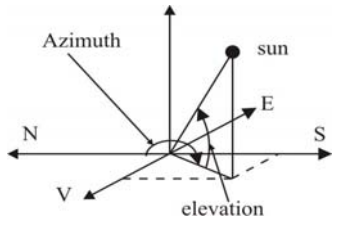


Figure 3 The zenith and elevation angles.

The second local intelligent servo drive controls the elevation angle from 0 to 90 degree. In this case the duty cycle of the command PWM signal must be 15% for 0 degree, respective 10% for 90 degree and result the following equation:

$$dc_2 = \frac{270 - u_2}{18}\% \quad (4)$$

where dc_2 is the duty cycle of the PWM signal generated by the second local intelligent drive.

After a local intelligent drive receives the broadcast message with the time, it has a break of 10ms, time in which they prepare for the transition and than starts the transition.

Because the local intelligent drives receive the time message in the same moment and they have the same type of microcontrollers; the same type of quartz oscillator and their programs are optimized to start the motion after 10 ms results that the moment when the two local drives start the transition is the same.

Further more, because the local intelligent drive respects the time received from the motion coordinator for the transition, it means that the movement of the two axes of the sun tracking system is a synchronized one.

C. The management of the CAN network

There are four main categories of messages on the CAN network:

- from the computer to the motion coordinator;
- from the motion coordinator to each local intelligent drive, message which contains the next position of the servos;
- from the motion coordinator to the all local intelligent servo drives, message which contains the motion time;
- status and error messages.

Because all the devices may need to transfer a piece of information at any moment of time, it is important to implement a table of priorities among the messages.

The message with the biggest priority is the broadcast message, with the time of the motion, from the motion coordinator to the local intelligent servo drives. The next message as priority is the message with the next position of the servo followed by the status and error messages.

The messages from the computer to the motion coordinator have the smallest priority because in this case there is no need for a deterministic communication.

D. 4.4 The software applications

The software of the local intelligent drives contains the command strategy of the servo, the synchronization algorithm and the communication protocol. It was designed and simulated in Softune, using a C compiler.

The software of the motion controller has also been developed in Softune and it includes the CAN and GPS communication protocols and the solar position algorithm.

The flexibility of the CAN network allows the designer to add on the network modules with different sensors (lights or temperature sensors) or even other control modules useful for the maintenance operations. In this case it is possible that motion controller to need a more powerful microcontroller or a digital signal processor. Because of this, all the software modules from the motion controller had been implemented in an ANSI C library, which can be included in any C program.

The library is useful for the programmer, because it contains functions for creating the commands and manages the message from/to the local intelligent drives and GPS module. The most important functions from the library are those who determine the position of the sun and the commands that should be send to the local intelligent servo drives in order to move the solar panel to the desired position.

IV. THE EXPERIMENTAL MODEL

The distributed control structure was tested with the hardware system presented in figure 4, which is from a LynxMotion articulated arm robot. Originally the robot had four degrees of freedom, but for this project it had been used only two of them.

The program for the host computer was designed in LabWindows/CVI and it manages the sun tracking system. From here the user can move the solar panel to a

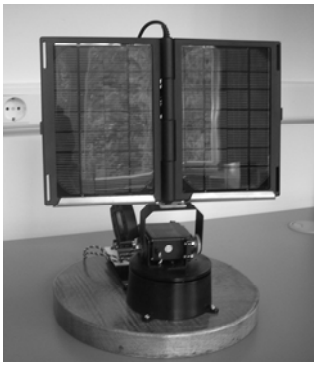


Figure 4 The hardware of the sun tracking system.

specific position for maintenance operations (for ex. cleaning the solar panel) or can monitor the position and the parameters of the sun tracking system.

The host computer can command only the motion controller because the control system maintained the specific of the distributed control system with hierarchic architecture where is no direct access from the host computer to the local intelligent servo drives (see figure 1). The host computer can only monitor the messages between the motion controller and local intelligent servo drives, advantage that is not possible in a hierarchic architecture control structure.

The interface of the program from host computer allows the user to send commands to the sun tracking system and to monitor the position of the solar panel.

Because the host computer has direct access to the messages transmitted by the motion coordinator, it can show the exact position of the solar panel, task that would have been very difficult in distributed control system with a hierarchic architecture.

V. CONCLUSIONS

The design of the project and the practical tests proved that the distributed architecture is a very effective one because it allows the designer to create and test small parts of the project and then put all of them together. Further more the complexity of the programs from the microcontrollers is smaller because each member of the control structure receives fewer tasks, rather than in centralized control systems where the central processing unit should do all the tasks.

In the case of the prototype presented in the experimental model section, because the sun trucking system is actuated with servos, the intelligent servo drives do not have the task of controlling the electrical motor it only has to generate the appropriate position reference for the servo. For a real sun trucking system the intelligent servo drive will have to control the electrical motor. Even in this case the structure presented is efficient because in case of a centralized control the microcontroller or the DSP has to solve the following tasks: the control of two electrical motors, the synchronization of the two actuators and the determination of the sun position. These tasks imply the use of a powerful microcontroller or DSP. But the recent advances of the microelectronics allow the cheap integration of the motor control and synchronization tasks in a single chip in the intelligent servo drive case, respective the sun position algorithm in the case of the motion controller.

Further more the design of the distributed control system allows that the same motion controller to coordinate several sun tracking systems.

An other fact proved by the hardware implementation is that the communication tasks, among the different modules, are easier because there is a single network and the messages do not have to pass through different processors.

In the next development stage the algorithm used to determine the angles azimuth and elevation will be moved from the motion controller into the local intelligent drives. In this way the motion controller will have to send only the information required by the position algorithm and the local intelligent drives will execute the command. This stage of development is important for the following step where the control system will be used for coordinating an array of mirrors that concentrate the light of the sun on a central tower, which converts the high temperature in steam and after that in electricity. In this case each mirror will be positioned by two local intelligent drives that will have their own algorithm depending on the position of the mirror in the field.

ACKNOWLEDGMENT

This paper is supported by the Sectoral Operational Programme Human Resources Development (SOP HRD), financed from the European Social Fund and by the Romanian Government under the contract number POSDRU/6/1.5/S/6.

REFERENCES

1. He, F., Tong, W., Wang, Q.: Synchronization Control Strategy of Multi-motor System Based on Profibus Network. In: IEEE Inter. Conf. on Automation and Logistics, pp. 3029--3034. Jinan (2007).
2. He, F., Zhao, S.: Research on Synchronous Control of Nodes in Distributed Network System. In: IEEE Inter. Conf. on Automation and Logistics, pp. 3029--3034. Qingdao (2008).
3. Jouve, D., Bui, D.: CANopen Servo Drives Provides High Performance Motion control. In: 38th Inter. Intelligent Motion Conf. – PCIM'02, pp. 1--6. Nuremberg (2002).
4. Khalil, A. A., El-Singaby, M.: Position Control of Sun Tracking Systems. In: 46th Inter. Midwest Symposium on Circuits and Systems – MWSCAS'03, pp. 1134--1137. Cairo, (2003).
5. Lee, K.C., Lee, S., Lee, M.H.: Worst Case Communication Delay of Real-Time Industrial Switched Ethernet With Multiple Levels. In: IEEE Trans. on Industrial Electronics, vol. 53, pp. 1669--1678 (2006).
6. Lin, S.Y., Ho, C.Y., Tzou, Y.Y.: Distributed Motion Control Using Real-Time Network Communication Techniques. In: 3rd Inter. Power Electronics and Motion Control Conf., pp. 843--847. Beijing, (2000).
7. Pacas, J. M.: Drives 2000 End of the Roadmap? – From the State-of-the Art to the Future Trends. In: 37th Inter. Intelligent Motion Conf. – PCIM'2000, pp. 41--50. Nuremberg (2000).
8. Pereira, C.E., Carro, L.: Distributed Real-time Embeded Systems: Recent Advances, Future Trends and Their Impact on Manufacturing Plant Control. In: Annual Reviews in Control, Vol. 31, pp. 81--92 (2007).
9. Reda, I., Afshin, A.: Solar Position Algorithm for Solar Radiation Applications. Technical Report, NREL, Colorado, TP-560-34302, (2008).
10. Taherbaneh, M., Ghafori, H., Rezaie, A. H., Karbasian, S.: Combination of Fuzzy-Based Maximum Power Point Tracker and Sun Tracker for Deployable Solar Panels in Photovoltaic Systems. In: IEEE Inter. Fuzzy Systems Conference – FUZZ-IEEE 2007, pp.1--6. London, (2007).

Electronic Equipment for Monitoring and Recording Analog and Digital Inputs

Dorina C. Purcaru, Anca I. Purcaru, Claudiu V. Rusu, and Marius N. Căpățînă

Abstract— **A distributed data acquisition system for process control divides its functions among data processing nodes. Such systems are organized on several levels. The PC-06/104 equipment for monitoring and recording analog and digital inputs is a functional node from the first level (signal acquisition); it was designed for industrial systems which perform the monitoring, recording and control of various parameters. PC-06/104 performs data processing and local storage, sending the resulted information to the higher level (process server) of the distributed data acquisition system. The paper presents the hardware architecture and features of this digital equipment organized around a PC/104 central unit. This equipment has four main functions: acquisition, configuration, local monitoring and communication. PC-06/104 is already used in several hydropower stations from Romania. Some experimental results, corresponding to a server and client application, are also presented and discussed in this paper.**

I. Introduction

THE distributed data acquisition systems for process control are open structures that enable the distribution of all functions in functional nodes. In such informatics systems, the functional nodes for signal acquisition are very important and their hardware design is focused on a high independency degree that is necessary for both the informatics system development and the further substitution of some functional nodes.

Catastrophic failures of power systems occur infrequently, but they cause great trouble to the industrial companies. In recent years, it has become evident that precise measurements of power systems' state in real time is a very important tool for managing the operation of power systems, as well as mitigating some of the effects of catastrophic failures. Many techniques have been developed to make the power system survives during disturbances and continue to operate. One recent developed technique that may be used is Synchronized Phasor Measurement; reference [1] presents a PC based phasor measurement prototype and its applications in power systems for improving protection system reliability. Reference [2] describes the distribution network monitoring methodology based on the mathematical model of load; an application of network analysis, modeling and estimation but also validation of results with real measurements obtained from the distribution network is presented too.

The power quality aspects are very actual and important in the last years. The impact of voltage imbalance on power distribution systems and equipments in various practical cases is investigated in [3]. The fluctuation voltage detection by using recursive DFT for voltage compensation control is presented in [4]. A new data acquisition system for power quality measure in electrical networks based on PC platform and modularized software packages, which will optimize the power grid operating mode for network optimization, technical energy losses reduction and customer satisfaction is presented in [5].

Several programmable electronic modules for data acquisition, designed for energetic systems, are presented in [6]–[9]; they are placed into different locations, near the input signals provided by the observed process. All these equipments are functional nodes which form the first level of the distributed data acquisition system and they perform the process control or monitoring. For example, the following programmable electronic modules (from the PC-XX family) run in transformer plants, electric power stations or micro-hydroelectric power plants in Romania :

- PC-01/104 - Device for storage batteries control, which performs the parameter monitoring and charge/boost-discharge cycle control of the storage batteries from transformer plants;
- PC-05/104 - Device for the implementation of various complex protection functions for hydrogenerators;
- PC-06/104 - Electronic equipment for monitoring and recording analog and digital inputs, presented in this paper;
- PC-08/104 - Disturbance monitoring device which performs the monitoring and recording some specific high speed transient event parameters from energetic systems, for after-failure analysis and diagnostic.

The RS-485 bus allows the communications between these programmable electronic modules for data acquisition (first level) and a process server (the second level of the distributed data acquisition system); the last one is a data processing and monitoring equipment that also represents the process server in a computer network with several clients. Each data acquisition module from the first level sends the requested data (provided by the process) to the

Manuscript received June 4, 2010.

D. C. Purcaru is with the University of Craiova, Faculty of Automation, Computers and Electronics, Department of Electronics and Instrumentation, 200-440 Craiova, Romania (phone and fax: +40-0251-438198; e-mail: dpurcaru@electronics.ucv.ro).

A. I. Purcaru is with VIG Impex, 200-233 Craiova, Romania (e-mail: aipurcaru@gmail.com).

C. V. Rusu is with VIG Impex, 200-233 Craiova, Romania.

M. N. Căpățînă is with VIG Impex, 200-233 Craiova, Romania.

process server; this data is then accessible for any client from the computer network.

This paper presents some design considerations and software aspects regarding the PC-06/104 equipment for monitoring and recording analog and digital inputs; this equipment already runs in several hydropower stations from Romania (Hidroelectrica, Transelectrica) and has proven to be very useful in systems for monitoring and recording industrial parameters.

PC-06/104 has obvious advantages, compared to the data acquisition systems presented in [5]: it performs more functions, allows the acquisition of more analog and digital voltages, is more versatile being also useful for other industrial applications, it is cheaper and allows future hardware and software developments.

Due to its hardware architecture based on actual embedded system, the GPS receiver and also its implemented functions, the electronic equipment presented in this paper is more performant than the equipment presented in [7] which can be considered obsolete.

II. Functions and Design Considerations of the Equipment for Monitoring and Recording Analog and Digital Inputs

The four main functions of the PC-06/104 equipment for monitoring and recording analog and digital inputs are the following:

- Acquisition (signal conditioning, analog to digital conversion, data storage in local UC memory);
- Configuration (setting the equipment parameters);
- Local monitoring (observing the process parameters);
- Communication (taking over the data from the local memory and then sending it to other equipment).

The equipment can be easily configured as intelligent functional node in a distributed data acquisition system. PC-06/104 performs the process monitoring according to the signals provided by the transducers that take over the process parameters. This electronic equipment has a serial connection to the PC-07/104 process server which runs at the higher level of the data acquisition system; PC-07/104 is an IMP-PC/AT compatible equipment for process monitoring and remote control. An archive can also be generated for the off-line analysis of the monitored parameters.

PC-06/104 allows the acquisition of analog inputs (voltages) and digital inputs (voltage levels), with programmable pacer clock for initiating repeated conversions. PC-06/104 performs the data processing and local storage and the resulted information can be then sent to the higher level of the distributed data acquisition system.

The *recording* function consists in storing the acquired samples in the nonvolatile memory of PC-06/104 for further post-failure analysis of the monitored equipments.

PC-06/104 is not only a simple implementation of a data acquisition system around a PC/104 central unit, as

- only two electronic modules (PC/104 CPU and GPS receiver) of PC-06/104 are acquired by purchase; the others are designed, made and assembled by the authors,
- the equipment's software is also implemented by the authors.

A. Hardware Architecture

The architecture (standard configuration) of the equipment for monitoring and recording analog and digital inputs is shown in Fig. 1.

The analog inputs (voltages) are configurable (4, 8, 12 or 16 channels), and the digital inputs are also configurable, as number of channels (8, 16 or 24) and voltage level. The signal conditioning modules (Digital IN, Analog IN in Fig. 1) contains impedance adapters, voltage-level adapter circuits and a galvanic separation between each analog or digital input channel and PC-06/104.

For the real time clock synchronization, this electronic equipment receives information from the low cost GPS internal receiver ET-332 GPS engine board [10], [11] and maintains high reliability and accuracy, making it an ideal choice for integration with OEM/ODM systems [12].

The local MMI (Man Machine Interface) consists in a flat keyboard (with 4×2 keys) and an alphanumeric LCD (with 2×16 characters). There are also five LEDs that allow the local signaling of the equipment status. The RS-232 interface is used in local monitoring, configuration and parameter setting of this data acquisition module; the Local Console from Fig. 2 enables these functions exploitation.

PC-06/104 contains a FB-232 compatible interface to communicate with the PC-07/104 process server; together, the two equipments allow various industrial applications implementation. The field bus protocol is serial (Modbus RTU), and it is implemented with RS-485; the baud rate is 57.600.

B. PC/104 CPU Module

The VSX-6154 family of low-power x86 embedded controller is designed to meet PC/104 specifications [13] and integrated with the features presented in [14].

VSX-6154 is a PC/104 CPU-Board that offers high reliability and performance at low power consumption. This CPU is suitable for broad range of data-acquisition, industrial automation, process control, automotive controller, intelligent vehicle management device, medical device, human machine interface, robotics, machinery control and other applications that require small footprint, low-power and low-cost hardware with open industry standard such as PC/104.

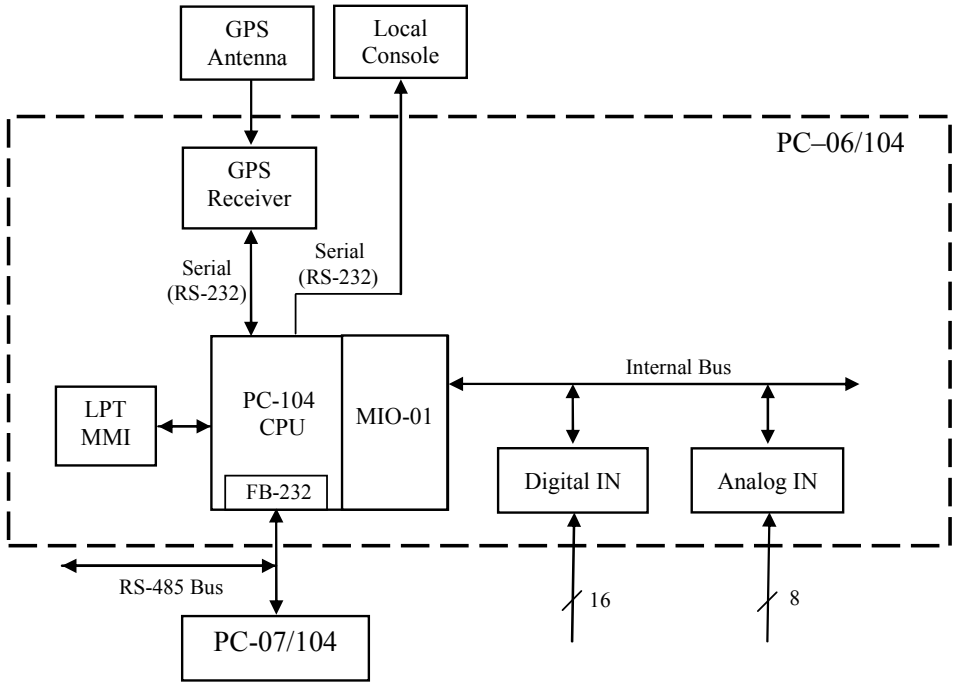


Fig. 1. Hardware architecture of the PC-06/104 equipment.

C. MIO-01 Multifunctional Interface for Data Acquisition

The MIO-01 electronic module is electrically and mechanically PC/104 compatible; it plugs directly onto the PC/104 CPU board, turning it into a high performance data acquisition and control system.

This multifunctional interface performs analog, digital and timing I/O functions. There are maximum 16 common-mode analog inputs (voltages, between -5V and +5V) and maximum 24 digital inputs/outputs (all TTL compatible) which allow MIO-01 to interface with a variety of signals.

For timing functions, this multifunctional interface uses 8254 (Programmable Interval Timer); it contains three programmable counter/timer channels with 4 or 10MHz the clock frequency. The A/D converter has a 16 bits resolution and a maximum conversion time of 10μs. MIO-01 has a settable base address.

The 8254 programmable timer [15] requests an interrupt with a settable level to the PC/104 processor so that the A/D conversion could be performed and the data could be then stored into defined memory.

III. SOFTWARE DESIGN AND EXPERIMENTAL RESULTS

There are three levels of software design:

a) *Data Acquisition* application, associated with the PC-06/104 equipment;
b) *Server* application, associated with the PC-07/104 equipment;
c) *Client* application, associated with an IBM PC-AT compatible computer which is a client from the computer network.

The *Server* application runs on the PC-07/104 process server, takes over the process parameter from a serial bus (using a RS-485 compatible protocol organized in communication packages), and performs a local data processing, providing the data to several clients (in a computer network).

Fig. 2 illustrates the window of *Server* application. The main function of this application is the acquisition of several input values (provided by PC-06/104) which compose a *recording*.

There are three buttons and two edit boxes on the left side of Fig. 2, having the following meanings:

- *Recording length (in seconds)* – edit box for recording length setting (this value is 3s in Fig. 2);
- *Sampling period (in milliseconds)* – edit box for each input sampling period setting (this value is 1ms in Fig. 2);
- *Recording start* – recording start button; all analog and digital inputs values are acquired during the recording length by the PC-06/104 electronic module, using the same sampling rate for each input;
- *Recording transfer* – allows the recording transfer on the PC-07/104 process server;
- *Open the recording file for drawing* – displays the loaded recording, on the color LCD of the PC-07/104 process server.

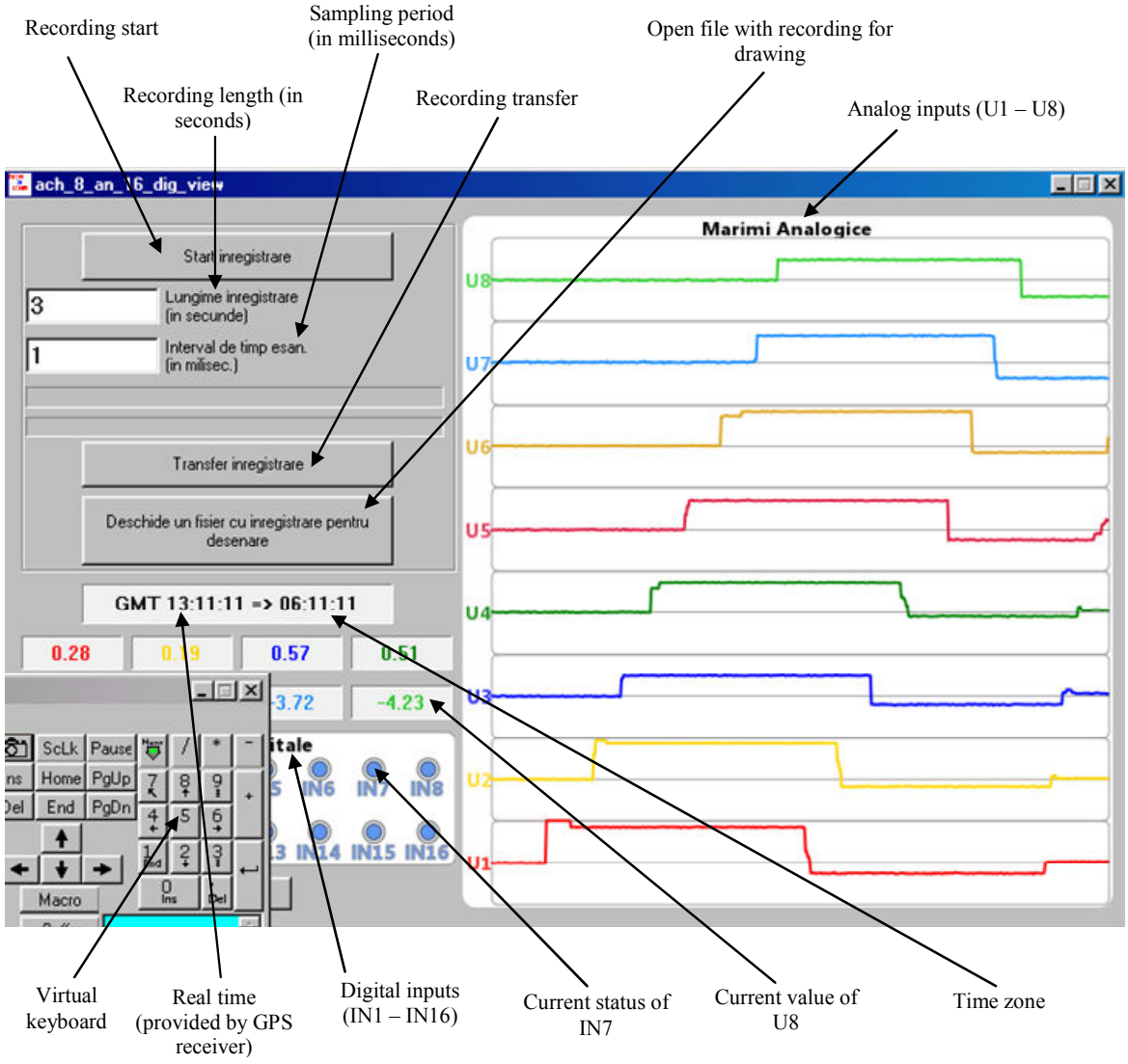


Fig. 2. The main window of the *Server* application.

The virtual keyboard, running on the server touch screen, allows the parameter setting of the recording.

The graphic of all analog inputs (U1, U2,…,U8) versus time (during a settable period) can be seen on the right side of Fig. 2 where the current time moment is the y-coordinate. Different colors are used for analog inputs. The current values of all analog inputs are displayed, with their corresponding colors, in the left side of the figure.

The current status of all digital inputs (IN1–IN16) can also be seen in Fig. 2; red signifies high level, and blue signifies low level.

The real time (provided by the GPS receiver and Bucharest (GMT+02:00) time zone) are also displayed.

The *Client* application is implemented mainly for testing the functionality of the following ensemble: data acquisition equipment (PC-06/104) – RS-485 protocol – process server (PC-07/104) – TC/IP (LAN) protocol – client.

The graphical user interface of this application is presented in Fig. 3. There are three ways of displaying the eight analog inputs: alphanumeric, graphic and bar graph. The graphic of all analog inputs (U1, U2,…,U8) versus time can be seen on the right side of Fig. 3 where the current time moment is the y-coordinate. Different colors are used for analog inputs. The current values of all analog inputs are displayed, with their corresponding colors, in the left side of the figure. The current status of all digital inputs (IN1, IN2,…, IN16) can also be seen in Fig. 3; each digital input changes the color when it changes its status.

The main scientific and technical contributions of the authors are shown below:

- The implementation, on this equipment, of a field bus protocol Modbus RTU using an own RS-485 interface controlled by the host processor; a low-cost interface is in this way obtained;

- The implementation of an own acquisition and communication software;
- The implementation and manufacturing of a low-cost data acquisition interface (MIO-01) PC/104 compatible;
- The software development for receiving and processing data provided by GPS Receiver;
- The development of a monitoring and control application for micro-hydroelectric power plants.

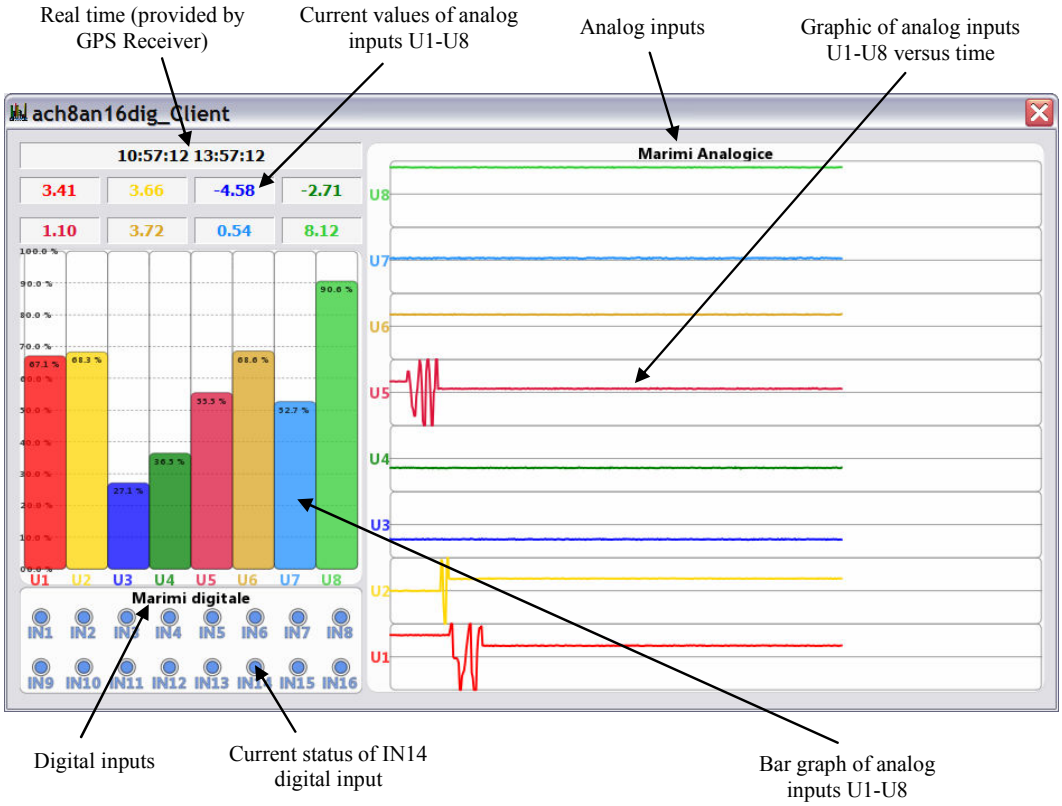


Fig. 3. The main window of the *Client* application.

IV. Conclusion

This paper presents an electronic equipment which has proven to be extremely useful in monitoring and recording parameters for industrial systems. Its main advantages are the following:

- PC-06/104 is a specific equipment;
- its hardware architecture and software can be adapted to proper requests of various industrial applications;
- a low-cost implementation of industrial monitoring applications is possible;
- this designed configuration (based on PC/104 standard) allows hardware and software future developments.

Further work is focused on the storage of current values of all analog and digital inputs, during a settable period, using a circular buffer; the resulted recording can be uploaded on the server or one client, as the user can then list the recording. In this way, The PC-07/104 can be considered a *recorder without paper*.

References

[1] H. Bentarzi, "Improving monitoring, control and protection of power grid using wide area synchro-phasor measurements", in *Proc.* of *The 12th WSEAS International Conference on Automatic Control, Modelling & Simulation,* Catania (Italy), 2010, pp.93–98.

[2] J. Kilter and M. Meldorf, "Monitoring of the electrical distribution network operation", in *Proc.* of *The 4th IASME / WSEAS International Conference on Energy & Environment,* Cambridge (UK), 2009, pp.253–256.

[3] T-H Chen, C-H Yang, and T-Y Hsieh, "Case studies of the impact of voltage imbalance on power distribution systems and equipment", in *Proc.* of *The 8th WSEAS International Conference on Applied Computer and Applied Computational Science,* Hangzhou (China), 2009, pp.461–465.

[4] W. Srisongkram, K. Bhumkittipich, N. Phanthuna, P. Unahalekhaka, and T. Trongtirakul, "Fluctuation voltage detection by recursive DFT for voltage compensation control in power system", in *Proc.* of *The 4th WSEAS International Conference on Circuits, Systems, Signal and Telecommunications,* Cambridge (USA), 2010, pp.140–143.

[5] S. D. Grigorescu, C. Cepisca, N. Jula, and N. Popoviciu, "Data acquisition system for energy quality monitoring in decision-making network nodes", in *Proc.* of *The 5th IASME WSEAS International Conference on Energy & Environment,* Cambridge (UK), 2010, pp. 101–105.

[6] I. Purcaru, D. Purcaru, C. Gordan, and M. Niculescu, "Digital equipment for the implementation of complex protection functions", *Journal of Electrical and Electronics Engineering*, Volume 3, Number 1, pp. 171–174, May 2010.

[7] S. Iordache, M. Nedelcu, I. Purcaru, and V. Tapardea, "Sistème de surveillance des grandeurs électriques et dépistage des défauts dans une station électriques", in *Proc. (Section Computer Science & Engineering)* of *International Symposium on Systems Theory, Robotics, Computers & Process Informatics,* Craiova (Romania), 1996, pp.105–112.

[8] I. Purcaru, D. Purcaru, "Aspects regarding the configuration of a functional node in a distributed data acquisition system, based on PC/104 compatible modules", Solid State Phenomena Vols. 166-167, pp. 327–332, July 2010.

[9] D. Purcaru, I. Purcaru, C. Gordan, E. Niculescu, and S.D. Nedelcuț, "Application of Fourier method in energetics, for root-mean square value and phase shift measurement", in *Proc. (Tome III)* of *IEEE International Conference on Automation, Quality and Testing, Robotics,* Cluj-Napoca (Romania), 2008, pp.187–192.

[10] NMEA Reference Manual, November 2008. Available: http://www.usglobalsat.com/downloads/NMEA_commands.pdf

[11] SiRF Binary Protocol. Reference Manual, November 2008. Available: http://www.usglobalsat.com/downloads/SiRF_Binary_Protocol.pdf

[12] C. Pancu, A. Baraboi, M. Adam, and A. Plesca: "GSM Based Solution for Monitoring and Diagnostic of Electrical Equipment", in Proc. of *The 13th WSEAS International Conference on Circuits,* Rodos (Greece), 2009, pp.58–63.

[13] *aJ-PC104 Datasheet,* April 29th 2000. Available: http://www.ajile.com/aJ-pc104_datasheet.htm.

[14] VSX-6154. DM&P Vortex86SX 300MHz PC/104 CPU Module with 4S/2USB/VGA/LCD/LAN/GPIO 128MB DDR2 Onboard. User's Manual, December 2007. Available: http://www.wdlsystems.com/downloads/manuals/1EVX61E_m.pdf

[15] Intersil, *82C54 Datasheet,* 2003.

Some Trends in Computer Graphics

Werner Purgathofer

***Abstract*— In this paper we give an overview of some current research trends and explore the challenges in some subfields of the scientific discipline of computer graphics: interactive and photorealistic rendering, visualization and visual analytics. Five challenges are extracted that play a role in each of these areas: scalability, semantics, fusion, interaction, acquisition. Of course, not all of these issues are disjunct to each other, however the chosen structure allows for an easy to follow overview of concrete future challenges.**

I. Introduction

COMPUTER graphics studies methods for producing digital images of data with the goal to communicate computer output to a human user in the form of pictures. The visual input channel has by far the broadest bandwidth of all our senses and therefore enables the most effective transport of information from computers to humans. Production includes synthesizing, manipulating and displaying the underlying data. The data may be almost any content one can think of: geometric or other spatial data just as well as statistical data, simulation results or abstract data, all real or virtual. Roughly speaking, the ultimate goal of rendering research is to create perfectly realistic looking real-time images of real-world objects, whereas visualization tries to create images of data and structures that are otherwise invisible to the human eye or completely abstract.

Computer graphics has been among the most successful computer science fields during the last three centuries and the methods and results available today have exceeded the expectations by far. Therefore some people consider most computer graphics problems as solved, providing a ready to use set of tools for applications. But while this is true for some areas with simple use of computer images, the embedding of computer graphics technology in increasingly complex surroundings generates many new challenges.

Usage of computer graphics is embedded in more and more complicated environments, making its combined use with other technologies more and more natural, so that many people talk about disciplines growing together. Such fields are computer vision, image processing, pattern recognition, tracking, scanning, video augmentation, information theory, user interface design, large data bases and several more. Multiple articles in the past decades have extracted future research problems in various subfields of computer graphics, e.g. [1, 2, 3]. In this article we will describe the research trends of the coming years based on five major challenges that are common to all computer graphics sub-fields. These are:

1. *Scalability* = how to cope with huge amounts of data, highly parallel computers and distributed devices.
2. *Semantics* = how can meaning be extracted from data and context and be used for better insight.
3. *Fusion* = how can multiple techniques, data streams, and models be combined to solve complex problems.
4. *Interaction* = how to combine multiple and ubiquitous input devices to create ergonomic user interfaces.
5. *Acquisition* = how can data from various input sources be processed to deal with missing data, contradictions, and uncertainty.

These challenges are often interdependent to some degree. For example, semantics can assist scalability by determining features that can be left away at a given level of detail, and acquisition by allowing a meaningful extrapolation of missing data and resolution of contradictions; petascale data will require new interaction metaphors and acquisition and data processing methods. Consequently, any research in these areas will typically cross boundaries and cannot be strictly attributed to a single challenge. Still, this discrimination facilitates the following description of the specific aspects of each challenge.

II. Scalability Challenges

The challenges posed by the enormous amount of data generated by current and especially future acquisition techniques will require fundamental research on scalable algorithms, techniques, and systems. However, many existing methods are designed for a relatively narrow range of data sizes and characteristics, and are not directly applicable to the enormous requirements of petascale computer graphics.

In addition to handling and visualizing petascale data, computations such as 3D reconstruction, segmentation, object identification, and the generation of derived data must also be able to operate on this new order of magnitude. Possible solutions lie in the parallelization and distribution of computation and visualization, exploiting all levels of non-uniform architectures. The possibilities offered by the

architectural levels of multi-core CPUs, GPUs, shared and distributed memory architectures, clusters for computation and visualization, and remote visualization and computation must be exploited in a coherent and scalable manner. This also necessitates aggressive multi-resolution approaches that work on a huge range of scales, while preserving important features and considering application and user semantics.

The overall challenge of future scalable systems ties in with the requirements of semantic interfaces and navigation through the enormous spread of scales in petascale data. The underlying algorithms must be scalable to different kinds and different levels of expertise of users, such as their specific knowledge about a given problem domain. This requires techniques to be able to supply quick but accurate overviews and deliver additional detail on demand, while preserving features of interest over a huge range of scales. With petascale data, millions of data elements can map to the same screen pixel of the output device. Handling this scalability challenge in a meaningful way that preserves application or user semantics, and enables users to actually work with data on this order of magnitude is one of the most crucial and fundamental scalability issues.

The development of scalable visualization algorithms and systems is one of the most fundamental future challenges in visualization. The main reasons for this are the enormous increase in data sizes that are routinely generated by high-resolution acquisition technologies such as Electron Microscopy; the variety of different data sources that need to be visualized and analyzed concurrently; and the amount of additional derived data that often need to be computed from the raw input data. Therefore, the main topics of research on scalability in this area are:

- Distributed and out-of-core visualization
- Approaches that exploit massively parallel CPU and GPU architectures
- Progressive, feature-preserving multi-resolution techniques
- Achieving scalability for users by integrating semantics (see also challenge semantics)

Due to the extremely large amounts of geometry that are provided for rendering in many projects, scalability with respect to the amount of data is one of the most urgent needs in rendering. In view of the ongoing trend towards mobile devices, it will also become increasingly important to support a wide variety of output devices, ranging from hand-held devices with small screen sizes, all the way to video-walls consisting of multiple monitors or projectors.

III. SEMANTICS CHALLENGES

In addition to the algorithmic challenge of handling the huge amounts of data that will have to be processed in future applications, the interpretation and analysis of this data will require additional semantic information. If all of these data sets are enriched with semantic information, it will be possible to formulate intelligent queries that retrieve subsets of the data that meet certain semantic criteria. Further processing of the data will be able to choose appropriate techniques based on semantic information. Semantic information can be based on the underlying data, but also on the analysis goal, the application scenario, use history or the user profile.

Visualization and rendering of such semantically enriched data sets can use this information to provide different visualization methods and views of the data based on the context: analysis of the data requires different strategies compared to presentation for larger audiences. Semantic information can also be used to compress huge data sets and reduce transmission costs in distributed setups. Given these benefits that can be realized if semantic information is available, the challenge for the field of computer graphics is three-fold: to research and develop the appropriate methods to extract semantic information from huge, heterogeneous unstructured data sets. Based on the area of application a number of different techniques such as atlases, refined matching methods and codification and sharing of insight will have to be used; to find appropriate data structures for semantic information that make it possible to refine and enhance the knowledge base as additional data becomes available; and finally to extend existing methods to make optimal use of semantic information in rendering and visualization. This requires novel, tightly integrated display techniques combining scientific and information visualization methods.

Today users of visualization systems are required to have detailed knowledge at a technical level about visualization and the data to explore and analyze. In the future, semantic visualization will put users in the center of visualization systems. This will be facilitated by using application domain specific conventions and offering semantic user interfaces as well as interaction techniques in the application domain instead of the data domain.

Semantic visualization is the most important step towards making visualization tools a part of the daily routine of domain experts. This requires the development of appropriate abstractions to provide an immediate overview as well as additional details and more time consuming interaction techniques on demand. The possibility to manipulate visualization parameters on a more technical level will be hidden, but will remain available to expert users. Therefore, the main topics of research on semantics in this area are:

- Knowledge-assisted visualization
- Knowledge-based navigation
- Integration of semantics with segmentation and feature-detection

In rendering, semantics apply in two different ways. Semantics are important to the user to make correct contextual decisions, a challenge that is closely coupled to visual analytics. However, semantics can also be used as an internal representation within the rendering system, independent of the rendering methods or usage scenario. As such, semantics are a highly compressed abstraction of the represented objects, and capture only the distinguishing features of a specific instance. In an ultimate semantically based rendering system, it would therefore be possible to specify each type of object with just those parameters that are necessary to create a highly realistic, detailed rendering of the object. If an object is an instance of a group of similar objects only a few parameters may be sufficient to completely specify the object. This goes further than purely procedural rendering, since in this case the actual methods (the procedures) are specified separate from the semantic models. A large part of the know-how of such a semantic rendering system is encoded in the rules that describe classes of objects.

IV. FUSION CHALLENGES

Fusion is a challenge with several interrelated aspects: the fusion of multiple fields of computer graphics, the fusion of computer graphics with other fields of computing, and the fusion of multiple data sources. Fusing multiple fields of computer graphics is essential for a holistic analysis of data. In many cases, no single display method is adequate for all aspects of a complex dataset. To effectively cope with the challenge of fusing multiple display methods is one of the future research topics.

In many applications, computer graphics is only used as static post-processing or for presenting the results of long and non-interactive computations. This may cause significant delays, as adjustments due to potential errors or improvements require expensive iterations. It is thus important to strive for human-centric, integrated approaches, which tightly combine interactive rendering and visualization with computational methods.

The third kind of fusion concerns the integration of multiple data sources in the visual analysis of data. Different methods for measuring data have typically different advantages and disadvantages. It is therefore a challenge to improve the ultimate result and maximize the gained insight by simultaneously processing and displaying related data from multiple sources. This applies to many application domains, among them medical data, data from scanning geometry, and from scanning motion. Another important issue is a unified analysis of both measured and simulated data, as required for meteorology.

The overall challenge posed to visualization is increased significantly by a variety of data sources with widely varying characteristics. Important examples are different imaging modalities, sensor types, or data computed in simulations; different representations such as structured or unstructured grids, point clouds, or geometry; or data of different dimensionality such as scalar-, vector-, or tensor fields. Moreover, data on different conceptual levels such as raw, processed or annotated data need to be integrated effectively. The goal of fusion research in this area is to aid in understanding and reasoning about such data through visual condensation and fusion of the available information.

Due to the growing heterogeneity of the data involving both spatial and non-spatial information, it has also become necessary to enrich 3D real-time rendering with overlays, and to consider spatial semantics in multivariate visualizations. Besides fusing display methods, a core topic of visual analytics is to integrate automatic approaches in the process of analyzing data. In this context, the ultimate goal is typically more concrete than just “gaining insight”, and often involves specifying / evaluating / optimizing a model as knowledge representation.

V. INTERACTION CHALLENGES

An interactive environment, where the user can explore and manipulate data in real-time in an effective and intuitive way is a powerful tool for many areas of application. Providing such an environment is a challenge in many respects. Emerging interface technologies like face, gesture, speech recognition, multi-touch displays, optical tracking, eye-tracking, even EEG based input, and the proliferation of ubiquitous systems, bringing computing into the user’s environment, call for innovative ways of supporting Human Computer Interaction (HCI). Non-classical interface techniques are already being adopted by the gaming industry and others.

The challenge will be to develop, adapt and evaluate such non-classical interface techniques as virtual environments, tangible user interfaces or vision based interaction so that they become effective and meaningful interaction tools for the respective user, task and device at hand. Depending on the intended target audience the level of interaction with the environment often needs to be adapted according to the needs of the user. Balancing user assisted, user guided/context aware and automatic approaches to achieve the appropriate level of interaction will be a challenge concerning both the evaluation of the users’ needs and, for many tasks, developing an automatic or guided, context

aware approach on its own. Along with the increasing pervasiveness of distributed data/systems the focus is shifting from individual users to small and large-scale interactions for groups of possibly highly mobile users. In such multi-user environments, the challenge will be the fusion of interaction with the environment and the other users to effectively support local and remote collaboration.

A successful visualization is not a collection of static images we can simply browse. A good visualization needs interaction to support the reasoning process. Interaction plays a major role in most visualization approaches and the design of efficient interaction is often crucial for the success of a visualization method.

In rendering and virtual reality, the interaction challenge appears mainly in the context of human-computer interaction, ie. the challenge of finding suitable input and display devices for a given application, and in collaborative interaction of multiple simultaneous users. In all rendering applications that are targeted at real-time or interactive usage scenarios, proper choice of interaction and display devices naturally plays an important role. The ideal interaction metaphor may vary widely with the actual application, and may range from traditional keyboard and mouse to more complex devices such as the SpaceMouse and VR interfaces. In addition to the input devices to be used, this challenge also includes the design and placement of user interface elements within a given application, a non-trivial issue.

VI. ACQUISITION CHALLENGES

Today Visual Computing focuses on the display and analysis of real world data gathered by an array of diverse measurement techniques. While former rendering methods concentrated on simulating complexity by using textures, approximated illumination, and simplified modeling of complex structures, nowadays we face the challenge of visualizing data gathered by data acquisition systems. While manually modeled data normally lacks detail and internal consistency, acquired data typically suffers from measurement errors, noise, dropouts, repetition, and lack of semantic information. Typical acquisition areas include: architectural data, laser scans, photogrammetric data, medical and industrial data, MRIs, X-ray, ultrasound, real-time acquisition of position and geometry from depth images, GPS, GSM triangulation, computer vision methods and satellite images. All of these examples describe the same physical phenomena by measuring them with different methods and instruments.

The downside of acquired data is that we cannot trust it to be consistent, precise, or even complete, the upside is that by using multiple instruments we gather in most cases redundant information about the same phenomenon. This results in the following challenges: generate consistent and unambiguous models from hybrid measurement data. This includes the statistically or empirically valid interpolation of gaps in the measurement, and correcting known artifacts of the applied measurement technologies; reduce data volume and create representations for the next processing step in the workflow.

Acquisition in the context of visualization refers mainly to techniques that derive information from raw data to generate high quality, high performance, meaningful, and user-friendly visualizations. This includes data enhancement methods like denoising and filtering, compression, hierarchical or topological re-organization of the data, feature extraction and classification methods, segmentation of structures of interest, and automatic derivation of high level information on the basis of previously generated segmentations. There is a direct relationship of acquisition in this context to the challenges semantics and fusion. Both require additional derived information from raw data – semantics is often related to features and objects present in the data. Such objects have to be detected and classified to provide the basis for a semantically defined visualization. Fusion needs a description of correspondence between individual datasets to be able to relate different representations of the same object across multi-source and multi-level data. Acquisition is a necessary pre-processing step for both challenges.

To generate consistent representations of real-world objects for rendering, documentation, or simulation purposes, methods have to be invented and improved to reduce these errors to an acceptable minimum. While visualization techniques may not require a topologically consistent representation of an object, some simulation algorithms will not work with ambiguous or non-manifold geometry. While this has been true even for manually modeled objects, it is especially problematic to reconstruct consistent representations from huge data sets acquired with laser scanners and other acquisition systems.

Systematic errors introduced by these instruments have to be handled on a consistent basis: one has to identify critical data at the earliest possible stage in the workflow, before erroneous data is merged into other, possibly correct data sets. There are multiple possible error sources that have to be handled. Random sampling noise can be reduced by over-sampling in time or space, thereby gathering more samples which can be smoothed using statistical tools. Systematic error can be reduced using redundant information, possibly from other modalities, or by previous gathered calibration data. Aliasing, i.e. under-sampling in the time- or spatial domain without low-pass filters, can produce artifacts which – according to the sampling theorem – cannot be removed using the measured data alone. Here additional data

acquisition is necessary.

VII. SUMMARY

Some people consider most computer graphics problems as solved. But while this is true for some areas with simple use of computer images, the embedding of computer graphics technology in increasingly complex surroundings generates many new challenges.

We have described five major challenges that are orthogonal to the traditional computer graphics fields rendering, visualization, virtual reality, and so on. Each of these challenges has consequences on the future development of visual computing techniques, especially for their application in many more application fields. These challenges are scalability, semantics, fusion, interaction, and acquisition. We have explained and highlighted many open research issues and why they have to be addressed.

ACKNOWLEDGEMENT

Robert Tobler has helped significantly in the discussion of these challenges and in the formulation of parts of the text.

REFERENCES

1. I.E. Sutherland, "Ten Unsolved Problems in Computer Graphics", Datamation, vol. 12, no. 5, May 1966, pp. 22-27.

2. P. Heckbert, "Ten Unsolved Problems in Rendering", Workshop on Rendering Algorithms and Systems, Graphics Interface ‘87, 1987.

3. C. Johnson, “Top Scientific Visualization Research Problems”, IEEE Computer Graphics and Applications, vol. 24, no. 4, July 2004, pp. 13-17.

Expert System Used to Help the Beginner Driver to Learn the Driver Skills

Gheorghe Puşcaşu, Codreş Bogdan, Codreş Alexandru
"Dunărea de Jos" University Galați, Faculty of Computer Science

***Abstract*—The aim of this paper is to develop an expert system which is dedicated to help the beginners to learn the driver skills.**

The mathematical model of the car represents the essence of the expert system. In training is used a simplified model of the car, retaining only what's important for the beginners.

A special importance in the development of an expert system it has the analysis of the requirements of the domain on operate.

For the proposed expert system, the rules become actives only when specified conditions are accomplished in a time range.

I. INTRODUCTION

USUALLY the activity that an expert system is accomplishing represents an activity specific for human experts. If the human expert has knowledge in some domain, acquired after training or a personal experience, an expert system uses the information acquired in a cognitive database specific for that domain. Generally, the cognitive database of an expert system is achieved from the knowledge transferred by a human expert.

In this context, it's enunciated the following definition: An expert system represents a software system that by the knowledge acquired from an expert (usually a human expert) solves complex problems from a certain domain of expertise.

In the *Expert System* definition [2],[3] are encountered the notions *expert* and *expertise* that have the following significations: An expert is a person that has specific knowledge in a certain domain, which means he is capable of making an expertise in that domain. That means the expert can solve problems corresponding to that domain, problems that other people cannot solve at all or cannot solve in an efficient way.

An expert system which contain a complete *knowledge base* [3] allows us a quickly and efficient solve of any problem encountered in the training.

Mostly, the expert systems are used when is necessary an ample analysis with time consuming for the proposed problem.

Domains where expert systems were implemented are: medical diagnosis [4], [5], fault diagnosis [6], service centre of spare parts [7], agriculture, education etc.

Expert systems typically have a number of several components. The *knowledge base* is the component that contains the knowledge obtained from the domain expert. The *inference engine* is the component that uses the knowledge found in the knowledge base, which are needed to solve the current task. The *user interface* is the component that allows the user to query the system and receive the results of those queries.

An expert system used in the assisting of student drivers [8],[9] is essential to acquire quicker driver skills. For the development of the knowledge base human experts which work in this area were used. The success of the expert system's implementation supposes a complete analysis of all aspects and use of those elements that guarantee efficiency and easiness in development.

In chapter II the structure of the expert system is presented and also some other components which traditionally don't belong to that, but for a practical implementation are necessary.

In chapter III the components of the expert system are presented in detail.

II. GENERAL PRESENTATION OF THE EXPERT SYSTEM

The expert system, dedicated for examination of the correctly car manoeuvring [10], [11], has the function to inform the student about the incorrect manoeuvring of the car, command and control elements. This expert system ensures that the students are supervised and corrected through the processes of car starting manoeuvre and traffic driving.

The opportunity of using an expert system in this domain is justified by the following affirmations:

- the absence of the expert system implies the presence of an instructor along the student during the training. As is already known, the instructor, besides the student training activity, has also the obligation to intervene in the car's control elements in order to avoid accidents. With the auto simulator, the second function of the instructor is not needed because the car manoeuvring errors don't have physically repercussions;
- usually, the instructor's job regarding the student's supervision and correction has a subjective character and sometimes needs a constant attention and availability through the driving process;
- the driving training domain is a clean-cut domain in which the rules are strictly formulated and well-known;
- the automotive training domain has a large number of people with a great experience. Therefore, the development of an expert system implies only the

acquisition of knowledge from these experts.

- during the use of the auto simulator, the student benefits from the continuous surveillance and objective information regarding the control of the vehicle and the behaviour in traffic;
- the student who is training on the auto simulator has the available option of invalidating the messages given by the expert system, but these are stocked and become available at the end of the training period;
- according to the mistakes made by the student, special circuits or tracks can be established to obtain a complete and fast training.

The block schema utilized to train the future drivers is shown in figure no.1 and is structured in three components:

- the acquisition block of resulted values following the action of the control elements of the car and the modelling of the car's engine (A) [12].
- the block corresponding to the expert system utilized to monitor the control elements of the car (B);
- the audio-video system necessary to inform the drivers about their mistakes (C);

In the following two chapters the reduced mathematical model of the car and the expert system will be presented.

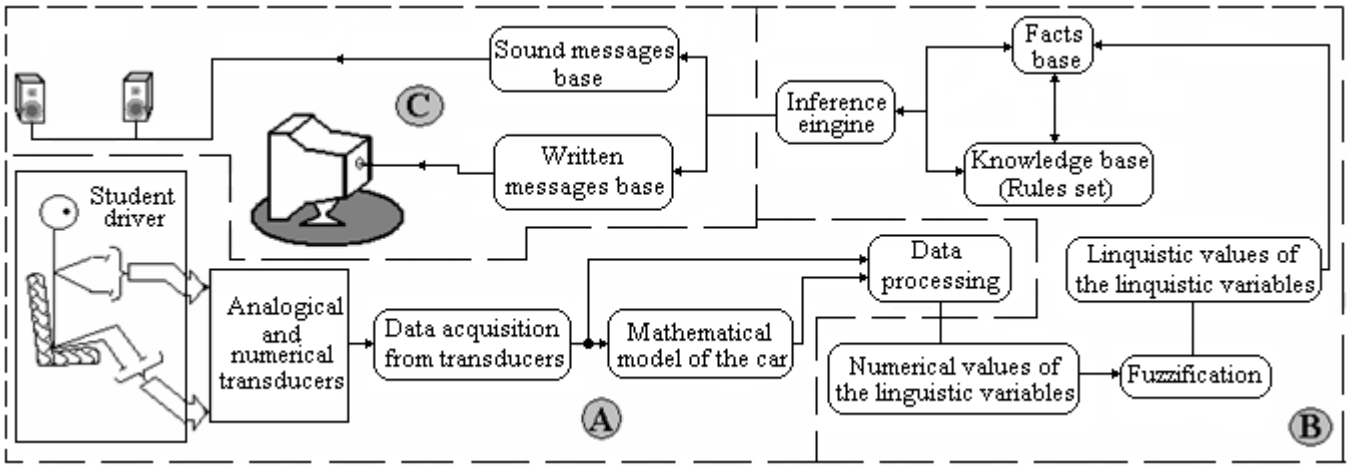


Fig. 1. The block schema of an expert system.

III. THE REDUCED MATHEMATICAL MODEL OF THE CAR

The reduced mathematical model of the car can be obtained using the following stages:

- compute the engine power (P) and engine moment (M) using the following relations :

$$P = P_{max} \cdot [\alpha_1 \cdot \frac{n}{n_P} + \alpha_2 \cdot (\frac{n}{n_P})^2 + \alpha_3 \cdot (\frac{n}{n_P})^3] \quad , \quad (1)$$

$$M = M_{max} \cdot [\alpha_1 + \alpha_2 \cdot \frac{n}{n_P} + \alpha_3 \cdot (\frac{n}{n_P})^2] \quad ,$$

where:

P_{max}, M_{max} - represent the maximum values of P and M;

α_1, α_2, α_3 - are coefficients which depend by the car type;

n – the revolution;

n_P – the revolution corresponding to P_{max}.

- compute the resistance moment of the clutch shaft (M_{rcs}) in function by the resistance forces using the following relations

$$M_{rcs} = \frac{(R_r + R_p + R_a) \cdot r_r}{i_t}, \quad (2)$$

where: R_r - the rolling resistance, R_p- the slope resistance, R_a - the air resistance, i_t - the transmission rapport, r_r - the wheel ray.

- compute the angular acceleration of engine shaft (ε_m), of clutch shaft (ε_a) and wheel (ε_r) in function by the transmission rapport as follows:

$$\varepsilon_m = \begin{cases} \frac{M_m - M_{cs}}{i_m} & if \quad \omega_a < \omega_m \\ \frac{M_m - (M_{rcs} + M_{fr})}{i_{ar} + i_m} & if \quad \omega_a = \omega_m \end{cases} \quad (3)$$

$$\varepsilon_a = \begin{cases} 1.8 \cdot \frac{M_{cs} - (M_{rcs} + M_{fr})}{i_a} & if \quad \omega_a < \omega_m \\ 1.8 \cdot \varepsilon_m \quad if \quad \omega_a = \omega_m \end{cases} \quad (4)$$

$$\varepsilon_r = \frac{\varepsilon_a}{i_t} \quad (5)$$

where: M_{fr} – the braking moment, M_{cs} - the clutch shaft moment, M_m – the engine moment, i_{ar} – the inertial moment of the clutch shaft, i_m – the inertial moment of the engine, i_t - the transmission rapport, ω_m , ω_a - the angular speeds of the engine and the clutch shaft.

- compute the angular speed of the wheel using the following equation with differences

$$\omega_r^{(t)} = \omega_r^{(t-1)} + \varepsilon_r \cdot \Delta t \, , \quad (6)$$

- compute the car linear speed using the angular speed

$$v = \omega_r^{(t)} \cdot r_r \, , \quad (7)$$

In fig. 2 the variables values resulted from the dynamical model of the car are illustrated. It is mentioned that the variables: the engine revolution, connecting gear revolution, speed rank and the brake are multiplied with a factor. A linear transformation of the values (LTV) is used for obtaining a good representation.

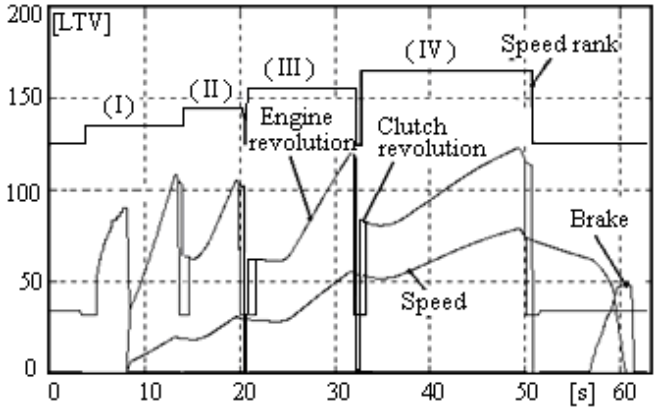


Fig. no.2 The evolution of variables during the simulation.

The obtained results underline the following aspects:

- the clutch revolution is equal with the engine revolution at the moment when the clutch is coupled;
- the engine revolution decrease at the nominal value when the clutch is not coupled;
- the speed gradient is big when the brake is applied;

IV. THE EXPERT SYSTEM USED TO MONITOR THE DRIVER'S ACTIONS OVER THE CONTROL ELEMENTS OF THE CAR

The proposed expert system allows us to monitor the driver's actions over the control elements of the car, so that the students get proper skills in driving a car.

The expert system dedicated to the student's training can be structured in four subsystems as follows:

- the generation subsystem of the numerical values corresponding to linguistic variables;
- the subsystem for the fuzzy approach [13], [14] of the variables;
- the subsystem of rules activation (inference engine);
- the knowledge rules – types of rules.

Next, the block components of the expert system designed to monitor the student will be presented in detail.

A. The generation subsystem of the numerical values corresponding to linguistic variables

The expert system continuously receives information from sensors or from the mathematical model [15] of the car. This information is obtained even when the vehicle speed is zero.

The signals obtained from the transducers come from: clutch, throttle, brake, handbrake, wheel and gear shifter.

The linguistic values and their working ranges are given in the table I.

TABLE I

Linguistic variable	The working range	The physical variables corresponding to the linguistic variables
L_V_Revolution	*[600 rot/min 5500rot/min]*	*Engine's revolution*
L_V_Speed1	*[0 km/h 100 km/h]*	*Car's speed in first gear*
L_V_Speed2	*[0 km/h 100 km/h]*	*Car's speed in second gear*
L_V_Speed3	*[0 km/h 120 km/h]*	*Car's speed in third gear*
L_V_Speed4	*[0 km/h 120 km/h]*	*Car's speed in fourth gear*
L_V_clutch	*[S_CLUTCH_MIN; S_CLUTCH_MAX]*	*Clutch pedal span*
L_V_throttle	*[S_THROTTLE_MIN; S_THROTTLE_MAX]*	*Throttle pedal span*
L_V_brake	*[S_BRAKE_MIN; S_BRAKE _MAX]*	*Brake pedal span*
L_V_wheel	*[S_WHEEL_MIN; S_WHEEL_MAX]*	*Wheel span*

The minimum and maximum limits of the variables: *L_V_clutch, L_V_throttle, L_V_brake, L_V_wheel* can be found in the transducer's configuration file.

B. The subsystem for the fuzzy approach of the variables

The linguistic variables will be described through the linguistic values, values that should have the following properties [16]:

- to be formulated using elements of natural language;
- to have an informational content close to the real one;
- the linguistic variable is defined in a certain context.

The linguistic variables, the linguistic values corresponding to the physical variables and the set values of the working range are presented in the table II. For the appreciation of the affiliation degree of a generic element to a certain set, fuzzy sets [17], [18] are used.

TABLE II

Linguistic variable		Numerical values wherefore the affiliation coefficient is equal to 1
L_V_Revolution	*SN (sub normal)* *N (normal)* *OR1(over rev level 1)* *OR2(over rev level 2)* *OR3(over rev level 3)*	*[600 rpm ; 1000 rpm]* *[1000 rpm ; 2400 rpm]* *[2400 rpm ; 4000 rpm]* *[4000 rpm ; 5000 rpm]* *[5000 rpm; 5500 rpm]*
	SN (sub normal)	*[0 km/h; 8 km/h]*

L_V_Speed1	*N (normal)* *ON (over normal)*	*[8 km/h; 17 km/h]* *[17 km/h; 100 km/h]*
L_V_Speed2	*SN (sub normal)* *N (normal)* *ON (over normal)*	*[0 km/h; 14 km/h]* *[14 km/h; 31 km/h]* *[31 km/h; 100 km/h]*
L_V_Speed3	*SN (sub normal)* *N (normal)* *ON (supra normal)*	*[0 km/h; 21 km/h]* *[21 km/h; 50 km/h]* *[50 km/h; 120 km/h]*
L_V_Speed4	*SN (sub normal)* *N (normal)* *ON (supra normal)*	*[0 km/h; 30 km/h]* *[30 km/h; 90 km/h]* *[90 km/h; 120 km/h]*
L_V_clutch	*TC (totally coupled)* *PC (partially coupled)* *NC (not coupled)*	*[MIN; MIN+(1/10)*DOM]* *[MIN+(1/10)*DOM; MIN+(1/2)*DOM]* *[MIN+(1/2)*DOM; MAX]*
L_V_throttle	*NP (not pressed)* *PP (partially pressed)* *TP (totally pressed)*	*[MIN; MIN+(1/3)*DOM]* *[MIN+(1/3)*DOM; MIN+(2/3)*DOM]* *[MIN+(2/3)*DOM; MAX]*
L_V_brake	*NP (not pressed)* *PP (partially pressed)* *TP (totally pressed)*	*[MIN; MIN+(1/3)*DOM]* *[MIN+(1/3)*DOM; MIN+(2/3)*DOM]* *[MIN+(2/3)*DOM; MAX]*
L_V_wheel	*LFT (left)* *MID (middle)* *RGT (right)*	*[MIN; MIN+(1/3)*DOM]* *[MIN+(1/3)*DOM; MIN+(2/3)*DOM]* *[MIN+(2/3)*DOM; MAX]*

Numerical values from table II corresponding to the last four linguistic variables have the following meanings:

MIN represents the minimum numerical value of the variable
MAX represents the maximum numerical value of the variable
DOM represents the domain of the linguistic variable expressed by the following subtraction:

DOM=(MAX-MIN)

The minimum and maximum values of the variables are loaded from the transducer's configuration file that is modified by a human operator.

After the fuzzy approach of the numerical values corresponding to the linguistic variables, the linguistic values represent facts. For example, at the moment when the clutch is uncoupled, the linguistic variable *L_V_clutch* has the linguistic value NC - *not coupled*, which is a fact that can activate a certain rule. The expert system also activates a message like: *"it is not indicated to keep pressed the clutch pedal"*. It must be mentioned that this rule activates itself when the beginner driver forgets to release completely the clutch pedal, keeping the foot on the pedal without making manoeuvres that involves the uncoupling of the clutch.

The linguistic variables that are used have assigned themselves a temporal variable that measures the time until the linguistic variable has a linguistic value well determined. For example, in the situation mentioned above, it is very important that the message *"it is not indicated to keep pressed the clutch pedal"* is activated only after the linguistic variable *L_V_clutch* has the linguistic value *NC - not coupled* for a time range bigger than 10 seconds.

C. *The subsystem of rules activation*

The rules activation, based on the facts obtained after the evaluation of the linguistic variables, is made by the inference engine. The used rules have a flexible format, similar to the natural language. The rules base is very flexible because new rules can be appended easily.

The rules base, at this moment, contains a number of 47 rules. The format of a rule is:

{
Driver type

Rule priority

Linguistic variable *Time variable*

Linguistic variable's value *The time corresponding to the linguistic variable*

.
.
.

Linguistic variable *Time variable*

Linguistic variable's value *The time corresponding to the linguistic variable*

Message identifier
}.

The rules format allows the differentiation of the messages depending on the type of the driver (beginner, advanced etc.) as well as the solving of conflicts between two rules that are activated simultaneously by assigning to each rule a priority. Thus, a rule with a higher priority will invalidate a rule with a smaller priority.

The rules base permits the training of the beginner driver when the engine is started and the car's speed is bigger than zero but also when the car's speed is zero.

In the next paragraph different types of rules are presented to accentuate their pattern.

D. Types of rules

In this section, the most important rules used in the expert system are presented.

- *Rules for warning the driver about the fact that he is over revving the engine*

The structure for such a rule is:

```
{
 1
 3
                L_V_Revolution     dt     L_V_clutch
                 OR1               >12      NC
 1
}
```

and it activates itself when *the clutch is uncoupled* and the engine is over revved (level 1) for a period of time longer than 12 sec. The written and audio message transmitted to the beginner driver is: *"do not rev the engine"*.

In the rules base there are 6 more rules that activate that message. The difference between them is that they activate the message after a shorter period of time in case that the linguistic variable *L_V_Revolution* has the value *OR(level 2)* or *OR (level 3)*.

- *Rules for warning the driver about the fact that the car's speed is not adapted with the current gear.*

These rules are activated, usually, when the car does not work in nominal parameters. The structure for such a rule is:

```
{
 1
 2
Gear     dt   L_V_Speed1  dt   L_V_clutch  L_V_brake
 1       >0.4   ON        2,2.5    TC           NP
 3
} .
```

The message *"the speed is too high for first gear"* is activated when: the current gear is the first gear for more than 0.4 sec, the linguistic variable *L_V_Speed1* has the value *ON (over normal)*, the clutch is totally coupled (the linguistic variable *L_V_clutch* has the value *TC*) and the brake is not pressed.

Rules from the rules base that are similar with the one enunciated above, differs from it by the fact that the gearshift may be in other gears, and the speed of the car doesn't necessarily have to correspond to that gear.

- *Rules concerning the gear change when the speed does not match the changed gear.*

For example, if the speed of the car is higher than 31 km/h and the car is in third gear, and the beginner driver wishes to change in the second gear, then he must be warned that usually it's not indicated to change in second gear at this speed.

The rule which informs the beginner driver that the speed is too high for the second gear has the following structure:

```
{
 1
 2
PreviousGear    Gear     dt   L_V_Speed2  L_V_clutch
 3              2        <3     ON           TC
 9
}.
```

It is noticeable that the time range in which this rule is activated is (0s 3s) from the moment the second gear was changed. If this rule would be activated after a longer period of time, then the linguistic variable *L_V_Speed2* could change its state from *ON* because the speed is lower.

The rule base may also contain rules that inform the beginner driver if he had changed in a higher gear and the speed is too low.

- *Rules that inform the* beginner *driver that he is using lower gears at high speed.*

When the beginner driver uses second or third gear and the speed is too high, he must be warned that the engine speed is too high.

The structure of the rule is:

```
{
 1
 3
Gear     dt   L_V_Speed3  dt   L_V_clutch
 2       >0.4   ON        >3    TC
 23
}.
```

It is noticeable that the rule is activated only if the car's speed in second gear is very high (engine over rev) for a period of time longer than 3s.

- *Rules that inform the* beginner *driver that he is rotating the steering wheel too much at high speed.*

  When the speed of the car is higher than 50 km/h, which means that the linguistic variable *L_V_Speed3* value is ON, and the linguistic variable *L_V_wheel* value is *LFT*, the beginner driver is informed by the next message: *"Danger of losing control of the vehicle at this speed"*.

```
{
 1
 2
L_V_wheel   L_V_Speed3
    LFT        SPN
 20
}
```

Also, when the steering wheel is rotated to the right, and the linguistic variable *L_V_Speed3* value is ON, the beginner driver receives the same message. In certain situations this message can be accompanied by car skidding.

- *Rules that are activated when the beginner driver keeps his foot down on the clutch longer than is needed to make the gear change.*

The rule has the following structure:

```
{
1
1
Gear      L_V_clutch      dt
1             PC          >25
22
}.
```

The rule is activated when the clutch is partially coupled longer than 25s, and the gearshift is in the first gear. The rules are similar for the other gears, only the time is shorter because at high speed the time needed to change the gear is also smaller.

- *Rules which inform the beginner driver he is changing in a lower gear skipping one gear.*

When the gear is changed from the fourth gear to the second gear, and the linguistic variable *L_V_Speed2* value is ON, the *beginner* driver is warned by the next message: *"You changed the gear inappropriate"*.

The rule mentioned above has the following structure:

```
{
1
3
PreviousGear     Gear     L_V_Speed2
4                  2          ON
21
}.
```

According to the rules mentioned above it can be stated the fact that the suggested expert system allows an objective and compelling information to the drivers during the training period on the simulator. Besides what was mentioned above, the expert system used for training will become, in a future approach, the module that allows collecting data for a knowledge acquisition system concerning the beginner driver's behaviour. According with the beginner driver's behaviour can be elaborated strategies to minimize the training time and forming a preventive behaviour.

V. CONCLUSION

The expert system presented in this paper represent an efficient instrument in drivers' training because helps them to correct the wrong manoeuvres. The implementation is based on numerous papers, books and experts existing in this domain.

The linguistic variables used in the expert system are calculated based on information obtained from the transducers and from the mathematical model of the car.

The rules of the expert system cover almost all possible manoeuvres that the beginner driver may execute during driving.

The time variable connected with the linguistic variables allow the differentiation between a correct and a wrong manoeuvre.

In future works we will develop analysis modules for driver's behaviour and will establish adaptive strategies for traffic's configuration and traffic's geometry for training in minimum time.

REFERENCES

[1] A. Bar, A.E. Feigenbaum, *The Handbook of Artificial Intelligence*, vol. I, II, William Kaufman. Los Altos, CA.

[2] *I. D. Crâstoiu*, "Expert Systems", Ed. All, 1994.

[3] *J.R. Mockler and G.D. Dologite*, *Knowledge-Based Systems. An Introduction to Expert Systems*, Macmillan Publishing New York, 1992.

[4] K. P. Adlassnig, "Fuzzy set theory in medical diagnosis," *IEEE Trans. Systems, Man, Cybern.*, vol. SMC-16, pp. 260–265, Mar. 1986.

[5] K. P. Adlassing and G. Kolarz, "Representation and semiautomatic acquisition of medical knowledge in Cadiag-1 and Cadiag-2," *Comput. Biomed. Res.*, vol. 19, pp. 63–79, 1988.

[6] T. Whalen, B. Schott, and F. Ganoe, "Fault diagnosis in fuzzy networks,"in *Proc. Int. Conf. Cybern. Soc.*, Seattle, WA, Oct. 1982, pp.35–39.

[7] I.B. Turksen, Y. Tian, and M. Berg, "A fuzzy expert system for a service centre of spare parts," *Int. J. Expert Syst. Applicat.*, vol. 5

[8] W. Kading and F. Hoffmeyer, "The Advanced Daimler-Benz Driving Simulator", SAE paper 950172, 1995.

[9] J. Miguel Leitão, A. Coelho, F. N. Ferreira, "DriS – A Virtual Driving Simulator", Proceedings of the Second International Seminar on Human Factors in Road Traffic, ISBN 972-8098-25-1, Braga, Portugal, 1997

[10] E. Boer, T. Yamamura, N. Kuge, A. Girshick "Experiencing the same road twice: a driver centered comparison between simulation and reality", in *Proceedings of the Driving Simulation Conference* DSC'2000, Paris, France, Sept 2000, pp 33-55.

[11] P. L. K. Jones, "REVEAL: An expert systems support environment," in *Expert Systems: Principles and Case Studies*, R. Forsythe, Ed., 1st ed. London: Chapman & Hall, 1984.

[12] Gh. Puşcaşu, V. Puşcaşu, "Aspects regarding the modelling of the car used to implementation of a simulator", 7th International Symposium on Automatic Control and Computer Science, Iasi, Romania, October 26-27, 2001.

[13] A. L. Zadeh, "Fuzzy Sets as a Basis for a Theory of Possibility", Advances in Fuzzy Systems-Applications and Theory, Fuzzy Sets and Systems, 1971.

[14] L. A. Zadeh, "Fuzzy sets," *Inform. Contr.*, vol. 8, pp. 338–353, 1965.

[15] Gheorghe Puscasu, Bogdan Codres, et all., "Nonlinear System Identification Based On Internal Recurrent Neural Networks", *International Journal of Neural Systems*, Vol. 19, No. 2 (2009) pp,115–125, 2009.

[16] D. Driankov, et. al., *An Introduction to Fuzzy Control*, Springer-Verlag, 1993.

[17] J. F. Baldwin, "Evidential support logic programming," *Fuzzy Sets Syst.*, vol. 24, pp. 1– 26, 1987

[18] M. Sugeno, *Industrial Applications of Fuzzy Control*. Amsterdam: North-Holland, 1985.

Control Algorithms for a Self Reconfiguring Robotic System

M. G. Răducu[1], M. Nițulescu[2]
1PhD Stud., University of Craiova, *mihai_raducu@yahoo.com*
2Prof. Eng., University of Craiova, Faculty of Automation, Computers and Electronics,
nitulescu@robotics.ucv.ro, *Member IEEE*

Abstract **- These self reconfiguring robots are complex robotic systems composed of a definite number of modules, where each module of the system is considered to be a simple agent. Based on this context, the process of movement and reconfiguration will have the following definitions: the reconfiguration represents the process of re-arranging each agent of the robotic system in order to have a working system. One versatile method to determine the most adequate movement and reconfiguration algorithms can be done using modeling methods and tools. Using this approach, the control algorithms necessary to drive such a system can be modeled, tested and implemented in an embedded system much easier than in a classical approach. This paper presents the case of a self reconfiguring robot for which, a reconfiguration algorithm and a movement algorithm were determined, analyzed, tested and validated.**

I. Introduction

THE appearance of self reconfiguring robots was generated by the need of implementing all the characteristics that the fixed robotic structures do not have.

The self reconfiguring robots must be able to adapt their shape to different environmental operation condition and at the same time do choose the optimal movement solution, based also on the outside world information.

A self reconfiguring robot is composed of a set of robotic modules which can reconfigure geometrical (autonomous and dynamic) in a variety of different shapes, in order to be able to adapt to the environmental condition and also to the task that it has to fulfill.

In a conventional design workflow, the functionality of a robot is determined using a top to bottom method, starting from the highest level of abstraction to the lowest one (it starts from the task that the robot should fulfill up to the connecting systems and the electronic devices needed to create such a robot).

The self reconfiguring robots can be viewed from the minimalist point of view of the versatile and extendable robots, which where designed from the beginning with more types of actuators, but which maintain those solutions that fit best to possible operation scenario that may appear in a real operating activity. .The design workflow for the self reconfiguring robotic structures starts with determining the functional requirements of the robot, which will determine the possible characteristics of the final robot. Due to the fact that these types of robots can modify their structure, they have a built-in latent potential.

One property of the self reconfiguring robots is the capability of auto assembly. In comparison with classic robots, the self reconfiguring robots can dynamically modify their structure so that the entire system can fulfill the task that was given. This process is called morphogenesis or shape changing. One important characteristic of morphogenesis is given by the type of the modules that compose the robotic structure: identical modules or modules that differ. The main problems that the morphogenesis has to deal with are:

- Finding an algorithm that will allow a reconfiguration from any type of shape to a specific one
- Finding a reconfiguration path from a certain configuration to another one
- Finding the right configuration based on the operation environment

Self reproducing capability is another characteristic of the self reconfigurable robots. This property represents the capability of the system to reproduce a copy of it own.

Another important characteristic is the possibility of movement generation. An autonomous movement for each possible configuration is needed. One of the most practical solutions is the creation of some kind of data base that will contain all the possible movements that the robot will do and starting from this, a movement typology can be generated for each structure, individually. An improved method of storing the commands for reconfigurations and movement processes is the use of hormones.

There is a close relation between the shape of a robot and the movement that this structure must do. The robotic structure will determine what types of movements can be executed, but on the other hand also, the type of movement that a self reconfiguring robot must do, will have an influence on the mechanical structure. Shape and movement are two undividable characteristics, but they also represent two different aspects of the evolution cycle.

II. Basic Characteristics Of The Robotic Module

One of the most suitable robotic structures that have the possibility of autonomous evolution and also is capable of reconfiguring for adapting to continuous changing environments is the modular one. Due to their structure, modular robots can be used in activities which require skills for locomotion, in inspection, intervention, for penetrating environments which have a low degree of accessibility. A general classification of the robots with modular structure will be the following one:

- Chain shape robotic structure
- Lattice shape robotic structure

Chain shape structures have a biological back ground, being inspired from living forms like snakes, worms and other insects. The way these modules will be placed in the entire structure will generate a structure with a serial disposal (the case of the snake configuration), or with a tree disposal (the case a legged configuration).

For this structure, reaching this final configuration, and even intermediate ones, will be possible only with the help of a control system and a communication system that will do the connection between all the involved subsystems.

The type of the robotic structure for which the reconfiguration and movements algorithms will be studied in this paper are based on a modular structure which contains six identical modules (as is the case of the ROMAR-Miniature robotic system with self-reconfiguring and self-replicating skills, developed in the frame-work of CEEX-ROMAR Project [8].

Each module contains three subsystems, each one of this having the shape of a cube. Each module has 2 DOF, generated by 2 rotations. One rotation of ± 90 ° is done around the longitudinal side of the module and the second one, also with a movement of ± 90 °, is done perpendicular on the longitudinal one (see Fig. 1). Both rotations will have a big role in the process of robotic system movement and reconfiguration one. The module from Fig. 1 is a 3D representation of the module that is used inside the structure of the ROMAR robot.

For implementing those rotation movements, the robotic module will be fitted with two engines. Besides these two motors, each module will have to a microcontroller which will be in charge of controlling the steering of the motors.

Taken into consideration that this type of robotic structure is composed from several modules and the reconfiguration and the movement is seen from a macro level, the microcontroller will be in charge also for implementing the communication between module and synchronization of commands.

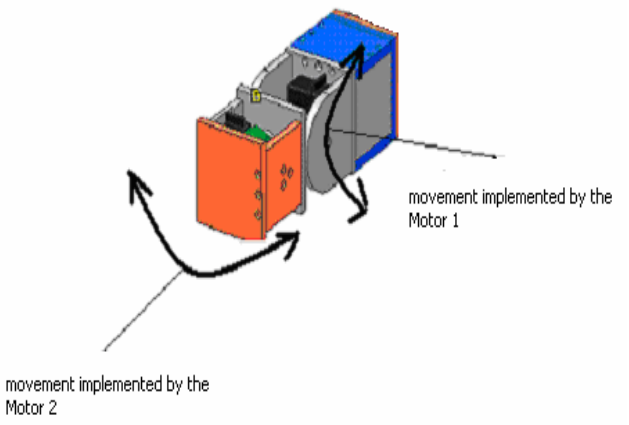


Fig. 1. Robotic module structure

Each module will have to use some special devices that will permit the connection to its neighbor at specific moments of time.

As we can imagine there is close relation between movement/reconfiguration algorithms, timing of events, information from the system, and the control mechanism.

III. Reconfiguration And Movement Algorithms

One of the most suitable robotic structures that have the possibility

Self reconfiguring robots with a modular structure try to cover all the applicable aspects that are meet on living organisms like snakes, worms, four legged animals, transmission of the information inside the human body, with the help of the hormones, and incorporate all these on a functional structure.

Analyzing all these aspects, several ways of implementing movement and reconfiguration algorithms were find:

- Reconfiguration from a snake structure to a four-legged one and vice versa
- Reconfiguration from a snake structure to a wheel one and vice versa
- Movement of the robot in snake configuration
- Movement of the robot in the four-legged configuration
- Movement of the robot in wheel configuration.

A. Reconfiguration from a snake structure to a leg type one

This type of configuration starts from a robotic configuration that has the shape of a snake or worm. In such structure all the modules are placed one after another and it has to reach a configuration that has four legs.

On the first phases of the reconfiguration the first 3 modules (Module1, Module2, Module3) will form a 90° angle with the remaining 3 modules (Module4, Module5, Module6) so that all the modules will be arranged so that they connect and start creating the body of the four-legged structure (see Fig.2).

The second major phase is considered to be the one in which all the future legs will be connected to the body. Module1 and Module2 will form the body of the new configuration robot and Module3, Module4, Module5 and Module6 will be the legs of the robotic structure (see Fig. 3).

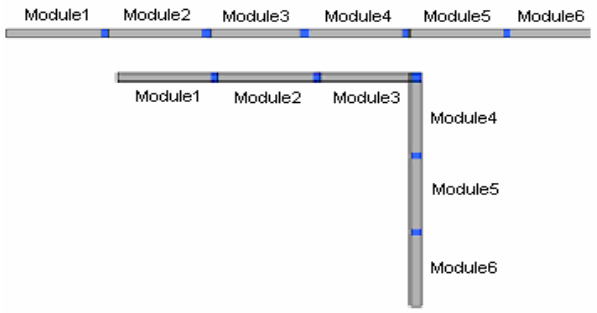


Fig. 2. First major stage of the snake to four-legged reconfiguration

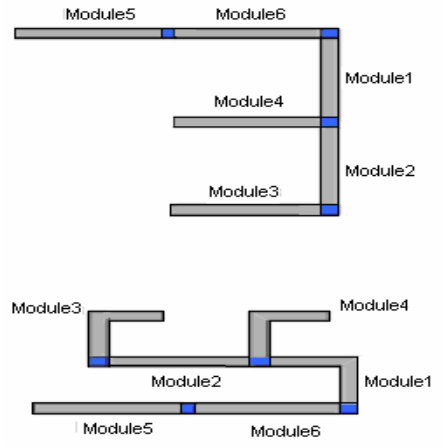


Fig. 3. Second major stage of the snake to four-legged reconfiguration

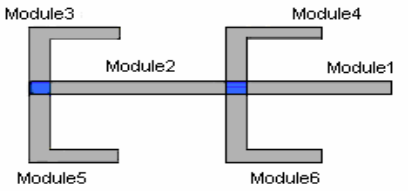


Fig. 4. Final major stage of the snake to four-legged reconfiguration

The third major activity is considered to summarize all the remaining movements until the robot reaches the four legged position. After finishing this phase, the robot can stand by its own, and must be able to move using the legs. Module5 and Module3 will be the front legs and the Module6 and Module4 will be the rear legs.

All these phases, with their intermediate sub phases, are summarized in the following table (see Table 1).

This table contains all the necessary movements for each module and the sequence of activation for each motor that will have to be activated. Also the table contains the angles that should be performed for each activation.

The sequence of events is designed that at certain moment of time only one motor must be activated. Each motor must be activated for a specific amount of time until the degree made by the module will be the one specified in the table. The degree mentioned in the table is a general one.

It is possible to group 2 ore more motor actuation under one sequence, but this combination can lead to structure blockage, and the reconfiguration process might reach a dead point from which will get out only via a new initialization.

B. Robot movement in four legged configuration

One of the important characteristics of today's robots that confer its mobility is the possibility of moving its structure in different types of environments.

One of the easiest environments that don't need special techniques is considered to be the plane surface. For this type of environment the best movement solution is the one when the robot is configured as a four-legged one.

TABLE 1.
RECONFIGURATION COMMAND FOR EACH MOTOR OF THE MODULES FROM THE ROBOTIC STRUCTURE IN CASE OF A TRANSFORMATION FROM SNAKE STRUCTURE TO FOUR-LEGGED ONE

	Module1		Module 2		Module 3		Module 4		Module 5		Module 6		Ang.
	M1	M2	M1	M2	M1	M2	M1	M2	M1	M2	M1	M2	
1								x					+90
2							x						+90
3								x					+90
4							x						-90
5									x				-90
6							x						-15
7					x								-90
8			x										-75
9	Disconnecting command between Mod.2 and Mod.3												
10			x										-90
11							x						+15
12					x								-90
13					x								+90
14									x				+90
15			x										+90
16											x		-15
17									x				-75
18					x								-50
19							x						-50
20	Disconnecting command between Mod.3 and Mod.4												
21											x		+50
22					x								+50
23							x						+50
24									x				+50
25	x												-50
26											x		+50
27							x						-90
28									x				-90
29	Disconnecting command between Mod.4 and Mod.5												
30									x				+90
31											x		+90
32	x												+90
33							x						+90
34		x											-90
35					x								-90
36							x						-90
37	x												-90
38									x				-90
39	Disconnecting command between Mod.6 and Mod.5												
40											x		
41	Disconnecting command between Mod.1 and Mod.6												
42	x												-90
43						x							+90
44								x					+90
45										x			+90
46												x	+90

Seen from up, the robot in a four-legged configuration has the same structure as the robot from Fig.4. In this situation each of the four legs are able to implement a rotational movement with a certain degree (this degree is totally dependent on the restriction generated by mechanical structure that is adopted). Each movement made by each model and the order in which these movements have to be done, will be presented next (see Table 2). Analyzing this table, it is possible to detect that a series of four commands are repeating – these commands can be interpreted as an informational hormone.

TABLE 2.
SEQUENCE COMMANDS FOR EACH MOTOR OF THE MODULES FROM THE ROBOTIC STRUCTURE IN CASE OF A FOUR-LEG MOVEMENT

	Module 1		Module 2		Module 3		Module 4		Module 5		Module 6		
	M1	M2	M1	M2	M1	M2	M1	M2	M1	M2	M1	M2	Ang.
1								x					+30
2												x	+30
3						x							-30
4										x			-30
5								x					-30
6												x	-30
7						x							+30
8										x			+30
9						x							+30
10										x			+30
11								x					-30
12												x	-30
13						x							-30
14										x			-30
15								x					+30
16												x	+30

IV. CONTROL ALGORITHMS IMPLEMENTATION

One way that can allow all these reconfiguration and control algorithms to be implemented, modeled and tested in a robotic structure is the use of powerful tools that allow modeling and simulation of a system. One tool that has all these characteristics is Matlab/ Simulink.

The approach, that is used for this model, starts from the basic idea of simulating the selection signal (or information) that is generated by a master control component (such is the case of a task planner) and all the information that can be received from the system (the feed-back information from different types of sensors), and give the right commands to the actuators. Each type of algorithm is implemented in a different subsystem that is activated when the corresponding activation information is received.

All the control algorithms that were presented in the previous paragraphs will be implemented in a Matlab model. The model is composed of two big subsystems and will have the classic structure of an automatic system (with system that has to be controlled and the feed-back that is collected from the operating environment, as it is presented in the Fig. 5).

One subsystem will simulate all the information that comes from the environment as feed-back information, and other command signal from a master entity and permits visualization of all the commands or outputs that belong to this system, outputs that have to be sent to actuators. This subsystem is called Environment (see Fig. 6).

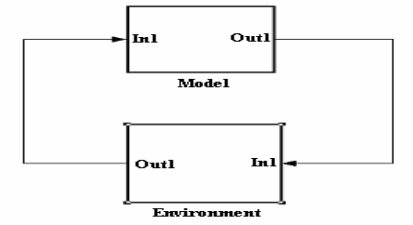


Fig. 5. Simulink model structure.

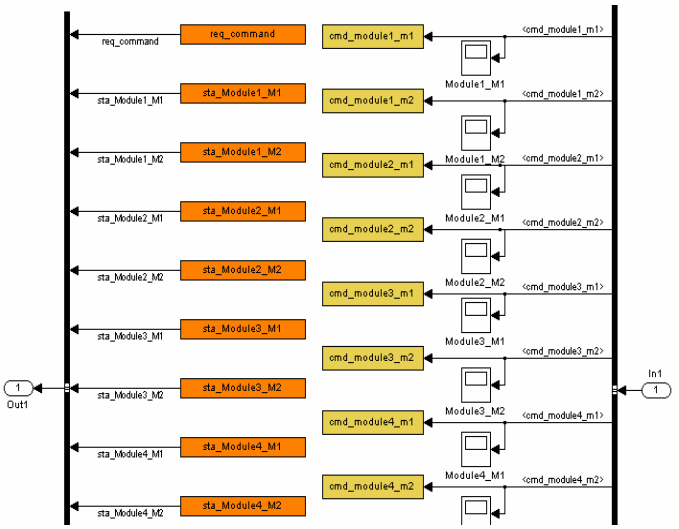


Fig. 6. Structure of the Environment Subsystem

The Model subsystem is the part of the Matlab model in which the control algorithms for reconfiguration and movement are implemented. This part of the simulation is used to validate the algorithms (via simulation) and after this procedure, this subsystem can be used to auto generate code for a specific type of microcontroller (the same type of microcontroller that is found on the real hardware).

All the information that is processed in this subsystem and is considered to be input information for the Model subsystem is placed in the Environment subsystem. There are two types of class signals. One of this is represented by a signal called req_command, and the other one is represented by the feedback information that is generated in a real system by all the sensors that are used for module position monitoring.

The signal req_command will have different values each one with a special meaning. For example if the value 1 is set, this will have the meaning of a request for configuration from a snake structure to a structure with 4 legs. If the value 2 is set, then the central control module will ask for a reconfiguration from snake structure to a wheel structure, and if the value of the signal is 3 this will have the meaning of a request for movement in four legs configuration.

The signals sta_Modulei_Mj, with i taking vales from 1 to 6 (because the robotic structure is made out of 6 modules) and j taken values from 1 to 2 (because there are only 2 motors for each module) represents the feed back information for each motor. This information will tell if the motor reached is desired value or not, and based on this, the next sequence can be started or not.

Due to the fact that there are 12 motors that have to be actuated, the model is designed with 12 command signals. Each command signal is named using the following format: cmd_modulei_mj with i representing the number of the module and j representing the number of the motor.

All the information is read and stored using the Matlab Workspace. In this way the testing procedure can be

automated and the sequence of events that have to generated and interpreted is better managed.

The core of the Simulink model is represented by the Model subsystem. This subsystem covers all the logic necessary to implement the reconfiguration and movement algorithms (see Fig. 7).

The Model subsystem is composed of 4 subsystems, each of them having a precise scope. Three of these subsystems are dealing with implementation of the algorithms presented in the previous paragraphs (one for configuration from snake to four legs, one for configuration from snake to wheel, and another one for movement algorithm).

Each subsystem will have as input the status information from the system as sta_Modulei_Mj and the type of command using the information req_command and based on this the system and the logic behind it will be activated or not.

The outputs of this subsystem will be the commands that will have to be sent to the 12 actuators, as they are generated by each subsystem. Due to the fact that each module generates its own outputs for driving the motors (but only one at a time) a decision block had to be implemented. This decision block takes each input from the previous 3 subsystems that are implement the reconfiguration and movement algorithms and based on the type of command that is received via the req_command signal, only one output is transmitted to the actuators.

Depending on the complexity of the algorithms and the implementation method, each subsystem of the model that controls the movement and the reconfiguration process has at least a Matlab Stateflow block. For example, the subsystem called “Snake _Four_legs”, which is in control of the reconfiguration mechanism from a snake structure to a four-legged structure has a Stateflow block (see Fig. 8).

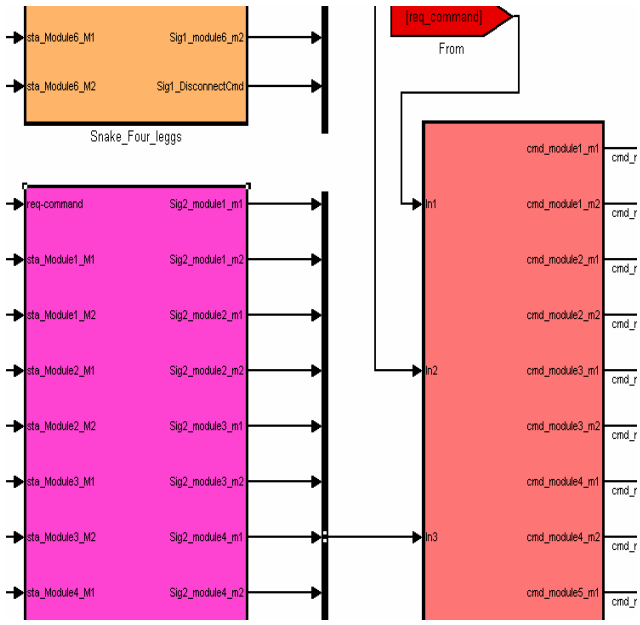


Fig. 7. Structure of the Model Subsystems

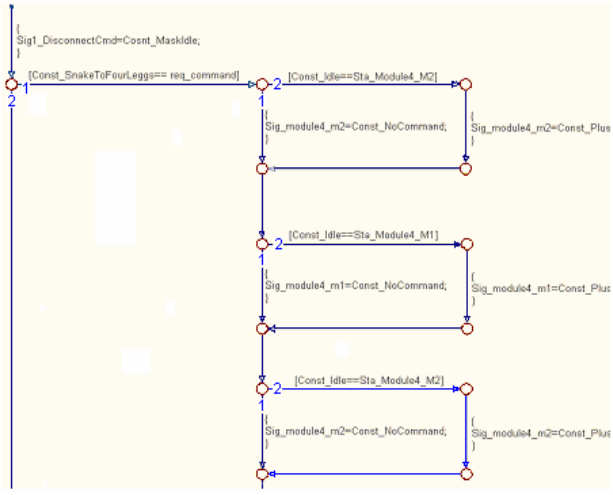


Fig. 8. Stateflow implementation for reconfiguration from snake structure to a four legged one.

Fig. 8 shows a part of the Stateflow block in which the synthesized command from the Table 1 are implemented. On the default transition the algorithm makes a reset of all the disconnect commands. After that it checks if the planner has sent the request for this specific reconfiguration and if the request is available, it starts activating the specific motors from the specific modules (as it was determined in Table 1).

The commands given to actuators have predefined values in order to make a difference between the values of the angular movements that have to be made. These values will be used at the basic software layer (the software that is composed of drivers and is the closest one to the actuators).

One characteristic of using Matlab/Simulink tool chain is the possibility to use the model that is created for testing and validating a certain algorithm, for code generating. This code can be placed with other code modules in a project that will cover all the logic for having a functional robotic system. This is why each value of the outputs from Model subsystem has to be clearly defined.

V. Testing Results and Conclusions

For testing the implementation from the Environment subsystem, the value of the signal req_command was set to the value 1. This means that there is a request from a snake configuration to a four-legged one.

The first 5 sequences of events were regarding command that imposed a movement of motors with degrees of 90° and -90°. For these types of commands there are the following commands values that have to be given:

- for a 90° movement the value of the command that has to be sent to the output of the system is 10.
- for a -90° movement the value of the command that has to be sent to the output of the system is 5.

The result of the simulation for the reconfiguration from snake to four leg structure (with the value 1)is presented in the following figure (see Fig. 9).

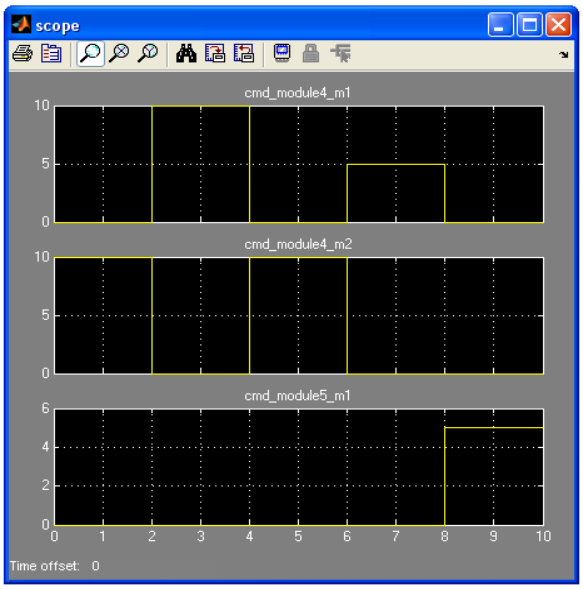


Fig. 9. Scope results in case of a reconfiguration from snake structure to four legged one.

This simulation covers only the first 5 sequences of commands that have to be given to the motors M1 and M2 of the robotic module 4 and first motor (M1 motor) of the fifth robotic module.

Other testing sequence that was done was the one in which the control algorithm that deals with the movement in a four- legged configuration. For making this selection, the value of the signal req_command was set to the value 3. This value is set from the exterior of the model by a special module that has to deal with the task scheduling.

The algorithm for moving the snake structure in a wheel structure is composed from 4 major stages, which repeats as long as the movement of the robot is needed. These 4 commands that are given impose motor commands that will have to be moved with angular degrees of 30° and -30°. For these types of commands the following commands values that have to be given:

- for a 30° movement the value of the command that has to be sent to the output of the system is 40.
- for a -30° movement the value of the command that has to be sent to the output of the system is 20.

The result of the simulation for the type of algorithm which has for the req_command the value 3 (movement algorithm in four-legged structure) is presented in the following figure (see Fig. 10).

The result of the simulation for the type of reconfiguration shows that the commands that have to be given to the motor number 2 of the modules that form the legs (Module3, Module 4, Module5, Module6) respect the order and the type of the command that is mentioned in the algorithm.

The motor M2 from the fourth robotic module is the first motor that is actuated. It receives a command with the value 40 which means that it executes a 30° angular movement.

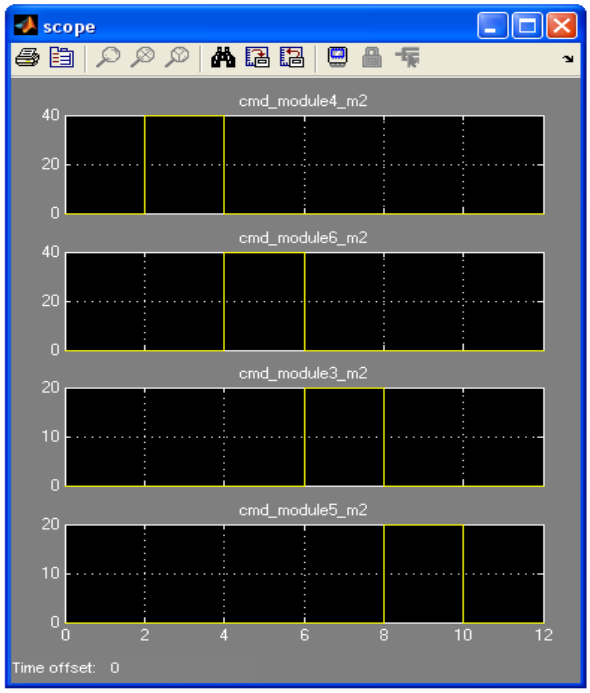


Fig. 10. Scope results in case of four-legged movement.

The last motor that is actuated is motor M2 from the fifth robotic module. This motor receives a command with the value 20 which means a -30° angular movement.

The timing of the command depends on the information that comes on the feed-back line and the sample time that is chosen (the system is considered to be a discrete one). The value of the sample time depends on the speed of the components that are used for implementing the system and the functional demand that the robotic system has to fulfill.

ACKNOWLEDGMENT

This research was developed in the framework of the Romanian project CEEX 91/2006 ROMAR - Miniature robotic system with self-reconfiguring and self-replicating skills.

REFERENCES

[1] K.J. Astrom, and B. Wittenmark, "Computer Controlled Systems", 3rd Edition, Prentice Hall, New Jersey,1997.
[2] T. Bräunl, "Embedded Robotics", Springer, 2003.
[3] A. Matsumoto, H. Asama, and Y. Ishida, "Communication in the autonomous and decentralized robot system ACTRESS", In: Proceeding of the IEEE International Workshop on Intelligent Robots and Systems, 1990, pages 835-840.
[4] Mathworks Inc. www.mathworks.com.
[5] A. Tanenbaum, "Computer Networks", 4th Edition, Prentice Hall, New Jersey, 2003.
[6] C. Nitu, B. Gramescu, S. Nitu, G. Bobocea, "Autonomous Modular Robotic Chain". Mecatronics 2008, 2008.
[7] M. Yim, "A reconfigurable modular robot with multiple modes of locomotion", In: Proceedings of the 1993 JSME Conference on Advanced Mechatronics, Tokyo, Japan,1993.
[8] http://www. robotics. ucv.ro/ceex91

Using a Virtualization Techniques – Based Platform for Advanced Studies on Operating Systems

Gabriel Rădulescu, Nicolae Paraschiv

***Abstract*—The virtualization technology allows several (sometimes critical) applications running on a single machine, but all isolated into virtual operating system images that do not interfere with each other. Such a working manner proves to be a revolutionary tool in computer science, especially when advanced studies on operating systems behavior are performed in order to reveal the hidden interactions in heterogeneous computing environments.**

I. Introduction

ALTHOUGH the virtual machine monitors (VMMs) theoretical principles were established about 40 years ago, focusing that time some stiff computing problems running on mainframes, at present they have a fresh and promising impact on computer technology, their application ranging from high-end servers to standard/entry-level platforms endowment.

Virtualization is the technique allowing multiple operating system instances running all at once by making use of only one hardware platform. Novel architectures and applications can be designed in order to benefit from such a modern approach in computing – especially when original, safe and reliable solutions have to be developed – and the perspectives are in a continuous spotlight.

In our previous works [1] and [2] the focus is put on how to practically build-up a platform based on virtualization techniques, designed as the main tool for studies on interactions between independent operating systems instances and a homogeneous hardware environment, when heterogeneous software platforms are involved. Since then, interesting results in this area were obtained, a brief selection being here presented.

II. A Short Overview On the Virtualization-Based Platform

A. Virtualization Principles

A VMM offers an abstract representation of one or more virtual machines (VM) by de-multiplexing the resources of a real hardware platform [3]. Such a VM may run a standard operating system (OS) together with its own native applications. Fig. 1 shows the classical architecture used by modern virtualization platforms (i.e. VMware, VirtualBox and VirtualPC solutions).

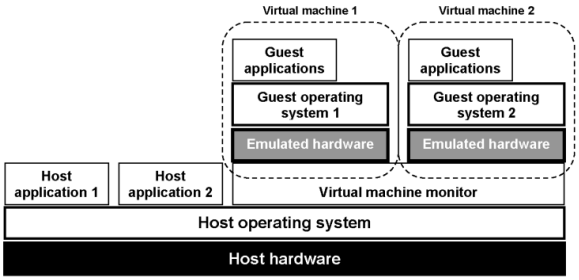


Fig. 1. The typical structure for VMM in the context of a host OS, providing the hardware abstraction layer for multiple VMs [2].

The software executed inside a VM is identified as *guest* (guest operating systems, guest applications), whereas the software running outside the virtual machine – typically the host platform operating system – represents the so-called *host software*.

The guest OS and its corresponding applications run in *user mode*, with hardly-limited control over the real hardware, but the VMM runs in the most privileged level (*kernel mode*), as the host OS in Fig. 1 is used to provide the basic access to physical devices for VMM and VMs [4]. The guest software uses the emulated hardware that is offered by VMM in the same transparent manner as it would do with real hardware. All complex interactions between guest software and the abstract hardware platform are trapped by the VMM and consequently emulated in software, allowing the guest OS to run in its standard way, also maintaining the strict control over the system at the VMM layer [1]. As previously shown in [2], by virtualization a perfect illusion of multiple, distinct virtual computers can be created, with separate operating systems and applications. For safely keeping the working environment, the VMM isolates each virtual computer and its emulated hardware through an adjustable redirection mechanism. The most easy-to-understand examples are, for instance, mapping a number of virtual disks to different zones of a physical disk or virtual memory mapping onto different pages in the real machine memory system.

The virtualization environments can be used not only for simply multiplexing the host resources, but also to add supplementary services to an existing system, including here special debuggers for new OSs, live machines migration [5],

Manuscript received May 5, 2010.

Gabriel Rădulescu is with the Petroleum-Gas University of Ploiesti, Automatic Control, Applied Informatics and Computers Department, 100680 Ploiești, Romania (e-mail: gabriel.radulescu@upg-ploiesti.ro).

Nicolae Paraschiv is also with the Petroleum-Gas University of Ploiesti, Automatic Control, Applied Informatics and Computers Department, 100680 Ploiești, Romania (e-mail: nparaschiv@upg-ploiesti.ro).

intrusion detection and prevention [6] as well as code integrity test [7]. It is also very important to mention that some of these services are implemented outside the guest machines and, as consequence, they do not affect at all the guest environment.

In this context, the virtual machine introspection (VMI) is represented by all technologies used to interpret and modify the inner states and events within the guest [8], [9]. By using VMI methods, the variables and guest memory addresses are translated (after reading the guest OS and applications symbols/pages tables) into real references for the host OS. More, through hardware and software breakpoints, a specific VM service is allowed to gain full control at a specific instruction address. Finally, by this technique, such a service may invoke the guest operating system or application code in order to make use of general functions (for instance reading a file from the guest OS file system). It is also very important to emphasize that all virtual machine services can be protected by disabling external I/O procedures. In the same time, VMs assure the guest data integrity by introducing the *snapshots* (restore points), which points to a frozen guest image where the system can be rolled back (after a crash, for example).

A special VMM application which deals with concurrent OS interactions is the so-called *virtual machine based rootkit* (VMBR), which monitors the VMs activity. The target (supervised) VM practically sees no difference in its memory space, disk availability or execution (depending on the virtualization quality) [1], [2]. There is a complete isolation between the event-generating host OSs and the targeted systems, so the software in the target system cannot see or modify the interacting software from the other system [8]. Also, apart from monitoring the target states and events, the VMBR can quietly read and modify them (without being observed from inside of the running VMs), as it fully controls the virtual hardware presented to the guests [1].

B. The Platform Architecture

As previously presented in [2], the authors have designed and implemented a robust, flexible and multi-client oriented platform at the "Petroleum-Gas" University of Ploieşti (Computers and Networks Laboratory) which is a valuable environment for studying the OS complex behavior. This section outlines its main hardware and software characteristics.

1) The hardware architecture

As shown in Fig. 2, the system is distributed over a high speed Local Area Network (LAN) and consists in one main server and the associated workstation-clients.

The central node is a HP Proliant ML310 server with RAID storage system and backup facility in order to prevent any loss of the user data. As mention, the server has proved an extraordinary robustness, with almost inexistent downtime up to now (in the case, no time required for service as no failure occurred during operating sessions for over 3 years).

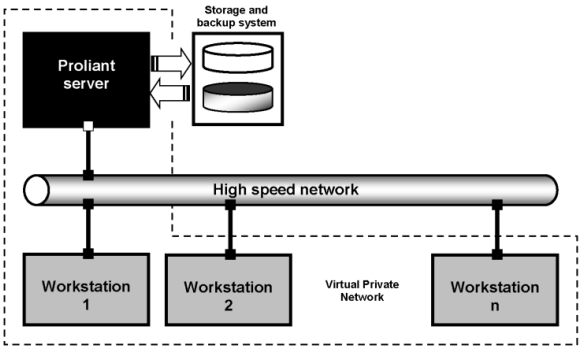


Fig. 2. A schematic representation for the system's hardware architecture [2].

The system's clients are located on independent workstations connected to the Proliant server through a high-speed managed Ethernet switch. While at the beginning all workstations had the same configuration, later we have used to run our experiments by targeting other hardware configurations (in order to extend the research results, if possible). The quality of service (QOS) inside the LAN was highly improved by configuring a dedicated virtual LAN (VLAN), which isolates the distributed system from the corporate network (at the University level) [2].

2) The software configuration

At first, in order to improve the compatibility between clients and server, the uniform OS installation was adopted (a very stable Linux SuSE distribution, both on server and client machines). But as the last research is focused on heterogeneous host OSs study, other Linux distributions (i.e. Fedora, Ubuntu) and Windows-class OSs are currently tested on workstations.

As suggested by Fig. 3, a uniform user account management was adopted too, by locating the users' database and home directories on the Proliant server, all clients being authenticated via NIS (Network Information Service), whereas the storage resources are exported via NFS (Networking File System). Except the *root*, any other user has a mobile profile, with a homogeneous way of accessing the system and server resources. Apart being completely transparent for the users, this solution offers an increased data safety, as any crash at the workstations level only interrupts the communication with the server, without affecting the data last saved here over the network. The good protection against malicious code (spywares, backdoors, viruses) is easily implemented because, by locating the user files only on server side, the central management of any security/integrity solution is natively allowed [1], [2].

As virtualization platform, VMware proved to be the most flexible/reliable solution, with a high level of the hardware abstracting, as well as an almost perfect compatibility with all guest OSs we have included in the test sessions (Windows-based systems, different Linux distributions, even the native MS DOS 6.22). On this infrastructure we currently run software compatibility and endurance tests with VMware Workstation, with versions ranging from 4.5.x to 7.0.x.

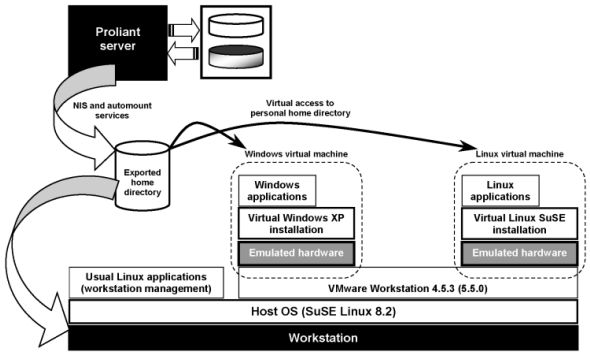


Fig. 3. The simplified software architecture [2].

3) Advantages

The major advantages when using such a multi-client virtualization platform were extensively presented in [2], so in this general outline only a few of them are emphasized:

--The possibility to pack and distribute software pieces in virtual machines with considerable less effort.

--The facility of virtual machines replication on all workstations ("instantaneous" software update).

--The "undo" at shutdown feature, for all changes inside virtual machine instances.

--Reduced time for technical interventions, network tuning and maintenance activities.

But the most important advantage is the open system architecture, from many different points of view. For instance, all network clients can be considered as identical, so adding a new workstation in the system is trivially easy. On the other side, at logical level, all VMs on every workstation have the same attributes, no matter the installed guest OS, being seen by the local VMM as part of a homogeneous environment. At last, any system upgrade may be performed, as time as the architecture and its functionality are preserved [2].

III. Case-Study: OS Regular Activity Monitoring

In order to have a complete image over the platform utility, here are presented some significant results when monitoring different OSs running a usual processor-test (a hard-encoded MPEG video file playback) This short case-study is taken from a more complex research (which includes, for instance, a performance improving analysis), but this will be subject of a future work..

A. The Test Configuration

Such a procedure may not give interesting results when the hardware resources are generous (because the processing power is enough to handle without problems the software decoding process), this being the reason why the authors performed all tests on simulated limited-power architectures.

As host machine the authors used a medium-performance platform based on Intel Celeron D352 processor, with 1GB RAM and SATA2 HDD, with dual boot feature (SuSE Linux and Windows XP Pro), integrated as client for the previously described network.

The software solution adopted for VMM was VMware Workstation 6.0 (running both under Linux and Windows host machines) coupled with Veeam Monitor analyzing tool. This configuration allows a perfect integration with the VMware virtualization layer, so the platform can read and present a wide range of operating parameters related to VMs – kernel errors/traps, CPU, memory, disk, network and pagefile usage statistics. In this hosting context, three identical virtual machines were created (each emulating an Intel Celeron platform with 256MB RAM and IDE HDD storage) and configured with typical installations of SuSE Linux 8.2, Ubuntu 7.2 and Windows XP Professional. In order to have a complete functional environment, all VMs include VMware Tools platform. All OS images were stored on the Proliant server and NFS exported over the network.

B. The Testing Scenario and Results

In the context of normal computer use, dealing with multimedia applications is also a regular task. But when the file to be played is hard-encoded (i.e. high bitrates/video resolutions), the CPU usage may become problematic if the processor performances are below a specific limit, so it is of high interest to know how an OS acts in order to keep the entire system in acceptable working state.

The tests performed by authors consisted in playing a local MPEG4 video file encoded with high bitrate (approx. 10 Mbps) by using the same version (1.0) of the famous MPlayer installed under SuSE Linux, Ubuntu and Windows XP VMs. There were two main scenarios, involving both separate tests (consecutively performed) and simultaneous (on all VMs in the same time), the results being presented and briefly commented here.

1) The 3 minutes testing session results (separate run)

Fig. 4 shows a comparison between CPU usage percent for the above mentioned OSs. By analyzing these graphs it is to mention that, although at the beginning in SuSE OS the CPU is used not more than 50%, after a few seconds the percent raises to a maximum of 85%. The Ubuntu machine acts totally different, with an initial peak of 92% processor usage but followed by a moderate behavior on long term (with lots of inverse peaks to 50%). The best response is obtained when using Windows XP, which acts between maximum 82% and minimum 31% CPU usage. In fact, this behavior is in total agreement with expectations, as time as high activity means lots of system calls (and for Linux OSs the rendering modules were compiled and integrated in kernel, unlike in Windows case).

The HDD usage during the playback process is depicted in Fig. 5. The SuSE Linux VM shows a uniform-time disc access, with long periods of inactivity (approx. 30 s) and high peaks between 8 and 22 Mbytes/s (read/write access). The same uniform behavior can be observed for Ubuntu, at shorter intervals (about 6 s), through moderate peaks (810 to 9216 Kbytes/s). Windows XP is non-predictable, having an almost continuous HDD access with maximum peaks of

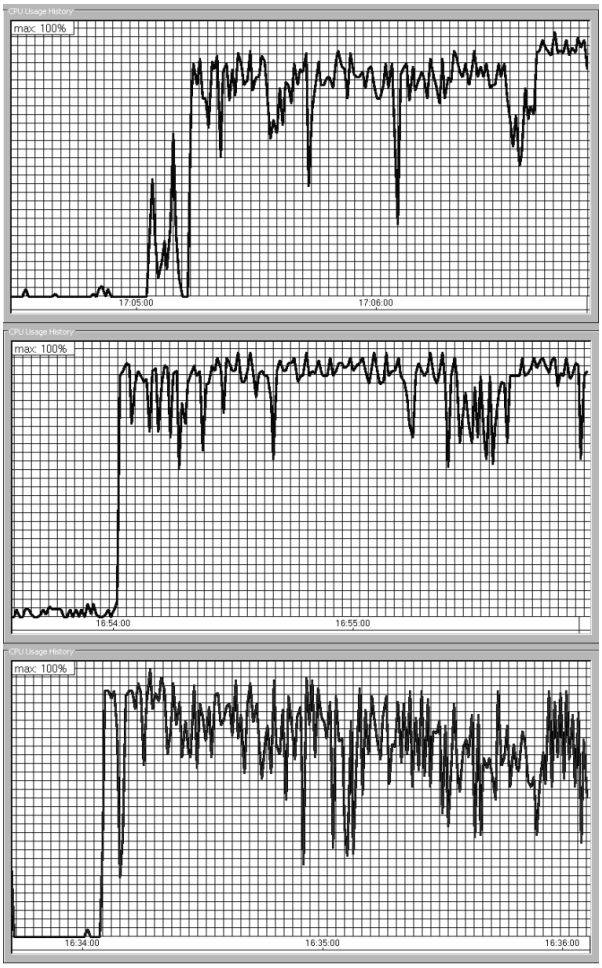


Fig. 4. CPU usage profiles for SuSE Linux (top), Ubuntu (middle) and Windows XP (bottom) – scenario 1.

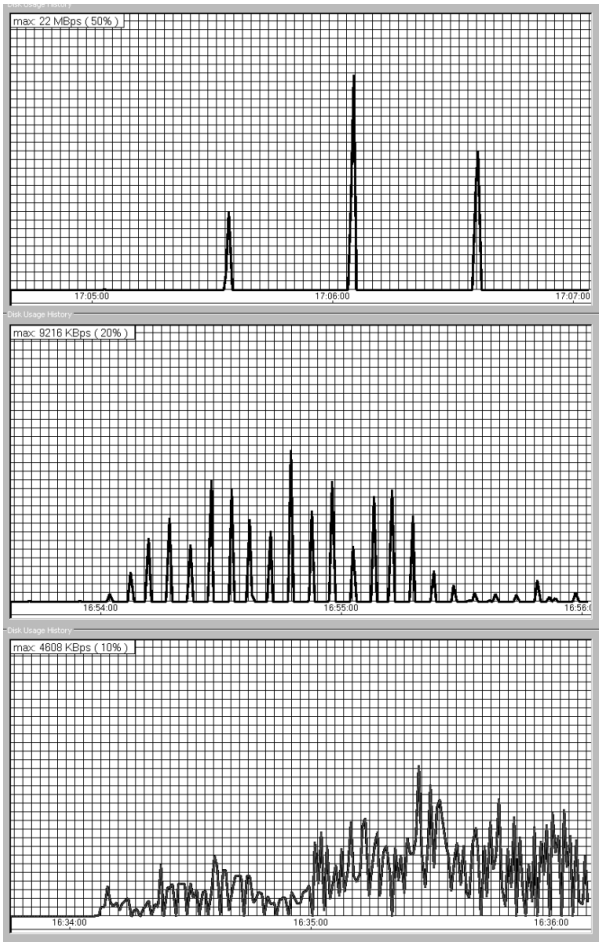


Fig. 5. Disk usage profiles for SuSE Linux (top), Ubuntu (middle) and Windows XP (bottom) – scenario 1.

4608 Kbytes/s (which may be not an optimal behavior).

As remark, the diagrams in Fig. 5 (and also in the following figures) are differently scaled for better details readability.

The authors further investigated this case and found that such an effect is induced by a different pagefile management (disc cache) style for the three OSs above mentioned.

SuSE Linux has an intense but shorter action on its swap partition (which keeps the pages ready to send into memory, respectively the dumped-to-file memory content) than Windows XP, although both have the same maximum access peak (2560 Kbytes/s), as seen in Fig. 6. During the last minute, for about 40 s, SuSE does not even use its disc cache, whereas Windows still keeps accessing the pagefile until the MPEG file playback finishes. Such a policy of *take it as needed* may be convenient when the hardware resources are continuously available during the working session, but may put serious problems when the limits are touched. Regarding Ubuntu, it has an almost inexistent swapping effect (maximum 512 Kbytes/s transfer speed) because it frequently access the HDD when reading small parts of the MPEG file and they seem to always fit into the free memory, with no need to swap memory pages.

2) The 3 minutes testing session results (concurrent run)

As shown above, the second scenario was based on simultaneously performing the video playback in all VMs, in order to reveal if the VMM has a good policy when trying to honestly allocate resources for the managed virtual computers.

In order to distinguish the stabilizing time at start, the MPlayer sessions were launched with short delays between them (approx. 3 s), first under Windows XP, Ubuntu and finally in the SuSE Linux VM, as Fig. 7 depicts.

It can be observed that, for all OS instances, after a short transient time the CPU usage practically follows the same average envelope, meaning that VMM (VMware) succeeds to fairly implement the resources allocation. It is also to remark the different profile shapes in comparison with the ones from Fig. 4. A careful look at Fig. 7 shows that, when – for instance – the Ubuntu machine CPU usage lowers, the remaining computing power is re-allocated to SuSE and Windows VMs, this way the overall efficiency being increased. There is no surprise when analyzing the HDD usage in this new context. Indeed, the main characteristics observed when performing the first scenario are preserved, excepting the maximum peak value which lowers to 9216

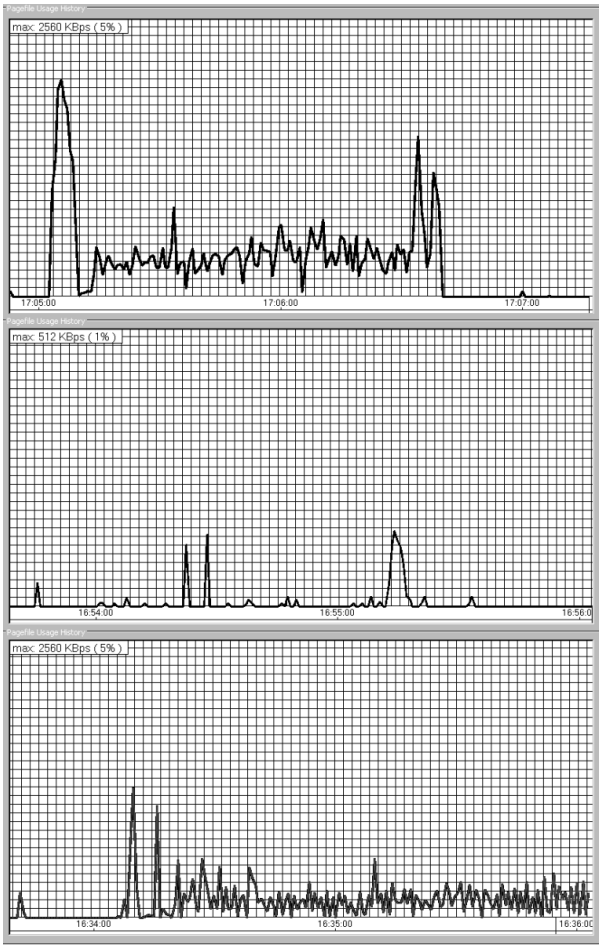


Fig. 6. Swap/pagefile usage profiles for SuSE Linux (top), Ubuntu (middle) and Windows XP (bottom) – scenario 1.

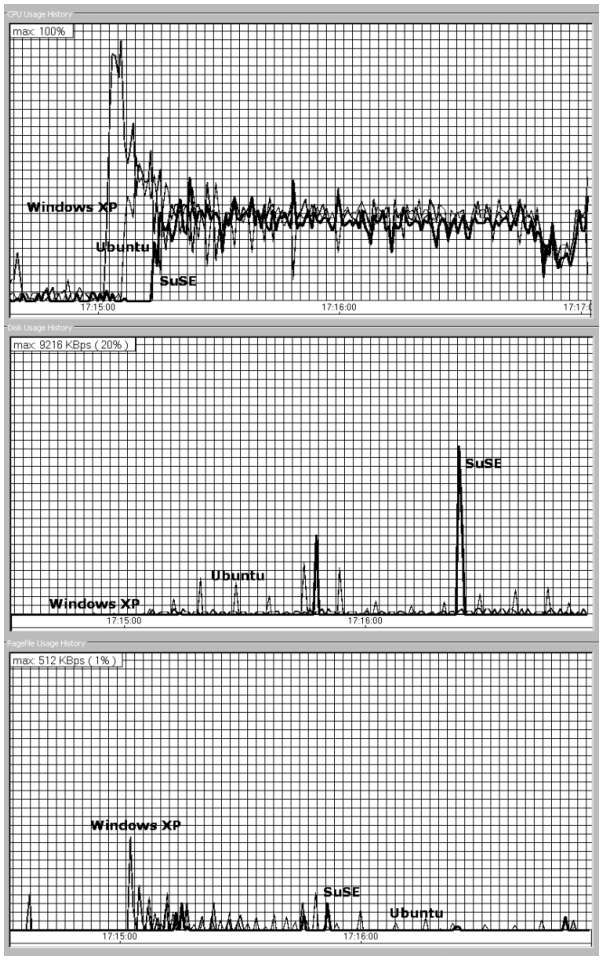


Fig. 7. CPU (top), HDD (middle) and swap/pagefile (bottom) usage profiles for SuSE Linux, Ubuntu and Windows XP VMs running simultaneously (scenario 2).

Kbytes/s (instead of 22 Mbytes/s) for the SuSE virtual machine. In a similar way, the memory swapping (pagefile access) seems to differ only by the highest peak value (512 Kbytes/s instead of 2560 Kbytes/s – as for the first scenario).

IV. Conclusion

The complex hardware/software virtualization technology allows new and revolutionary approaches in computer science, especially in the field of operating systems practical studies. In this context, the present paper presented some selected results over the complex interactions between multiple operating systems (SuSE Linux, Ubuntu, Windows XP) and a homogeneous virtualized hardware platform. The studies were performed by using the non-invasive monitoring technique of VMBR. The behavioral characteristics of the three mentioned OSs are also outlined.

From the authors' point of view, extending such a research approach has a true practical relevance, as time as it may be used to reach a perfect fit between user's needs and the software environment particularities.

References

[1] G. Rădulescu, N. Paraschiv, "Virtualization techniques in computer science – a practical point of view", *Proc. of the UNIVERSITARIA SIMPRO 2006 (Petroşani) Symposium*, pp. 41-44, 2006.

[2] G. Rădulescu, N. Paraschiv, "Developing a virtualization techniques – based platform for advanced studies on operating systems", *The Petroleum-Gas Univ. of Ploieşti Bulletin*, Vol. LIX, Technical Series, No. 3, pp. 1-6, 2007.

[3] 5. R. P. Goldberg, "Survey of virtual machine research", *IEEE Computer*, pp. 34–45, June 1974.

[4] 9. J. Sugerman, J., G. Venkitachalam , B. H. Lim, "Virtualizing I/O devices on VMware Workstation's hosted virtual machine monitor", *Proc. of the 2001 USENIX Technical Conference*, June 2001.

[5] 8. C. P. Sapuntzakis *et al.*, "Optimizing the migration of virtual computers", *Proc. of the 2002 Symposium on Operating Systems Design and Implementation*, December 2002.

[6] 1. K. G. Anagnostakis *et al*., "Detecting targeted attacks using shadow honeypots". *Proc. of the 2004 USENIX Security Symposium*, August 2005.

[7] O. Agensen, D. Detlefs, "Mixed-mode bytecode execution", *Technical Report SMLI TR-200-87*, Sun Microsystems, Inc., 2000.

[8] S. T. King, G. W. Dunlap, P. M. Chen, "Debugging operating systems with time-traveling virtual machines", *Proceedings of the USENIX Annual Technical Conference (USENIX'05)*, pp. 1-15, April 2005.

[9] A. Whitaker, R. S. Cox, S. D. Gribble, "Configuration debugging as search: finding the needle in the haystack", *Proc. of the 2004 OSDI Symposium*, December 2004.

Monotone and slope restricted nonlinearities - a PIO II case study

Vladimir Răsvan, Dan Popescu and Daniela Danciu

***Abstract*— In this paper there is presented a rather straightforward application of the absolute stability frequency domain inequalities to the practical problem of PIO (Pilot In-the-loop Oscillations) proneness of aircrafts. An extended to critical cases version of the Yakubovich criterion for the case of slope restricted nonlinearities is applied to the benchmark case of the X15 landing flare incident.**

I. PROBLEM STATEMENT AND STATE OF THE ART

This paper has two starting points.

A. The first one is theoretical and arises from the theory of absolute stability. A brief reminder, with reference to Fig. 1, *defines absolute stability as global asymptotic stability of the zero equilibrium of the system with the feedback structure there, this global asymptotic stability being valid for the entire class of systems defined (induced) by the class of nonlinear functions describing the nonlinear block of Fig. 1*.

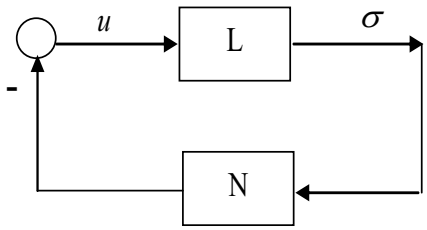


Fig. 1. Absolute stability feedback structure.

It should be added to this definition that the absolute stability conditions have to be expressed in the language of the linear sub-system and use information concerning the class of the nonlinearities (i.e. no specific nonlinearity of the class should be involved). An example at hand of these assertions is given below: if the nonlinear element is described by a function $\varphi : \mathbb{R} \mapsto \mathbb{R}$ such that

$$0 \leq \varphi(\sigma)\sigma \leq \overline{\varphi}\sigma^2 \tag{1}$$

and the linear part by a strictly proper irreducible transfer function with its poles in $\mathbb{C}^-$, the Popov frequency domain inequality for absolute stability reads

This work has been supported by the Research Project CNCSIS ID-95 of the Romanian Council for University Research.

The authors are with Department of Automatic Control, University of Craiova, A.I. Cuza, 13, Craiova, RO-200585,Romania. {vrasvan,dpopescu,daniela}@automation.ucv.ro

$$\frac{1}{\overline{\varphi}} + Re(1 + \jmath\omega\theta)H(\jmath\omega) > 0, \quad \forall\omega \in \mathbb{R}_+ \tag{2}$$

for some suitably chosen real θ.

As in most stability problems for nonlinear systems, the stability conditions are, generally speaking, only sufficient; the gap between them and the (possible) necessary and sufficient conditions, is called *method's "conservatism"*.

Aiming to reduce the "conservatism" suggested the researchers to impose additional restrictions on the nonlinear functions (by considering monotonic, odd monotonic, slope restricted etc nonlinearities). The general framework of this approach is summarized in the so-called Integral Quadratic Constraints (IQC) method as defined in the pioneering paper of Yakubovich [1], developed, among others, in [2] and given the actual form in [3]. The idea is as follows: more restrictions are imposed on the nonlinear part of the system, less restrictive are the conditions on the linear part yielded by the frequency domain inequality or by the equivalent to it Liapunov function. In practice this means that the frequency domain inequality will contain more free parameters to choose in the necessary way, however this does not make the inequality easier to manipulate. Here also a trade-off appears as necessary: its significance is a necessary limitation of the number of IQCs that are considered in a specific problem and we shall deal with this fact in the following.

One of the oldest additional restrictions is the slope restriction deduced from the fact that the linear functions are both sector and slope restricted in the same sector. Starting with the suggestions of Kalman [4], the results have been obtained by Yakubovich [5], [6], extended to the case of several nonlinear elements in [7] and to the case of hysteresis nonlinearities in [8]. All these papers take into account both sector and slope restrictions.

The note [9] aims to take into account the above mentioned trade-off by proposing "a stability criterion incorporating only the slope restrictions about the nonlinear function". The result was interesting but the proofs - far from convincing. In order to make the result credible several papers followed [10], [11], [12], [13], the reported criteria containing various restrictions concerning the linear part, the nonlinear part or the overall closed loop system. The presence of the restrictions witnessed about rigorous approaches but made more narrow the class of applications.

Consequently a comparison of all these approaches appears as both useful and necessary.

B. The second starting point is more application oriented. It is connected with PIO - P(ilot)-I(n the loop)-O(scillations),

a rather complex phenomenon which can nevertheless be viewed as described by self-sustained oscillations of a feedback structure where the airframe dynamics represents the controlled object while the pilot dynamics acts for the controller. It is a well know fact now [14] that there are three categories of PIO. They are defined as quasi-linear pilot-vehicle system oscillations except that series rate or position limiters are involved. At the physical level there are described by stating that "rate limiting", either as a series element or as a rate-limited surface actuator, modifies the Category I situation by adding an amplitude-dependent lag and by setting the limit cycle amplitude" [15].

The saturation is in fact the basic nonlinear function of a rate limiter which is described as in Fig. 2.

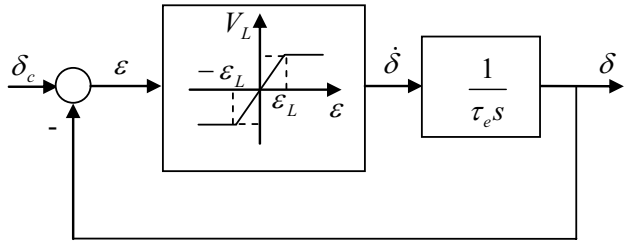


Fig. 2. Rate limiter

When incorporated in the PIO II structure the rate limiter will produce the diagram of Fig. 3 which at its turn may be reduced to the feedback loop of Fig. 1. The nonlinear block will be the saturation function described by

$$f(\varepsilon) \quad = \quad \begin{cases} V_L \text{sgn}\, \varepsilon \ , \ |\varepsilon| \geq \varepsilon_L \\ \dfrac{V_L}{\varepsilon_L} \varepsilon \ , \ |\varepsilon| < \varepsilon_L \end{cases} \tag{3}$$

This nonlinear function is: sector restricted, monotone and slope restricted.

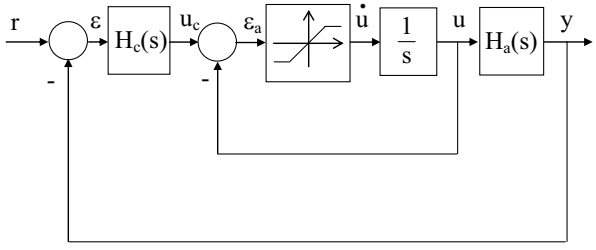


Fig. 3. Feedback structure with rate limiter

The resulting transfer function of the equivalent linear block will be, as expected

$$H(s) = \frac{1}{s} H_c(s) H_a(s) \tag{4}$$

The conclusion of this short exposition is that PIO II can be approached by absolute stability techniques.

C. We already gave some hints concerning the absolute stability criteria in the previous consideration. In a pioneering paper of Yakubovich [1] it was shown that the properties of the nonlinear function may be expressed as some quadratic constraints (local or integral) on the input and output of the nonlinear block of Fig. 1; that paper is full of examples of association of such quadratic constraints. The approach was continued in [2] and become extremely popular in our days due to the contributions in [3] where the quadratic constraints where considered for frequency domain also - see also [16]. For our purposes it is important mentioning [17] where the rate limiters are considered in the new context.

The achieved results correspond to the normal trends of the field of absolute stability: more information we have on the nonlinear subsystems i.e. more quadratic constraints are at our disposal, more free parameters are contained in the frequency domain inequality(ies). From here one obtains, again in a natural logic, that the sufficient conditions for stability thus obtained may be closer to the necessary and sufficient ones (less "conservative"). The counterpart is that more free parameters are, more difficult to manipulate (both analytically and numerically).

For these reasons the multi-parameter frequency domain stability inequalities are used with due caution: for instance, the Yakubovich type criterion for systems with monotonic nonlinearities [5], [6] is easier to cope with than the Zames-Falb criteria [18]. This assertion will be tested on a PIO II model.

In order to end this introductory part we shall give here our methodology of applying the frequency domain inequalities. It is based on "parsimony principle" i.e. to have as few free parameters as possible in the frequency domain inequality to recover as much as possible from the linear stability sector for the nonlinearity one ("reducing the conservatism" i.e. the gap between sufficient and necessary and sufficient conditions for stability in the nonlinear case).

II. THE STABILITY INEQUALITIES FOR SLOPE RESTRICTED NONLINEARITIES

A. The oldest frequency domain inequality for sector and slope restricted nonlinearities appears in the papers of Yakubovich [5], [6]. If besides (1), the following slope restrictions are observed

$$\underline{\nu} < \varphi'(\sigma) < \overline{\nu} \tag{5}$$

and also the linear block is stable then the frequency domain inequality is

$$\begin{aligned} &\tau_1 \left(\tfrac{1}{\overline{\varphi}} + \Re e H(\jmath\omega) \right) + \tau_2 \Re e \left(\jmath\omega H(\jmath\omega) \right) + \\ &+ \tau_3 \omega^2 \Re e \left(1 + \underline{\nu} H(\jmath\omega) \right)^* \left(1 + \overline{\nu} H(\jmath\omega) \right) \geq 0 \ , \ \forall \omega \in \mathbb{R} \end{aligned} \tag{6}$$

for some freely chosen $\tau_1 > 0$, $\tau_2 \in \mathbb{R}$, $\tau_3 \geq 0$.

For hysteresis-like nonlinearities the sign of τ_2 depends on some computed parameter ψ whose sign is determined by the sense on the hysteresis loop: it is required that $\tau_2 \psi \leq 0$ [8].

If only slope restrictions are taken into account, then we have to take $\tau_1 = 0$ (thus eliminating the possibility of S-procedure). In this way we are closer to the cases of [9], [12] or [11]. Requiring, additionally

$$(1 + \underline{\nu} H(0))(1 + \overline{\nu} H(0)) > 0 \tag{7}$$

we may write that (6) is equivalent to

$$\frac{\tau_3}{\nu_2} + Re\left[\left(\tau_3(1 + \underline{\nu}/\overline{\nu}) - \frac{\tau_2}{\overline{\nu}}\frac{1}{j\omega}\right) H(j\omega) + \right. \\ \left. + (\tau_3 \underline{\nu})|H(j\omega)|^2\right] \geq 0 \tag{8}$$

and this is exactly the condition in [8] or in [11]; note that in these papers the matrix A need not be Hurwitz but only hyperbolic.

It is thus obvious that the first frequency condition (6) is also the most general. Moreover, even the proof is performed in the least restrictive assumptions among all these methods.

B. Examine now conditions (6) or (8) which contain both the term $\underline{\nu}|H(j\omega)|^2$. Therefore, if $\underline{\nu} < 0$ this term is damaging the frequency domain inequality. But $\underline{\nu} \geq 0$ means that the nonlinear function is also monotonically increasing and, if $\overline{\nu} < +\infty$, globally Lipschitz. These properties are significant for the existence of forced oscillations. On the other hand monotonicity will allow introduction of the Zames-Falb or Brockett-Willems multipliers which might be helpful in certain frequency domain characteristics. We give below, for the sake of completeness, the Brockett-Willems criterion, reproduced after [19]. Assume that besides (1) $\varphi(\sigma)$ is monotone increasing e.g. (5) holds for $\underline{\nu} = 0$ and $\overline{\nu}$ arbitrary; then the frequency domain inequality is as below

$$\frac{1}{\overline{\varphi}} + ReZ(j\omega)H(j\omega) \geq 0 \ , \ \forall \omega \in \mathbb{R} \tag{9}$$

where

$$Z(s) = 1 + s\theta + \sum_1^p \delta_j \frac{s + \rho_j}{s + \rho_j + \rho_j'} \tag{10}$$

where $\theta \geq 0$, $\delta_j > 0$, $\rho_j \geq 0$, $\rho_j' > 0$, $j = 1, ..., p$ are freely chosen parameters. Clearly we have here a typical example of a criterion with many free parameters and this does not make it easier to manipulate. The choice of all these parameters is connected with circuit synthesis since (9) is the expression of the condition of positive realness requirement which is basic in circuit synthesis; also $Z(s)$ may be viewed as a phase correction to ensure positive realness.

On the other hand the multiplier defined by (10) contains the non-causal part $1 + \theta s$ which is exactly the Popov multiplier; the additional term, being proper, is causal. The fact led other researchers to consider general non-causal multipliers: this is the case of the Zames-Falb multiplier [20], also of other ones [21], [22]

III. A CASE STUDY - THE X-15 LANDING FLARE PIO

This aviation incident occurred on June 8, 1959 and, during the following half-century, became some kind of benchmark problem even in the further accumulation of PIO databases. A useful while concise description may be found in [15]. It appears that the basic block diagram of Fig 3 is applicable here and the flight data show that in that unpowered glide flown the pilot was compliant hence $H_c(s) \equiv K_p$. Also the flight data show that the actuator was operating in the highly saturated region; moreover all subsequent reference showed that "not only do all of the applied Category I criteria indicate that the X-15 would not be susceptible to PIO but also the aircraft was found to be level 1 for most of the applied handling quality measures"; moreover, the experimental data show that the instability frequency for the linear system with a synchronous pilot loop closure is 5.31 rad/sec, while the observed PIO frequency was 3.3 rad/sec.

A. We have thus the case of an obvious PIO II case. This case will be analyzed using the frequency domain inequalities discussed above. According to [23] we shall have

$$H_c(s) = \frac{3.476(s + 0.0292)(s + 0.883)}{(s^2 + 0.019s + 0.01)(s^2 + 0.8418s + 5.29)} \tag{11}$$

The time constant of the actuator is 0.04 sec. Since the slope of the saturable actuator equals 1 for unsaturated case, the open loop transfer function of the linear (unsaturated) system of Fig. 3 is

$$H_b(s) = \frac{K}{(s + 25)} H_c(s) \tag{12}$$

where $K = 25 \cdot K_p$

A standard technique would be to compute root locus for the above open loop transfer function.

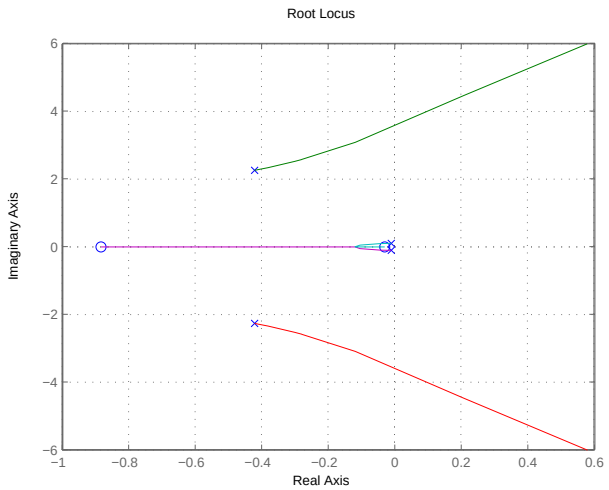


Fig. 4. Root locus for linear X15 PIO I model

We shall discuss in brief the root loci of fig.4. Clearly the far left real pole goes to $-\infty$ on the real axis; the starting dominant complex poles $-0.0095 \pm 0.1\jmath$ will approach asymptotically the two real zeros -0.0292 and -0.883. The other starting complex poles $-0.4207 \pm 2.3\jmath$, while not being dominant, migrate to the RHP thus generating *oscillatory instability*; this instability corresponds to the imaginary poles $\pm 3.551\jmath$ and the pilot gain $K_p = 2.28$. It is higher than the value prescribed by [23] which was 2.0425. On the other hand the frequency 3.551 is lower than 5.31 reported by [23] but closer to the PIO II frequency 3.3 reported by [15]. The only explanation at hand is that the cited references are using experimental data and there is some mismatch between real data and the reported model.

B. We shall turn now to the application of the frequency domain inequalities for the following transfer function of the linear part

$$H_L(s) = \frac{(s+0.0292)(s+0.883)}{s(s^2+0.019s+0.01)(s^2+0.8418s+5.29)} \tag{13}$$

Clearly we are here in the first critical case since we have a simple pole at $s = 0$.

The Popov frequency domain inequality for (13) means

$$\frac{1}{k} + \Re e\ (1+\jmath\omega\theta)H_L(j\omega) \geq 0 \tag{14}$$

for some $\theta \geq 0$; note that in the usual graphical interpretation of (14) we have to introduce the Popov locus by

$$X_P(\omega) = \Re e\ H_L(\jmath\omega)\ ,\ Y_P(\omega) = \omega \Im m\ H_L(\jmath\omega) \tag{15}$$

thus obtaining

$$\frac{1}{k} + X_P(\omega) - \theta Y_P(\omega) \geq 0\ ,\ \forall \omega \in \mathbb{R} \tag{16}$$

The Yakubovich frequency domain inequality (6) will be considered for globally Lipschitz functions satisfying

$$0 \leq \frac{\varphi(\sigma_1) - \varphi(\sigma_2)}{\sigma_1 - \sigma_2} \leq K \tag{17}$$

with $K > 0$ as above. Since this means $\underline{\nu} = 0$, $\overline{\nu} = \overline{\varphi} = K$ the frequency domain inequality (6) becomes [24]

$$\Re e\ (1+\jmath\omega\theta+\gamma\omega^2)\left(\frac{1}{K} + H_L(\jmath\omega)\right) \geq 0 \tag{18}$$

This inequality also may be written as

$$\frac{1}{K} + X_Y(\omega) - \theta Y_Y(\omega) \geq 0 \tag{19}$$

where the modified transfer loci family (with respect to the parameter $\gamma > 0$) is defined by

$$X_Y(\omega) = \Re e\ H_L(\jmath\omega)\ ,\ Y_Y(\omega) = \frac{\omega}{1+\gamma\omega^2}\ \Im m\ H_L(\jmath\omega) \tag{20}$$

In order to judge sharpness of the frequency domain criteria for absolute stability, we turned to the *Aizerman problem* in the given case: to find the maximal sector in the linear case, the corresponding frequency of the oscillatory instability and, then, make a comparison to the nonlinear case. With respect to this, the root locus for $H_L(s)$ given by (13) and announcing, as mentioned, the first critical case, gave $K = 3.818$ and the oscillatory instability at $\omega = 2.13$.

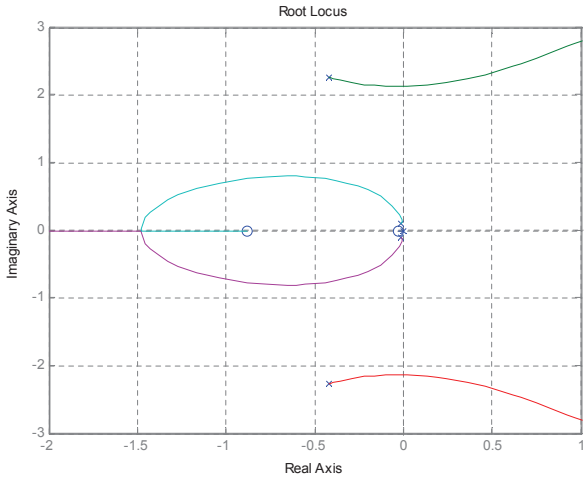


Fig. 5. Root locus for linearized X15 PIO II model

We next applied the frequency domain inequalities. According to the "parsimony principle" which here means that we start with a minimum number of free parameters in the inequality, there was considered first the Popov inequality; as it may be seen in fig. 6, the chosen scale required a zoom at relatively "high" frequencies and it clearly appears as obvious that the answer to the Aizerman problem would be negative i.e. we shall have $K < 3.818$; more precisely $K_{max} \approx 1/7.5$.

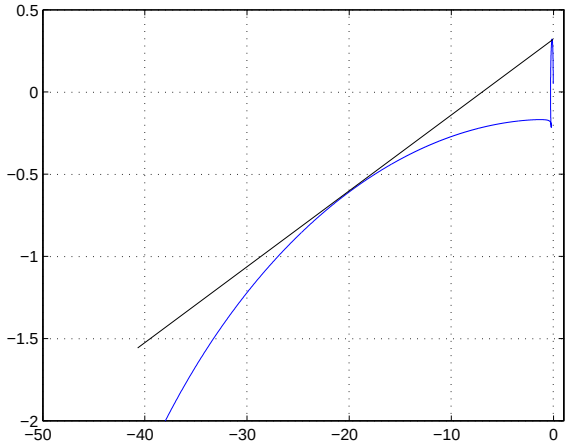


Fig. 6. Popov locus for X15 PIO II model

Therefore we turned to the Yakubovich criterion. The

factor $(1+\gamma\omega^2)^{-1}$ will attenuate the peak of fig.6 - diagram that corresponds to $\gamma = 0$: larger is $\gamma > 0$, lower is the peak - see fig.7,fig.8, corresponding to $\gamma = 0.1$ and $\gamma = 0.5$ respectively

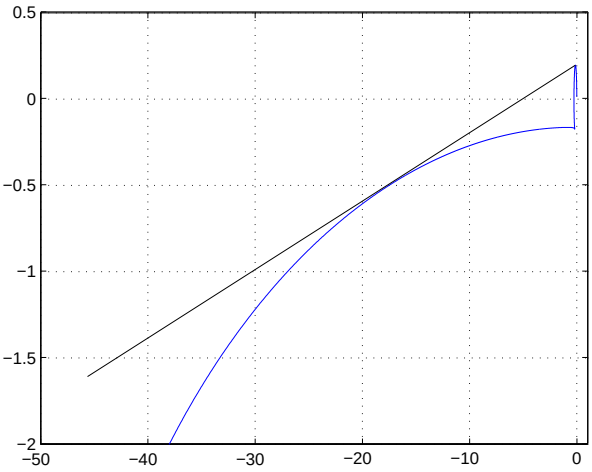


Fig. 7. Yakubovich locus for X15 PIO II model ($\gamma = 0.1$)

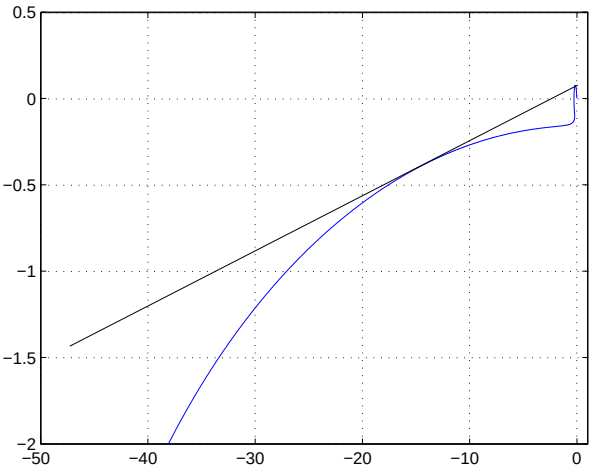


Fig. 8. Yakubovich locus for X15 PIO II model ($\gamma = 0.5$)

Observe from fig.9 corresponding to $\gamma = 1$ that the peak almost disappeared and this shifted the Popov Yakubovich line crossing to the right of the real axis, thus increasing the admissible K_{max}; here $K_{max} \approx 3$ but graphics errors may be suspected. However, the comparison to $K_{max} \approx 1/7.5$ speaks for the improvement due to the application of the Yakubovich criterion.

IV. CONCLUSIONS AND PERSPECTIVE

The research displayed in this paper represents the application to a real case analysis of the philosophy that considers PIO I and PIO II from the same point of view - that of

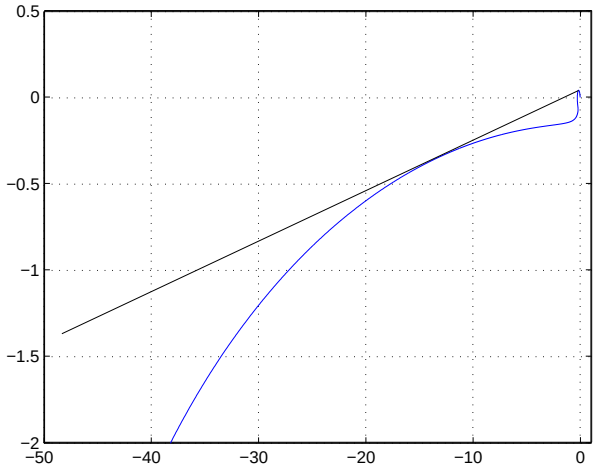


Fig. 9. Yakubovich locus for X15 PIO II model ($\gamma = 1$)

the absolute stability and of the Aizerman problem. This underlying idea of the present research (and not of the only one) is, as mentioned in the introduction, to make use of the fact that in PIO II like in PIO I the airframe and pilot models are linear but the position and rate limiters are activated hence at least one saturation nonlinearity is involved. The saturation is a "weak" nonlinearity: sector restricted, monotone, non-decreasing, piecewise differentiable. Consequently, discussion of PIO proneness implies an absolute stability problem and unitary treatment of PIO I and PIO II sends to the Aizerman problem, where the linear stability (Hurwitz) and absolute stability sectors are put in comparison.

From the methodology point of view this requires implication of the absolute stability tools. Since all aircraft databases for PIO analysis use frequency domain characteristics, it is only natural to use frequency domain inequalities. A case study for X15 landing flare incident showed the role of the criteria with many parameters e.g. the criterion of Yakubovich for slope restricted nonlinearity to reduce the gap between PIO I and PIO II estimates. Worth mentioning that we use a version of the Yakubovich criterion extended to critical cases since $H_L(s)$ in (13) had a simple pole at $s = 0$.

The encouraging results suggest that the research should continue.

REFERENCES

[1] V. A. Yakubovich, "Frequency domain for the absolute stability of systems containing several nonlinear or linear time invariant blocks (in russian)," *Avtomat. i Telemekhanika*, vol. 31, pp. 5–30, June 1967.

[2] V. Răsvan, *Absolute stability of time lag control systems (in Romanian)*, 1st ed. Bucharest: Editura Academiei, 1975.

[3] A. Megretski and A. Rantzer, "System analysis via integral quadratic constraints," *IEEE Trans. Autom. Control*, vol. 42, pp. 819–830, June 1997.

[4] R. E. Kalman, "Physical and mathematical mechanisms of instability in nonlinear automatic control systems," *Trans. ASME*, vol. 79, pp. 553–563, 1957.

[5] V. A. Yakubovich, “Frequency domain conditions of absolute stability and dissipativeness of control systems with a single differentiable element(in russian),” *Dokl. Akad. Nauk SSSR*, vol. 160(2), pp. 298–301, 1965.

[6] ——, “Matrix inequalities method in the theory of stability of controlled systems ii. absolute stability in a class of nonlinearities with the restrictions on the derivative(in russian),” *Avtomat. i Telemekhanika*, vol. 29, pp. 577–583, April 1965.

[7] I. Barbălat and A. Halanay, “Conditions de comportement “presque linéaire” dans la théorie des oscillations,” *Rev. Roum. Sci. Techn. Electrotechn. et Energ.*, vol. 29, pp. 961–979, April-June 1974.

[8] N. E. Barabanov and V. A. Yakubovich, “Absolute stability of control systems having one hysteresis-like nonlinearity(in russian),” *Avtomat. i Telemekhanika*, vol. 40, pp. 5–12, December 1979.

[9] V. Singh, “A stability inequality for nonlinear feedback systems with slope restricted nonlinearity,” *IEEE Trans. Autom. Control*, vol. 29, pp. 743–744, August 1984.

[10] V. Răsvan, “New results and applications of the frequency domain criteria to absolute stability of nonlinear systems,” in *Qualitative Theory of Differential Equations*, 1988, vol. 53, pp. 577–594.

[11] A. Halanay and V. Răsvan, “Absolute stability of feedback systems with several differentiable non-linearities,” *Int. J. Systems Sci.*, vol. 22, pp. 1911–1927, November 1991.

[12] W. M. Haddad and V. Kapila, “Absolute stability criteria for multiple slope-restricted monotonic nonlinearities,” *IEEE Trans. Autom. Control*, vol. 40, pp. 361–365, February 1995.

[13] W. M. Haddad, “Correction to “absolute stability criteria for multiple slope-restricted monotonic nonlinearities”,” *IEEE Trans. Autom. Control*, vol. 42, p. 591, April 1997.

[14] Anon., *Flight Control Design - Best Practices*, NATO-RTO Technical Report 29, December 2000, 2000.

[15] D. H. Klyde, D. T. McRuer, and T. T. Myers, “Pio analysis with actuator rate limiting,” AIAA, Paper 96-3432-CP, 1996.

[16] A. S. Shiraev, “Some remarks on “system analysis via integral quadratic constraints”,” *IEEE Trans. Autom. Control*, vol. 45, pp. 1527–1532, August 2000.

[17] A. Megretski, “Integral quadratic constraints for systems with rate limiters,” Massachussets Institute of Technology, Cambridge MA, Technical Report LIDS-P-2407, December 1997.

[18] U. Jonsson and A. Megretski, “The zames falb iqc for critically stable systems,” Massachusetts Inst. of Technology, Cambridge, MA, LIDS Tech. Rep. P2405, 1997.

[19] V. M. Popov, *Hyperstability of Control Systems*, 1st ed. Berlin-Heidelberg-New York: Springer Verlag, 1973.

[20] G. Zames and P. L. Falb, “Stability conditions for systems with monotone and slope-restricted nonlinearities,” *SIAM J. Control*, vol. 6, pp. 89–106, March 1968.

[21] Y. Venkatesh, “Noncausal multipliers for nonlinear system stability,” *IEEE Trans. Autom. Control*, vol. 15, pp. 195–204, January 1970.

[22] S. Gheţaru, “Hyperstability of discrete time systems (in romanian),” Ph.D. dissertation, The Energy Institute of the Romanian Academy, Bucharest, Romania, 1969.

[23] F. Amato, R. Iervolino, S. Scala, and L. Verde, “Category ii pilot in-the-loop oscillations analysis from robust stability methods,” *J. Guidance, Control and Dynamics*, vol. 24, pp. 531–538, June 2001.

[24] V. Răsvan, D. Popescu, and D. Danciu, “Frequency domain stability inequalities for nonlinear time delay systems,” in *15th IEEE Mediterranian Electromechanical Conference*, Valletta, Malta, 2010.

Smith Predictor Structure Experiments for a Quanser Servo Motor

Ionuț Cristian Reşceanu, George-Cristian Călugăru, Cristina Floriana Reşceanu

***Abstract* — This paper studies the impact of using a Smith predictor in a networked control system. In this purpose, a system like the Quanser SRV-02 DC motor is selected to study the impact of network transmission delays. A number of simulations are performed to reveal the efficiency of the Smith predictor in a control loop subject to random delays. The paper also presents remote configurations using client-server architecture. To complete the demonstration, the results obtained during a real-time experiment are presented for which a great number of delay situations are taken into account in order to achieve the control purpose.**

I. INTRODUCTION

NETWORKED control can be defined as being: the remote control of an operation, this implying a connection, usually electric, between the control device and the equipment with which an operation is performed and can be achieved through:

a. direct wiring
b. other types of interconnection channels such as currents or microwave
c. supervisory control
d. mechanical means [7]

The type of control specified at point b. can be extended to environmental transmissions such as optical or acoustical transmissions.

In 1957, Smith introduced a new technique dealing with the instability issues that occur in control systems, also experimenting with delays. The main idea of this technique [2] is to substitute the conventional controller $C(s)$ with a new design $C^*(s)$ in such a way that closed loop control of the process dynamics without delay can be achieved. Given a process, with a delay of τ seconds, Smith proposed that a new controller $C^*(s)$ be introduced in the closed loop with the delayed process $G(s)e^{-\tau s}$ as seen in figure 1a.

The effect of the Smith controller $C^*(s)$ is to eliminate the delay from the control loop as can be seen in figure 1 b. and to effectively realize the control of the non-delayed process dynamics $G(s)$ using a conventional controller $C(s)$.

A conventional control structure like in figure 1a induces infinity of poles in the system. Thus, $H_0(s)$ is:

$$H_0(s) = \frac{C(s)G(s)e^{-\tau s}}{1 + C(s)G(s)e^{-\tau s}} \qquad (1)$$

In order to eliminate the unwanted effect of the delay from the process model, a separation of the delay inside the model is attempted and also the use of variable $y(t)$ instead of $y(t-\tau)$ is realized. The structure of the control system, in this case, is the one presented in figure 1b with the following transfer function:

$$H_0^*(s) = \frac{C^*(s)G(s)e^{-\tau s}}{1 + C^*(s)G(s)e^{-\tau s}} \qquad (2)$$

Imposing the condition that the structures have the same input-output behavior, we obtain:

$$H(s) = H_0^*(s) \qquad (3)$$

The Smith predictor will be defined by the following transfer function:

$$C^*(s) = \frac{C(s)}{1 + C(s)G(s)(1 - e^{-\tau s})} \qquad (4)$$

For the design of the control system, any known method for non-delayed systems can be used in order to determine $C^*(s)$.

Manuscript received May 10, 2010

Ionuț Cristian Reşceanu is with the University of Craiova, Faculty of Automation, Computers and Electronics, Decebal Blvd., no.107, Craiova, Dolj, Romania (phone: +40.251.438198; fax: +40.251.438198; e-mail: resceanu@robotics.ucv.ro).

George Cristian Călugăru is with the University of Craiova, Faculty of Automation, Computers and Electronics, Decebal Blvd., no.107, Craiova, Dolj, Romania (phone: +40.251.438198; fax: +40.251.438198; e-mail: calugaru.george.nds@gmail.com).

Cristina Floriana Reşceanu is with the University of Craiova, Faculty of Automation, Computers and Electronics, Decebal Blvd., no.107, Craiova, Dolj, Romania (phone: +40.251.438198; fax: +40.251.438198; e-mail: cristina@robotics.ucv.ro).

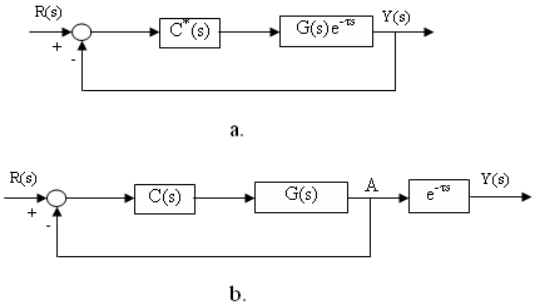


Fig.1 The Smith controller and its equivalent conventional controller

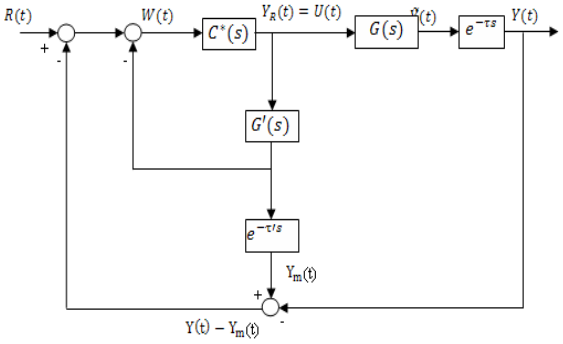


Fig 2 Smith Predictor classic configuration

Even though this paper runs its applications on the classical Smith predictor configuration, many more Smith predictor configurations have emerged in literature: improved versions of the original Smith predictor, Smith predictor with network delay adaptation, Smith predictor for controlling integrator processes etc. More details can be seen in [1], [3] and [4].

II. QUANSER SRV-02 PLANT PRESENTATION

The Quanser SRV-02 plant has two speed configurations available: "low gear ratio" and "high gear ratio" [10].

Attached to the system is the encoder that measures 1024x4 impulses in a complete rotation. This means that 360 degrees are equivalent to 1024x4=4096 impulses. The resolution of the encoder counter is 360/4096= =0.0878906250 degrees. The model used during the experiments is a US Digital Optical Encoder kit.

The tachometer supplies an analogical signal and generates 1,5V every 1000 RPM. For the Quanser SRV-02 plant, the tachometer is attached directly to the motor and there are no response delays. The potentiometer supplies an analogical signal proportional with the rotation angle. This is a 10kΩ sensor which measures 352 degrees expressed in voltages ranging from +5V to -5V. The difference between the potentiometer and the encoder is that the latter can measure several complete rotations as the potentiometer can only measure 360 degrees and back.

The open loop transfer function is [5]:

$$H_1(s) = \frac{\Omega_0(s)}{V_i(s)} = \frac{\frac{\eta k_m k_g}{R_a J_{eq}}}{s + \frac{B_{eq}}{J_{eq}} + \frac{\eta k_m^2 k_g^2}{R_a J_{eq}}} \quad (5)$$

Where:

$$\eta = \eta_{mr} \cdot \eta_{gb} \quad (6)$$

$$J_{eq} = k_g^2 (J_m + J_{tach}) + J_L \quad (7)$$

$$H_1(s) = \frac{\Omega_0(s)}{V_i(s)} = \frac{a_m}{s + b_m} \quad (8)$$

Where $a_m = \frac{\eta k_m k_g}{R_a J_{eq}}$ and $b_m = \frac{B_{eq}}{J_{eq}} + \frac{\eta k_m^2 k_g^2}{R_a J_{eq}}$.

The transfer function of the plant with the output $\theta_0(s)$ is:

$$H_2(s) = \frac{\theta_0(s)}{V_i(s)} = \frac{\frac{\eta k_m k_g}{R_a J_{eq}}}{s\left(s + \frac{B_{eq}}{J_{eq}} + \frac{\eta k_m^2 k_g^2}{R_a J_{eq}}\right)} \quad (9)$$

$$H_2(s) = \frac{\theta_0(s)}{V_i(s)} = \frac{a_m}{s + b_m} \quad (10)$$

All the parameters from the transfer functions are listed in table 1.

TABLE I
PLANT MODEL PARAMETERS

Parameter	Significance	Value
R_a	Armature resistance	2.6 Ω
k_m	Motor voltage constant	0.00767 V-s/rad
k_g	Low gear ratio	14
η_{mr}	Motor efficiency due to rotational loss	0.87
η_{gb}	Gearbox efficiency	0.85
B_{eq}	Equivalent friction referred to the secondary gear	0.001 Nm/(rad/s)
J_m	Armature inertia	3.87 x 10^{-7} kg m^2
J_{tach}	Tachometer inertia	0.7 x 10^{-7} kg m^2
J_L	Load inertia	16.333 x 10^{-6} kg m^2

III. Remote Position Control of the Quanser SRV-02 Actuator Using a Modified Smith Predictor Structure

The control loop for the Quanser SRV-02 actuator is presented in the figure below, using a PI controller [2]:

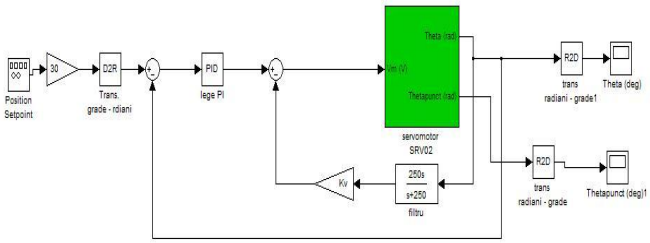


Fig 3. Control loop of the actuator using Simulink

In the simulation above , the aim is to control the position of the motor [9]. The reference is set to 30 degrees. Theta represents the angle and ThetaPunct represents the angular velocity.

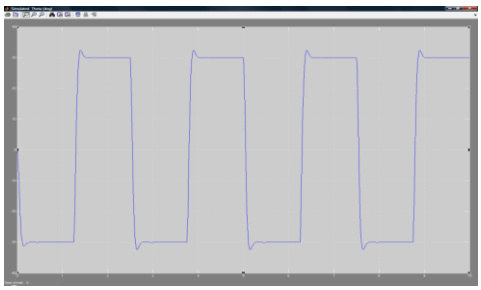


Fig.4. The actuator response to a signal generator

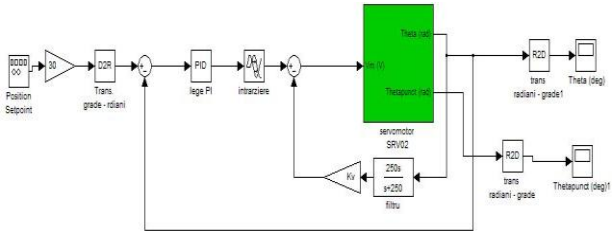


Fig. 5. The actuator control in the case of a delay

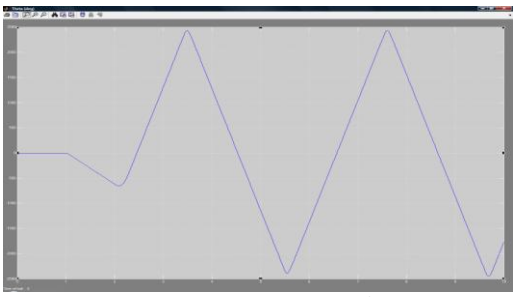


Fig. 6. The actuator response in the case of delay

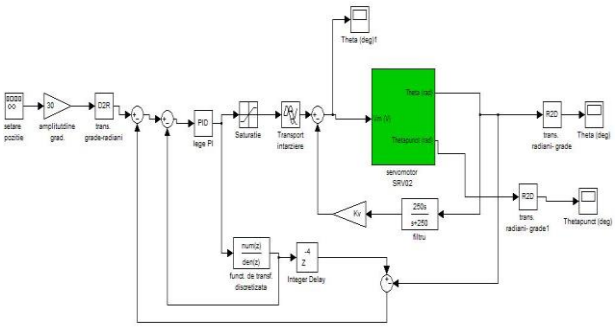


Fig. 7. The Quanser SRV-02 actuator control using a Smith predictor

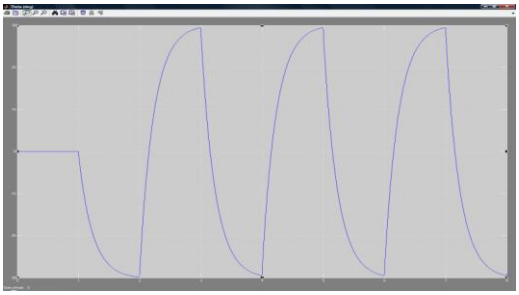


Fig. 8. The system response when using a Smith predictor

IV. Remote Position Control For the Quanser srv-02 Plant Using a Modified Smith predictor – Real-time Experiment

The remote control will be achieved through a local area network, intranet or internet, using the client-server architecture provided by the WinCon environment. The configuration used will have a server and a WinCon client functioning as nodes on a TCP/IP network.

WinCon is a real-time Windows 2000/XP application. It allows you to run code generated from Simulink diagram in real-time on the same PC (also known as local PC) or on a remote PC. The application also allows plots of real-time data on-line through WinCon scopes. Also, model parameters can be changed in running-time. The automatically generated real-time constitutes a stand-alone controller in this case independent from Simulink.

WinCon software consists of two distinct parts: WinCon Client and WinCon Server. Communication is established through the TCP/IP protocol. WinCon Client runs in hard real-time while WinCon Server represents a separate graphical interface, running in user mode. WinCon supports two configurations: the local configuration (a simple machine) and the remote configuration (two or more machines). In the local configuration, WinCon Client and WinCon Server run at the same time on the same machine.

In remote configurations, WinCon Client functions on a platform separated from Simulink and WinCon Server. [10]

The minimal remote configuration (configuration 1) needs two computers as can be seen in the following figure:

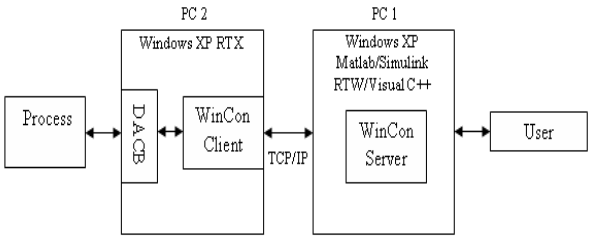


Fig.9 The minimal remote configuration

A much more complex case is the second configuration which implies a number n of computers. The communication between these computers is done through a local network whether it is intranet of internet. In the figure above, DACB stands for Data Acquisition and Control Board which is used to interface the real-time code to the controlled process.

The general case is the third configuration which has many WinCon servers and many WinCon clients functioning as nodes on a TCP/IP network. For example, WinCon server S1 can download a code to the C1 WinCon client and a different code to the C2 WinCon client. The S2 WinCon server can download the code to the WinCon C3 client. The difference between the WinCon servers S1 and S2 is that S1 is a primary server for clients C1 and C2 and can stop the code that it downloads. The S2 WinCon server is a secondary server for WinCon clients C1 and C2 and can only collect data from these and is primary server for WinCon client C3.

A. The timing of events in real-time

The timing diagram presented in figure 10 illustrates the principle of a real-time operation. The system or hardware clock generates interrupts at a fixed interval T_s, this being the sampling rate chosen by the user.

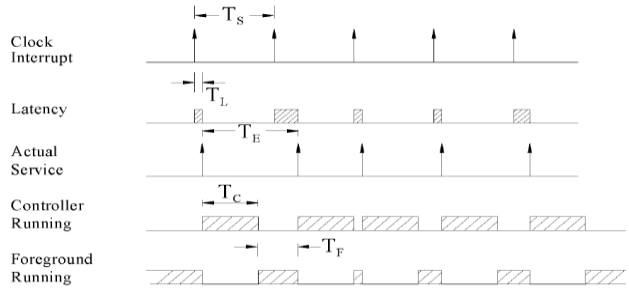


Fig. 10. The timing diagram of a real-time operation

Because the CPU can serve other hardware interrupts or the direct memory access requirements it takes an intermediary timp period in order to serve the interrupt (IRQ). This period is called interruption period T_L. The main tasks indirectly affect this delay period by using hardware resources like the video drive or the hard drive. While the main operations and even the control operations in Windows XP function with lower priority that real-time code, operations performed directly by hardware means (DMA – Direct Memory Access), can affect this delay period. At the end of this delay period, the controller performs the necessary calculations for that sampling rate within a time interval called T_C. The effective sampling rate T_E is the time between two interrupt service routines (ISR). T_E is variable and depends on the size of the delay. Performed tests indicate a delay of tens of microseconds under Windows XP within the real-time application. This value depends on the number of processes running, available hardware resources and the system's frequency. When operating in remote configurations in which the WinCon client runs on a dedicated computer, the delay can be under 10 seconds. The calculation delay T_C depends mainly on the complexity of the running real-time controller. Taking into consideration the fact that the sampling rate is dictated by a hardware timer, the average sampling rate is very accurate.

The timer of the calculation system is called a system clock and a timer from de data acquisition board is called a hardware clock. The sampling rate obtained under RTX is limited to 10 kHz because of the minimal period of the RTX HAL timer which is under 100 microseconds. For sampling rates larger than 10 kHz under Windows XP the use of the hardware clock is recommended.

B. The real-time experiment

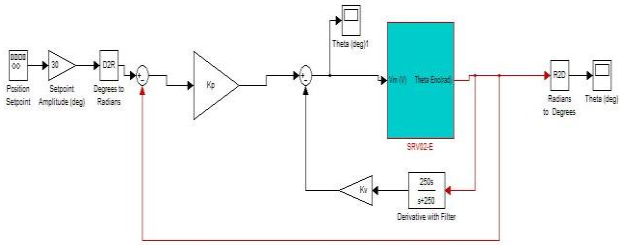


Fig.11. The control structure for the SRV-02 plant

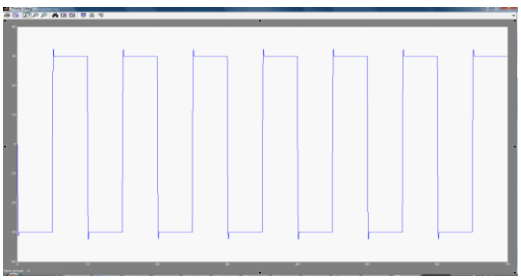


Fig. 12. The θ angle measured by the encoder for a frequency of 0.1 Hz

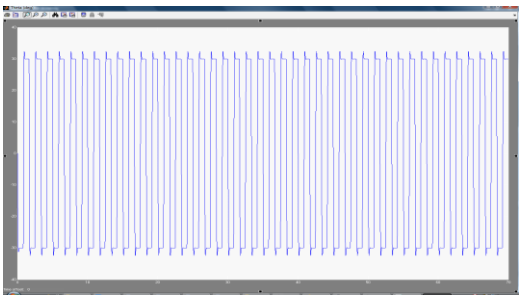


Fig. 13. The θ angle measured by the encoder for a frequency of 0.6 Hz

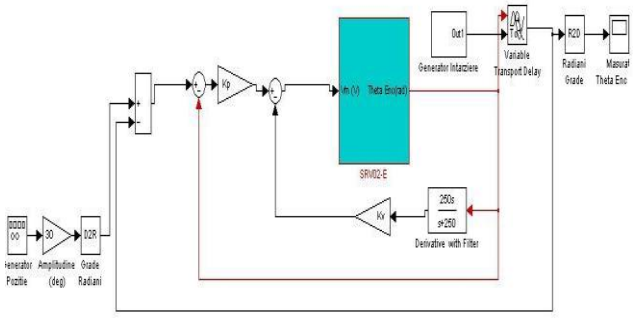


Fig. 14. The control loop for SRV-02 plant with a variable delay between [0;0.5]

The subsystem for the simulation of a variable delay is presented in [3]:

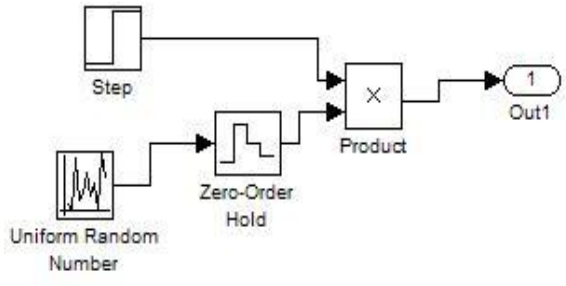


Fig.15. System for generating a uniform distributed random signal

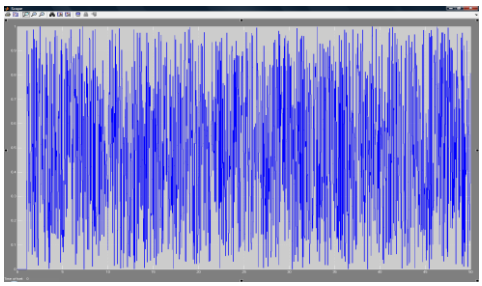


Fig.16. Time distribution of communication delay

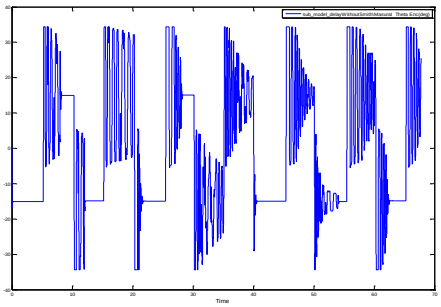


Fig.17. θ angle measured by the encoder in the case of variable delay occurrence

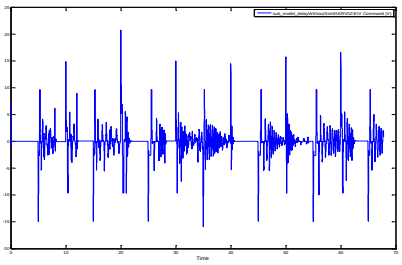


Fig. 18. Plant command in the case of variable delay occurrence

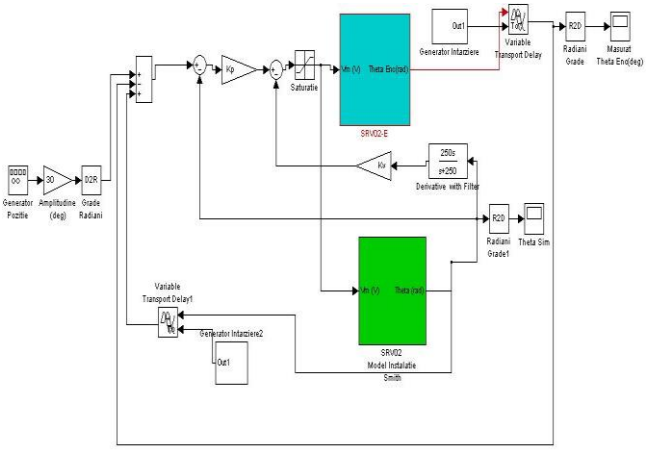


Fig.19. Control loop for the SRV-02 plant using a Smith predictor

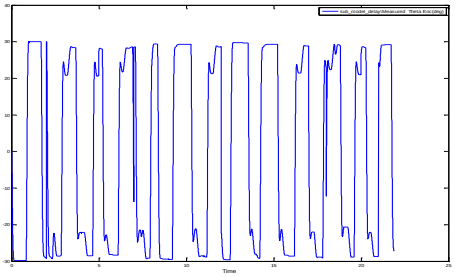


Fig. 20. The θ angle measured by the encoder when using a Smith predictor – the frequency of the command signal is 0.6 Hz

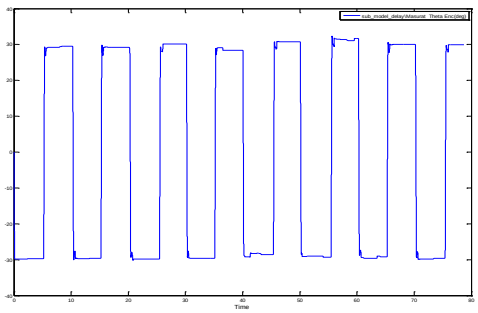


Fig. 21. The θ angle measured by the encoder when using a Smith predictor – the frequency of the command signal is 0.1 Hz

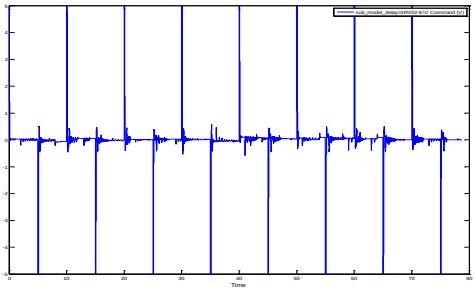


Fig.22. Actuator command when using a Smith predictor – the frequency of the command signal is 0.1 Hz

The delay is compensated well enough because the frequency of the command signal has been decreased.

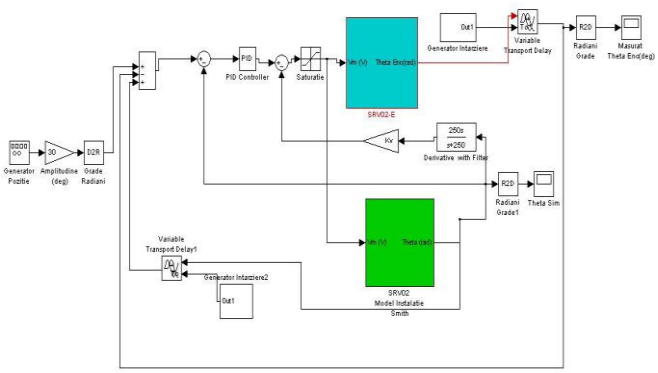


Fig.23. Control loop for the Smith predictor structure with an extra PID controller

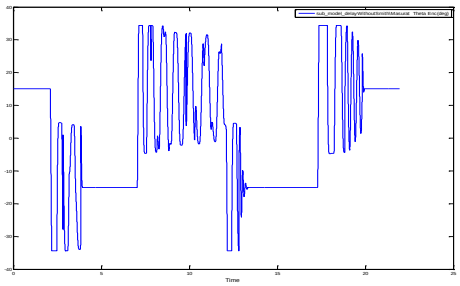


Fig. 24. The θ angle measured by the encoder in case of variable delay, with an extra PID controller, but no Smith predictor

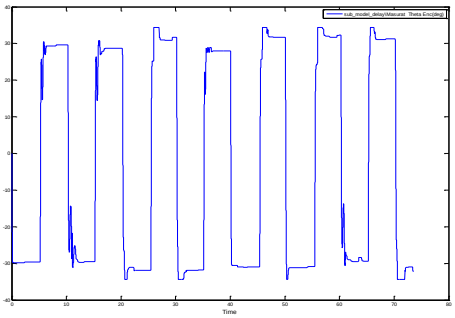


Fig.25 The θ angle measured by the encoder in case of variable delay, with an extra PID controller and with Smith predictor

An interesting application about tracking the position of a DC servo motor can be seen in [6].

V. CONCLUSION

This paper presents a control application based on a DC motor. The application implies the remote position control of the DC motor using a Smith predictor structure in which random transmission delays occur. The impact of these delays is studied and the results of a real-time experiment are presented.

The conclusion of the real-time experiment using the Quanser SRV-02 plant is that the delay occurrence frequency depends on the time constant of the process. If the delay occurrence frequency is high and the process time constant is low, the compensation will not be fully achieved.

The sampling rate is in strict relation with the time constant of the process.

In practice, the sampling rate must be between 4 to 8 times lower than the dominant time constant of the process.

The real time experiment proves that in some cases it is recommended that the sampling rate exceed that 1/8 dominant time constant ratio.

REFERENCES

[1] Aström K.J., Hang C.C. and Lin B.C.(1994), *A New Smith Predictor for Controlling a Process With an Integrator and Long Dead-Time*, IEEE Transaction on Automatic Control, 39(2), pp. 343-345

[2] Vânătoru M, (2001), *Conducerea automată a proceselor industriale,* Editura Universitaria Craiova

[3] Velagic J. (2008), *Design of Smith-like Predictive Controller with Communication Delay Adaptation*, World Academy of Science, Engineering and Technology, no. 47, pp 199-203

[4] Veronesi M. (2003), *Performance Improvement of Smith Predictor through Automatic Computation of Dead Time*, Yokogawa Technical Report English Edition, pp 25-30

[5] Saadat H. (2004), *Control Systems I Laboratory Manual*, Electrical Engineering and Computer Science Department, Milwaukee School of Engineering.

[6] Song K.W., Choi G.S., Choi C.H. (2008), *Tracking Position Control of DC Servo Motor in Lon Works/IP Network*, International Journal of Control, Automation and Systems, vol. 6, no. 2, pp. 186-193.

[7] ****IEEE Standard Dictionary of Electrical and Electronic Terms*, the Institute of Electrical and Electronic Engineers, Inc., New York

[8] *** Quanser Consulting Inc., SRV-02 series, User Manual

[9] *** Quanser Consulting Inc., SRV-02 series, Rotary Experiment #1, Position Control

[10] *** Quanser Consulting Inc., WinCon 5.0 User Manual.

Control of Legged Robot with Locked Joint

Cristina Floriana Reşceanu

***Abstract*—Several factors must be considered for robotic task execution in the presence of a fault, including: detection, identification, and accommodation for the fault. Fault-tolerant locomotion is defined as gaits with which legged robot can continue their walking after a failure event occurs to a leg of the robot. This means that for a given type of failure, the problem of finding fault-tolerant gaits can be formulated with which legged robots can continue their walking after an occurrence of a failure, maintaining static stability. In this paper, a fault tolerant control using dynamics of leg is determined. The failure considered in this paper is a locked joint failure. Also, problem of kinematics constraints of the failed leg in fault-tolerant gaits for a locked joint failure is presented.**

I. Introduction

GENERALLY, it is recognized that the mobility characteristics of terrestrial animals are in many respects superior to those of wheeled or tracked vehicles for off-road locomotion. Indeed, it is well know that much of the earth's surface is accessible only to men on foot or to certain types of multi-legged animals. This fact has motivated considerable attempts to develop the multi-legged walking vehicles. During the past several years, the generalized walking vehicle with multiple degrees of freedom was considered, and the coordinated control of legs by computer to complete the walk by the suitable choice of lifting legs was investigated.

The basic control problems that all the walking vehicles confront are:

- How to generate the trajectory and the average speed?
- How to determine the best sequence for liftoff and placing the feet?
- What is the suitable distance that each leg should transfer in order to maintain a prescribed statically stability?
- How to control the body's inclination and height?
- How to develop a measurement system and information processing method to support the motion planning?

The problem of choosing the best sequence for liftoff and placing the feet of a walking vehicle is the gait selection problem.

Another advantage of legged robots is their robustness to damages to legs (in specially, in static walking). Legged robots are able to continue to walk against a fault in a leg. These may maintain static stability even if a leg is broken so that is cannot support the robot body. Adaptation to a leg failure is one of the most important requirements for robust walking of legged robots, because the repair of the failed leg is almost impossible after legged robots have been launched in most applications. From their characteristics of having multi-legs, walking robots have inherent fault tolerance capability against a leg failure since a failed leg for itself may not cause catastrophic failure or instability in static walking. Among various leg failures, a locked joint failure is one of common failures that can be frequently observed in dynamics of robot manipulators [1].

If failed joints are supposed to be locked individually, a single joint failure reduces the number of degrees of freedom of the robot manipulator by one and reduces its workspace to a certain limit.

In this paper, we focus our concern on the problem of kinematics constraints of the failed leg in fault-tolerant gaits for a locked joint failure and, also, on the dynamic control of the robot's leg in fault condition.

II. Legged Robot Model

A. Kinematics Consideration

We consider a quadruped robot whose two-dimensional model is shown in Fig. 1 [2].

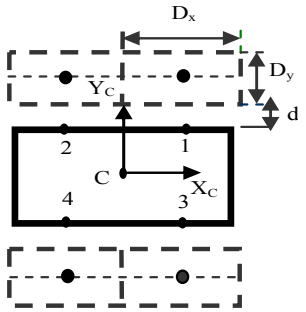


Fig. 1. Bidimensional model of quadruped robot

The four legs are placed symmetrically about the longitudinal axis and have rectangular working areas with the length D_x and the width D_y. C_i is the centre point of leg i working area. The robot body is also in the shape of a rectangle with 2U width and distant from working areas by d. C is the centre of gravity of the body and the origin of the robot coordinate system X–Y.

A leg attached to the quadruped robot has the geometry of the articulated arm as showed in Fig. 2. This model has two rigid links and three revolute joints; the lower link is

Cristina Floriana Reşceanu is with the University of Craiova, Faculty of Automation, Computers and Electronics, Decebal Blvd., no.107, Craiova, Dolj, Romania (phone: +40.251.438198; fax: +40.251.438198; e-mail: cristina@robotics.ucv.ro).

connected to the upper link via an active revolute joint and the upper link is connected to the body via an active revolute joint which is parallel with the body's longitudinal axis.

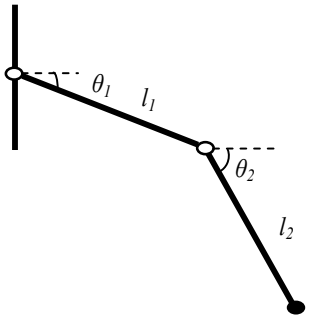


Fig. 2 . Generalized coordination of quadruped leg

B. *Kinematic Constraint*

In this section, we show that there exists a range of kinematic constraints which the configuration of the failed leg should satisfy for guaranteeing the existence of the previous fault-tolerant gait. We assume that a locked joint failure occurs to any joint of leg 1.

Case I: Failure of Joint One

When joint one of leg is locked because failure, the leg can swing only the second link by the knee joint in the lateral direction. The resulting reachable region is thus of an arc shape as is shown in Fig. 3.

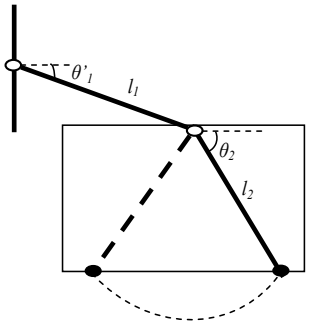


Fig. 3. Locked failure at joint one in straight-line walking (lateral view)

Depending on the features of its lateral motion, the leg can be placed on the inner foothold position P' of the outer position P, as show in Fig. 4, or cannot be placed on the ground in the worst case.

The kinematic constraint for guaranteeing such foothold positions can be described as

$$\frac{D_y}{2} + d \le r \le \bar{r} \qquad (1)$$

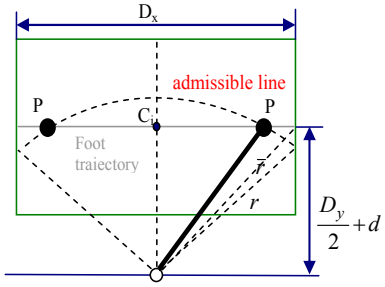


Fig. 4. Kinematics constraint of the quadruped leg

where r is the radius of the arc and $\bar{r}$ is the distance between the leg attachment point and the front (or rear) boundary of the foot trajectory projected onto the X–Y plane.

If the radius of the arc r is in the above range, there exists at least one intersection point of the arc and the foot trajectory. For describing (1) in terms of joint angles and robot parameters, let us rewrite $\bar{r}$ and r as

$$\bar{r} = \frac{1}{2}\sqrt{D_x^2 + (D_y + 2d)^2} \qquad (2)$$

$$r = l_1 \cos\theta'_1 + l_2 \cos\theta_2 \qquad (3)$$

where θ'_1 is the locked angle of joint one. Note that r is identical to the length of the leg projection onto the working area. Since the robot body is supposed to have a constant altitude, the angle of joint three θ_2 should also remain the same in the support phase. Substituting (2) and (3) into (1) leads to

$$\frac{D_y}{2} + d \le l_1 \cos\theta'_1 + l_2 \cos\theta_2 \le \frac{1}{2}\sqrt{D_x^2 + (D_y + 2d)^2} \qquad (4)$$

The above result prescribes the kinematic constraint of locked angle θ'_1 which guarantees the existence of the fault-tolerant gait for straight-line walking proposed in [3].

Case II: Failure of Joint Two

The motion of leg in the case of a locked failure at joint two is almost similarly as that of a locked failure of joint one. If joint two, or the knee joint, is locked because failure, the leg is reduced to a manipulator with one link and one revolute joint. The reduced reachable region in the working area is an arc as in Fig. 5.

The lateral motion of the failed leg is similar with that from figure 4, in which the leg is moved only by joint one and the second link is passively lifted associated with the lift-off of the first link.

The kinematics constraint for the existence of the fault-tolerant gait is the same as the case of joint one:

$$\frac{D_y}{2} + d \le l_1 \cos\theta_1 + l_2 \cos\theta'_2 \le \frac{1}{2}\sqrt{D_x^2 + (D_y + 2d)^2} \qquad (5)$$

where θ'_2 is the locked angle of joint two.

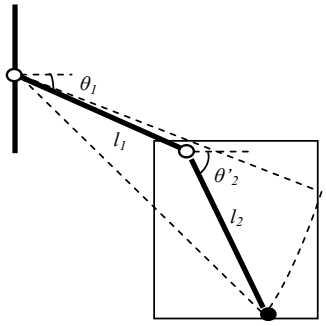


Fig. 5. Locked failure at joint two in straight-line walking (lateral view)

C. *Dynamic model of leg*

Dynamic equations of robot's leg are the following [4]:

$$\tau_1 = M_{11}\ddot{\theta}_1 + M_{12}\ddot{\theta}_2 - c\dot{\theta}_2^2 - 2c\dot{\theta}_1\dot{\theta}_2 + G_1 + F_{ex1} \quad (6)$$

$$\tau_2 = M_{12}\ddot{\theta}_1 + M_{22}\ddot{\theta}_2 + c\dot{\theta}_1^2 + G_2 + F_{ex2} \quad (7)$$

where

$$\begin{aligned}
M_{11} &= m_1 l_{c1}^2 + I_1 + m_2\left(l_1^2 + l_{c2}^2 + 2l_1 l_{c2}\cos\theta_2\right) + I_2 \\
M_{22} &= m_2 l_{c2}^2 + I_2 \\
M_{12} &= M_{21} = m_2 l_{c2}^2 + m_2 l_1 l_{c2}\cos\theta_2 + I_2 \\
c &= m_2 l_1 l_{c2}\sin\theta_2 \\
G_1 &= -g\left(m_1 l_{c1}\cos\theta_1 + m_2\left[l_{c2}\cos(\theta_1+\theta_2) + l_1\cos\theta_1\right]\right) \\
G_2 &= -g m_2 l_{c2}\cos(\theta_1+\theta_2) \\
F_{ex1} &= -F_z\left[l_1\cos\theta_1 + l_2\cos(\theta_1+\theta_2)\right] + F_x\left[l_1\sin\theta_1 + l_2\sin(\theta_1+\theta_2)\right] \\
F_{ex2} &= -F_y l_2\cos(\theta_1+\theta_2) + F_x l_2\sin(\theta_1+\theta_2)
\end{aligned} \quad (8)$$

G_1 and G_2 are gravitational moments due to mass m_1 and m_2, F_{ex1} and F_{ex2} environmental interaction of the foot with the ground. During stepping, when foot is in transfer phase, F_z the foot weight and $F_x = 0$. When the foot is in contact phase with the ground and the robot is a quadruped, then $F_z = 1/4$ of the whole weight of the robot. In this case $F_x = fF_z$, where f the coefficient of friction. In following simulations we will consider that $m_1 = m_2 = m$, $l_1 = l_2 = l$ and $l_{c1} = l_{c2} = l/2$.

III. Control of Leg Position

A. *Normal Condition*

We propose a closed loop control system to achieve a desired position using the mathematical model of the leg as shown below [5]:

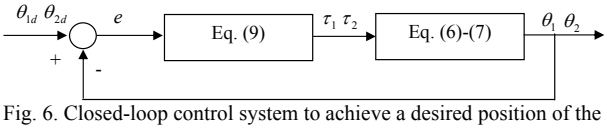


Fig. 6. Closed-loop control system to achieve a desired position of the foot

Error of the control system will be defined by:

$$e = e_i = q_i - q_{id} = \theta_i - \theta_{id},\ i \in [1,2] \quad (9)$$

We propose a control law of the next form:

$$\tau_i = k_q^1 e_i + k_q^2 \dot{e}_i \quad (10)$$

where k_q^1, k_q^2 are positive factors.

The simulation results are suggestive illustrated in Fig. 7, which can track initial positions, final and intermediate positions. Mechanical parameters of the system are m = 1400g, length of each link is the total length of the leg is L = 0.8m. We consider the initial position of the foot is lifted to its position [5].

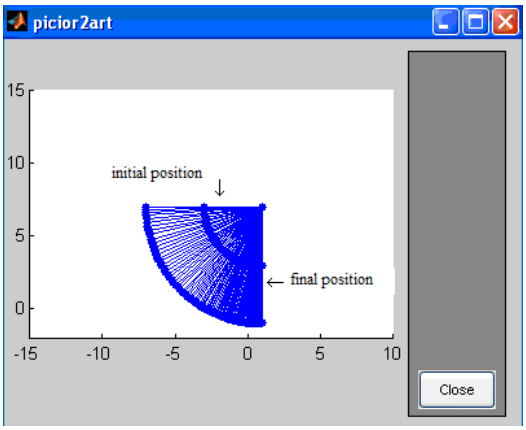


Fig. 7. Evolution of foot to the desired position

For a quantitative understanding of system evolution, as represented in Fig.8, phase portrait of the movement is represented, where we considered the overall error of control system defined by [5]:

$$e(t) = q(t) - q_d(t) \quad (11)$$

$$\dot{e}(t) = \frac{\partial q(t)}{\partial t} \quad (12)$$

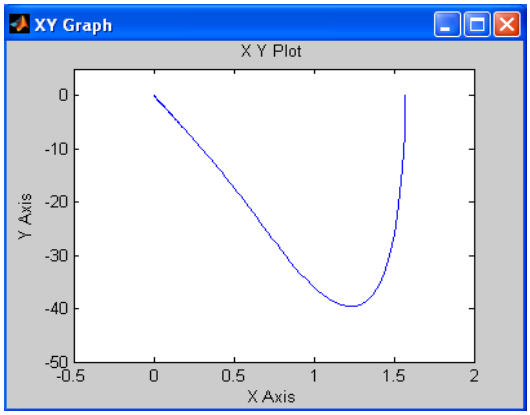


Fig. 8. Phase portrait

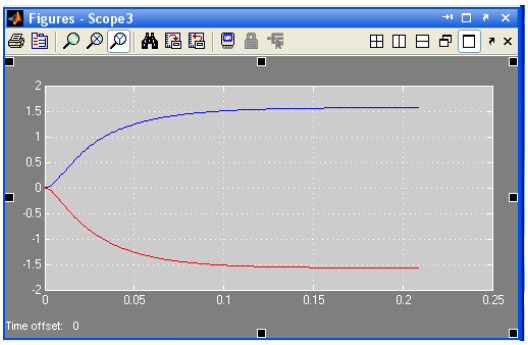


Fig. 9. Evolution of generalized internal coordinates q

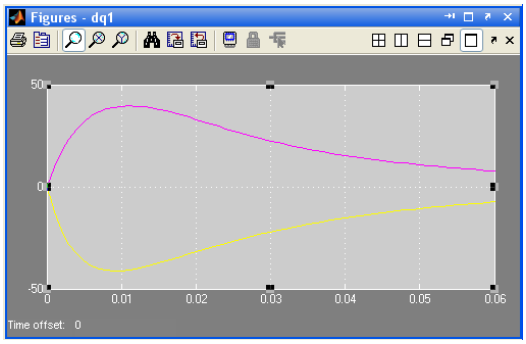


Fig. 10. Evolution of derivatives dq / dt

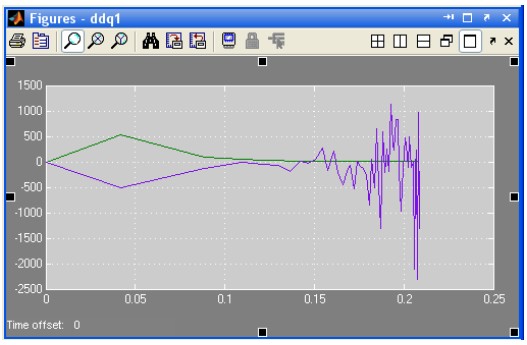


Fig. 11. Evolution of angular acceleration d2q/dt2

B. *Fault Condition*

We propose a closed loop control system to achieve a desired position of leg with a blocked joint, as shown following:

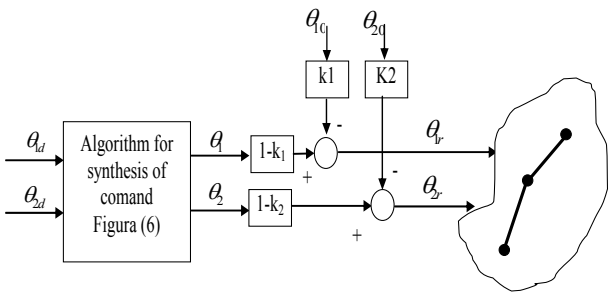


Fig. 12. Control system to achieve a desired position of the foot when blocking a joint

According to the figure, commands were sent to foot the expression:

$$\theta_{ir} = (1-k_i)\theta_i + k_i\theta_{i0},\ \text{i=[1,2]} \tag{13}$$

where:

k_i =0, when joint actuator is in good condition;
k_i =1, when joint actuator is locked in an arbitrary angle θ_{i0}.

The vector of commands has the following expression:

$$\Theta_r = \begin{bmatrix}\theta_{1r}\\ \theta_{2r}\end{bmatrix} = (I - K_D)\begin{bmatrix}\theta_1\\ \theta_2\end{bmatrix} + K_D\begin{bmatrix}\theta_{10}\\ \theta_{20}\end{bmatrix} \tag{14}$$

where:

$$K_D = \begin{bmatrix}k_1 & 0\\ 0 & k_2\end{bmatrix} \tag{15}$$

is matrix of fault. When $K_D \equiv 0$, it can make the claim that the system works correctly.

The vector $[\Theta_0]^T = [\theta_{10} \quad \theta_{20}]$ represent values of blocking angles. Respect to Fig. 12 and relations (14) and (15), we study the behavior of leg by the choosing particular structures for matrix of fault.

Case I: Failure of Joint One

The simulation results are suggestive illustrated in Fig. 13, which can track initial positions, final and intermediate positions. Mechanical parameters of the system are m = 1400g, length of each link is the total length of the leg is L = 0.8m. We consider the initial position of the foot is lifted to its position and the first joint is blocked at angle $\theta_{10} = -\pi/4$ [5].

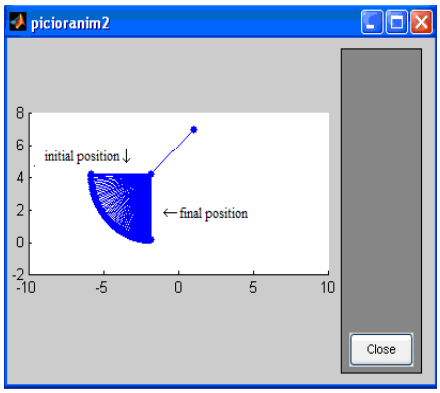


Fig. 13. Evolution of foot to the desired position

Evolution of error and its derivative, represented in phase plane, it can see the following figure:

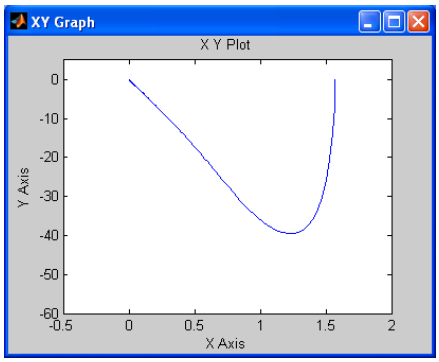


Fig. 14. Evolution of foot to the desired position

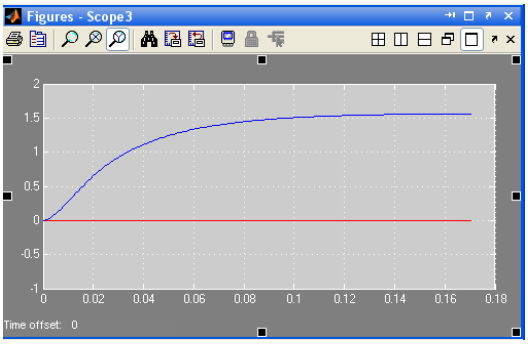


Fig. 15. Evolution of generalized internal coordinates q

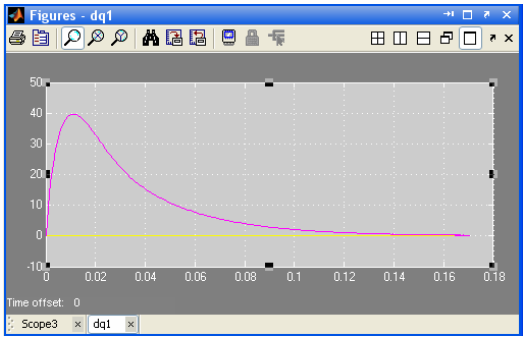


Fig. 16. Evolution of derivatives dq / dt

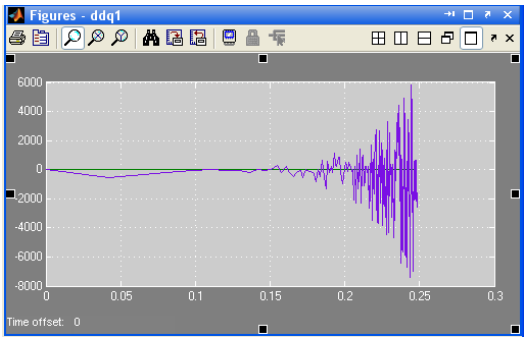


Fig. 17. Evolution of angular acceleration d2q/dt2

Case II: Failure of Joint Two

We consider the initial position of the foot is lifted to its position and the first joint is blocked at angle $\theta_{20} = \pi/2$ [5]. The simulation results are suggestive illustrated in Fig. 12, which can track initial positions, final and intermediate positions.

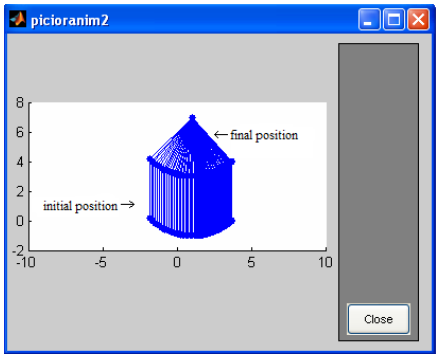


Fig. 18. Evolution of foot to the desired position

Evolution of error and its derivative, represented in phase plane, it can see the following figure:

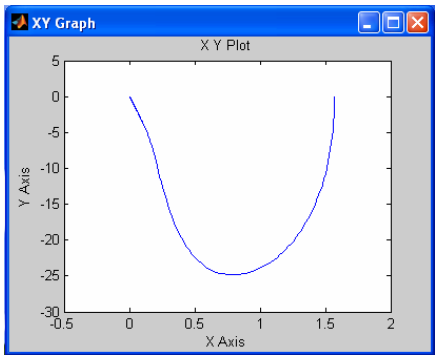


Fig. 19. Evolution of foot to the desired position

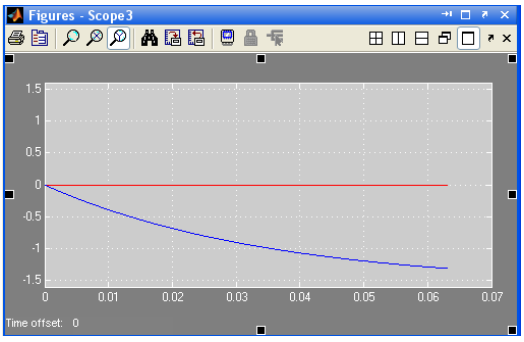


Fig. 20. Evolution of generalized internal coordinates q

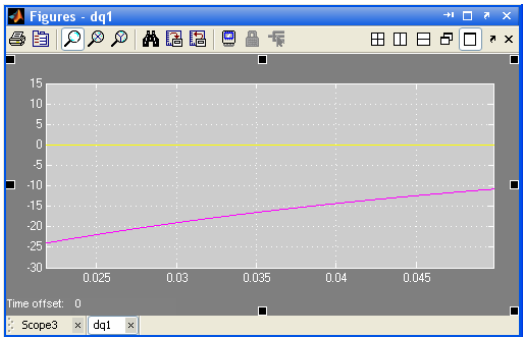


Fig. 21. Evolution of derivatives dq / dt

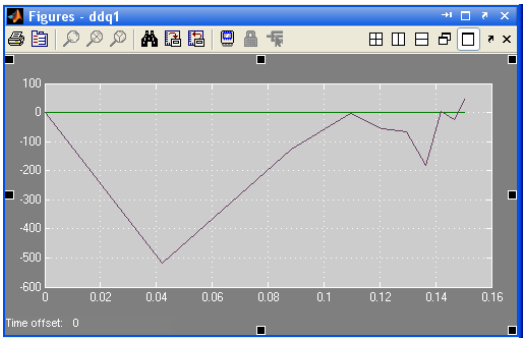


Fig. 22. Evolution of angular acceleration d2q/dt2

From the simulations presented were obtained the following results:

-Fig. 7 illustrates the motion-control robot right leg, which is highlighted in Figure 8 the error and its derivative tend to zero. In Fig. 9 to 11 are presented the evolution of robot foot generalized coordinates, velocity and acceleration.

-Fig. 13 illustrates motion-control when blocking first leg joints and in Figure 14 presents the evolution of the error and its derivative tend to zero. In Fig. 15 to 17 are presented the evolution of robot foot generalized coordinates, velocity and acceleration.

-Fig. 18 illustrates motion-control when blocking second leg joints and in Figure 19 presents the evolution of the error and its derivative tend to zero. In Fig. 20 to 22 are presented the evolution of robot foot generalized coordinates, velocity and acceleration.

IV. Conclusion

In conclusion, a fault-tolerant control is proposed for robot's leg. This control is based on the dynamics of the leg model. The fault-tolerant control can be applied to a generic class of actuator faults that are second-order differentiable. The effectiveness of the proposed fault-tolerant control is illustrated through experimental results.

Acknowledgment

This work was supported by the strategic grant POSDRU/89/1.5/S/61968, Project ID61968 (2009), co-financed by the European Social Fund within the Sectorial Operational Program Human Resources Development 2007-2013.

References

[1] C.L Lewis, A.A. Maciejewski, "Fault tolerant operation of kinematically redundant manipulators for locked joint failures" IEEE Trans. Robot. Autom., Vol.13, No.4, pp. 622–629, 1997.

[2] F.T. Chen, H.L. Lee, D.E. Orin, "Increasing the locomotive stability margin of multilegged vehicles" Proc. IEEE Int. Conf. Robot. Automat., pp. 1708–1714, 1999.

[3] Hirose S., "A study of design and control of a quadruped walking vehicle," *Int. J. Robotics Res.,* vol. 3, no. 2, pp. 113-133, 1984.

[4] C. Pana, V.l Stoian, "A Fault-Tolerant Control System for a Hexapod Robot", 4th Conference Mechatronic System and Materials 2008, MSM 2008 Bialystok, Poland , July 14-17 2008.

[5] C. Pana, "Control algorithms for legged robots in fault conditions," Ph.D. dissertation, Dept. Mechatronics., University of Craiova., 2009.

Disturbance model in explicit control laws

Pedro Rodriguez-Ayerbe and Sorin Olaru.

Abstract— **The paper deals with the predictive control for linear systems, described by piecewise affine (PWA) control laws formulations. The main goal is to reduce the sensitivity of these schemes with respect to the model uncertainties. This objective can be attained by considering worst-case (min-max) formulations, optimization over the control policies or tube predictive control. These comprehensive approaches may lead to fastidious on-line optimization thus reducing the range of application. In the present paper, a two stage predictive strategy is proposed, which synthesizes in the first place an analytical (continuous and piecewise linear) control law based on the nominal model and secondly robustifies the control law in the neighborhood of the equilibrium point (the feedback gain obtained for the unconstrained control problem - most often assimilated to the LQR gain). This robustification is globally expanded to all the state space of the piecewise structure by means of its corresponding disturbance model.**

I. INTRODUCTION

The model predictive control (MPC) laws are optimisation based techniques which allow constraints handling from the design stage. Their practical implementation is related to the real-time computation of a finite horizon optimal control sequence. The analytical formulation of the optimum and its on-line evaluation avoids the important computational effort required for real-time optimisation. Solutions in this direction exist for linear and quadratic cost functions subject to linear constraints thanks to the Abadie constraint qualification [1]. It must be said that these are in fact a part of a larger class of parametric convex programs [2] for which exact [3], [4], [5] or approximate [6], [7] algorithms exist.

In the case of robust predictive control laws, the model uncertainties and the disturbances can be taken into account at the design stage. A popular methodology in this direction is the one based on a min-max criterium (when the extreme combination of disturbances or uncertainties are known) [8], [9], [10], which comes finally to the resolution of a single parametric linear program. The structure of this ultimate optimisation is however quite complex and large prediction horizons cannot be handled due to the exponential growth of disturbance realisations that have to be taken into account. The exact explicit solutions being prohibitive in terms of computational complexity, the construction of approximations can be an interesting alternative [11]. Different approaches emerged in the last decade for an optimisation over the control policies instead of an optimisation over the control actions, thus leading to attractive robust formulation [12], [13]. Tube MPC is another approach to this complex robust control problem [14] and is somehow connected to the output feedback MPC studies [15]. Furthermore, we note here the fact that the input to state stability concepts were adapted to robust MPC context in the recent studies [16], [17] with implications to the systems/control law presenting discontinuities. We have thus the picture of a growing interest for the robustness issues related to the MPC synthesis.

Pedro Rodriguez-Ayerbe and Sorin Olaru are with the Control Department SUPELEC (E3S), 91190 Gif sur Yvette, France pedro.rodriguez@supelec.fr, sorin.olaru@supelec.fr

In the present paper we will approach this problem in a slightly different manner, close to the construction of an estimation mechanism [1] for the constrained variables. In [1], a robust control structure is obtained but the parametric optimisation remains intricate as long as the feasible domains are not polytopic. A first study regarding the possible robustness improvement for the explicit affine feedback policy constructed upon predictive control strategy for linear systems was presented in [18]. The simplest way to proceed is to consider an observer of the state variables [1]. The use of an observer preserves the dimension of the state space and by consequence the piecewise structure of controller. An interesting feature is the fact that the same observer can be used over the entire domain independently of the active region of the controller. We note that the observer can also be considered as a noise characterization for the prediction model. Nevertheless, the observer does not allow spanning the entire space of stabilizing controllers.

The present paper introduces an improved result based on the Youla-Kučera parametrization which spans the space of stabilizing controllers. For a two-degree of freedom controller, one has access to all the stabilizing controllers that preserve the same input/output behavior, so the Youla-Kučera parameter offers more degrees of freedom than the use of an observer. The robustification is made such that the state space dimension of the controller is augmented. The direct consequence is that the use of the same parameter in each region is not possible. The continuity between critical regions can be lost with severe degradation in stability and performances. The main contribution here is the reconstruction of the noise model induced by the Youla-Kučera parameter for the unconstrained case, and its use for the generation of the robust piecewise controller corresponding to the constrained MPC case.

In the following, section 2 briefly recalls the constrained MPC control, and the explicit solution to the associated parametric optimisation problem. Section 3 considers the robustification of a linear controller using the Youla-Kučera parameter and the equivalent disturbance model. Numerical examples are presented in section 4 and the final conclusions is drawn in section 5.

II. CONSTRAINED MPC

A. From receding horizon control problem to the QP/LP formulation

The design of a predictive control law is based on the existence of an analytic/simulation model of the system to be controlled. In the linear time invariant framework, consider the state space model:

$$\begin{array}{l} x_{t+1} = Ax_t + Bu_t \quad t \in Z^+ \\ y_t = Cx_t + Du_t \end{array} \tag{1}$$

with $x_t,\ x_{t+1} \in \mathbb{R}^n$ the state vector at time t and $t+T_e$ respectively, $u_t \in \mathbb{R}^m$ the control vector at time t; A and B matrices of adequate dimensions and the pair (A,B) assumed to be stabilisable.

At each sampling time, the current state vector (assumed to be measurable) $x_t = x_{t|t}$ is used to elaborate the open loop optimal control sequence $\mathbf{u}^*$:

$$\mathbf{u}_t^* = \left[\begin{array}{ccc} u'_{t|t} & \cdots & u'_{t+N-1|t} \end{array} \right]' \tag{2}$$

with respect to a given cost function:

$$\mathbf{u}_t^* = arg\min_{\mathbf{u}_t} \quad \|Px_{t+N|t}\|_p + \sum_{k=0}^{N-1} \{\|Qx_{t+k|t}\|_p + \|Ru_{t+k|t}\|_p\} \tag{3}$$

where $\|.\|_p$ represents the norm $p = \{1,2,\infty\}$ and the pair (Q,A) is assumed to be detectable. The prediction horizon N, the weighting terms $Q = Q' \geq 0$, $R = R' > 0$ and the final cost defined by $P = P' \geq 0$ are the tuning knobs of the control law.

The optimisation of this cost function is performed subject to constraints imposed by the system dynamic, the functional constraints and terminal or stability constraints:

$$\left\{ \begin{array}{l} x_{t+k+1|t} = Ax_{t+k|t} + Bu_{t+k|t} \quad k \geq 0 \\ H_x x_{t+k|t} + H_u u_{t+k|t} \leq \gamma, 0 \leq k \leq N, \\ x_{t+N|t} \in X_N \end{array} \right. \tag{4}$$

It is considered in the following that all constraints in (4) are of polyhedral type. The finite set of constraints can be restructured to obtain a compact formulation:

- Case p = 2:

$$\begin{array}{l} \mathbf{u}_t^* = arg\min_{\mathbf{u}_t} 0.5\mathbf{u}_t' H\mathbf{u}_t + x'F\mathbf{u}_t \\ \texttt{subjet to:} \quad G\mathbf{u}_t \leq W + Sx \end{array} \tag{5}$$

- Case p = 1,∞:

$$\begin{array}{l} \mathbf{z}^* = arg\min_{\mathbf{z}} c^T \mathbf{z} \\ \texttt{subjet to:} \quad G\mathbf{z} \leq W + Sx \end{array} \tag{6}$$

  with $\mathbf{z} = \{\mathbf{u}_t; \xi_1, \ldots, \xi_{N_\xi}\}$ and $\xi_1, \ldots, \xi_{N_\xi}$ auxiliary variables, the number N_ξ of these variables depending on the optimisation horizon and the prediction model [19].

For both cases (5) - (6), the optimal argument includes the control sequence $\mathbf{u}_t^*$. Only the first part of this sequence is applied effectively to the system input, the complete procedure is reiterated at the next sampling time according to the receding horizon principle [20]. Real time implementation is usually performed through on-line optimisation procedures (linear or quadratic programming) in order to determine the optimum corresponding to a particular value of the state vector x.

In the following section we concentrate on the explicit formulations for the predictive control law. We focus on the quadratic case by exploiting the uniqueness and continuity of the solution in this case. One should note that the same results can be obtained for the LP formulations as long as a continuous selection is assured for the optimal solution see e.g. [21], [22].

B. Explicit solution for quadratic case

The analytic solution of (5) - (6) can be constructed along the lines of sensitivity analysis for parametric optimisation problems (see [2] for a review of the control problems under these framework). The optimal solution will be expressed as an explicit function of the state vector x.

$$f : \mathbb{R}^n \to \mathbb{R}^m \texttt{ so that } u_t^{MPC} = f(x_t) \tag{7}$$

For the quadratic cost functions and the analytical solution of the parametric Quadratic Program (8), see for example the review paper [23].

$$\begin{array}{ll} QP(x): & V^*(x) = \frac{1}{2}x'Yx + \min_{\mathbf{z}} \frac{1}{2}\mathbf{u}_t' H\mathbf{u}_t + x'F'\mathbf{u}_t \\ & \texttt{s.t.} \quad G\mathbf{u}_t \leq W + Sx \end{array} \tag{8}$$

Regarding the structure of the multiparametric problems it can be observed that the feasible domain is represented by a parameterized polyhedron. If bounded, then the optimum is given by a convex combination of parameterized vertices. If the optimal solution is not unique (usually the case of linear cost functions (6)), the explicit solution is equivalent to a point to set mapping [21], and the continuity of the solution must be a crucial criterion when implementing the solution. Indeed, a continuous control law avoids discontinuous variations on the control in case of disturbances appearing on the state vector.

The use of a dual representation of the feasible domain and projection mechanisms (see [4] - [24] for details) provides an insight on the topology of the optimisation problems and can be advantageous if there exist unbounded directions due to the fact that the generators representation offers the right tool for their description as well as for the control of the constraints redundancy. Once the explicit solution of (5) - (6) is obtained, we dispose of an analytic description of the control law. Several studies were dedicated to the piecewise affine characterisation ([3], [5], [4], [25]).

Indeed, the explicit predictive control law is described by a collection of piecewise affine function:

$$u_t^{MPC} = f(x_t) = \left\{ \begin{array}{lll} L_1 x_t + l_1 & if & x_t \in R_1 \\ \cdots & & \cdots \\ L_k x_t + l_k & if & x_t \in R_k \\ \cdots & & \cdots \end{array} \right. \tag{9}$$

with R_k polyhedral critical regions covering feasible states.

The structure of such a piecewise controller is shown in Fig. 1. Once the look-up table of local laws is available, an

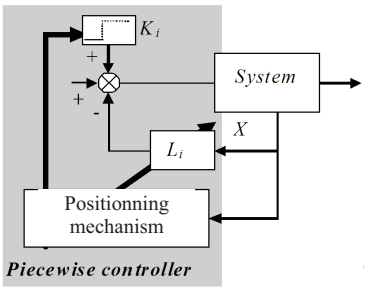


Fig. 1. Piecewise affine controller implementation

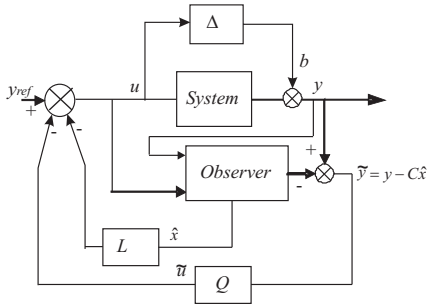


Fig. 2. Robustified controller via the **Q** parameter

efficient positioning mechanism (based on a search tree) can be constructed such that the on-line evaluation routine can find the optimal control action [26].

The implementation of the controller is based on the availability of the current state. For the case when the state is not directly measured, optimal solution is proposed in [27]. Nevertheless, the optimality in this case does not usually justify the complexity of the solution and thus the use of an observer is usually considered [1].

III. YOULA-KUČERA PARAMETRIZTION AND NOISE MODEL

A. Generalities

This paragraphe proposes the obtention of the disturbance model through the synthesis of a Youla-Kučera parameter [28], [29], [30], [31] robustifying the central controller (corresponding to an empty subset of active constraints in the MPC formulation).

The Youla-Kučera parametrization is a well known technique in the literature, and its main advantage is to provide a representation of an entire class of stabilizing controllers. In fact, the Youla parameter, denoted **Q** parameter, establishes a bijection between the class of all stable transfer functions and the class of all stabilizing controllers. If it is inserted in a special way in a closed-loop, the **Q** parameter does not affect the I/O system behavior, but it allows modifying the sensitivity functions in order to improve robustness of the controlled system [31].

Considering a system (1) with a state feedback controller and an observer (10), the **Q** parameter is added as shown in Fig. 2. L is a static feedback gain, and **Q** is a dynamical system. Applying the small gain theorem, robustification of this structure towards unstructured uncertainties, as the additive one shown in Fig. 2, can be performed by minimization of H_∞ norm of the transfer seen by the uncertainty. This problem is convex and can be solved by linear programming or LMI [30], [32].

$$\begin{aligned} \hat{x}_{t+1} &= A\hat{x}_t + Bu_t + K(y_t - C\hat{x}_t) \\ u_t &= -L\hat{x} - \tilde{u}_t \end{aligned} \tag{10}$$

In the case of the uncertainty shown in Fig. 2, the robustification can be expressed as:

$$\min_{\mathbf{Q}\in\mathbb{R}H_\infty} \|WT_{ub}\|_\infty \tag{11}$$

where $\mathbb{R}H_\infty$ is the space of stable transfers, W is a weighting transfer function that reflects the frequency ranges where model uncertainties are more important, and T_{bu} represents the transfer between b and u in Fig. 2.

Other class of specifications, as for example temporal templates, can be used for the synthesis of the **Q** parameter, see e.g. [31], [30], [32] for details.

B. Disturbance model of Q parameter

Considering a state space (A_Q, B_Q, C_Q, D_Q) representation for the **Q** parameter:

$$\begin{cases} x_{Q_{t+1}} = A_Q x_{Q_t} + B_Q \tilde{y}_t \\ \tilde{u}_t = C_Q x_{Q_t} + D_Q \tilde{y}_t \\ \tilde{y}_t = y_t - C\hat{x}_t \end{cases} \tag{12}$$

The controller obtained with a state feedback gain L, an observer (10) and a **Q** parameter is:

$$\begin{aligned} \begin{bmatrix} \hat{x}_{t+1} \\ x_{Q_{t+1}} \end{bmatrix} &= A_{c_{QL}} \begin{bmatrix} \hat{x}_t \\ x_{Q_t} \end{bmatrix} + B_{c_{QL}} y_t \\ u_t &= C_{c_{QL}} \begin{bmatrix} \hat{x}_t \\ x_{Q_t} \end{bmatrix} - D_Q y_t \end{aligned} \tag{13}$$

and:

$$\begin{aligned} A_{c_{QL}} &= \begin{bmatrix} A - KC - BL + BD_QC & -BC_Q \\ -B_QC & A_Q \end{bmatrix} \\ B_{c_{QL}} &= \begin{bmatrix} K - BD_Q \\ B_Q \end{bmatrix} C_{c_{QL}} = \begin{bmatrix} -L + D_QC & -C_Q \end{bmatrix} \end{aligned} \tag{14}$$

In order to obtain the disturbance model corresponding to the **Q** parameter, we consider an augmented model of (1). We consider that the unknown dynamics and uncertainties in the model are represented by a noise model. Considering an innovation representation [33] we obtain:

$$\begin{cases} x_{t+1} = Ax_t + Bu_t + Ke_t \\ y_t = Cx_t + e_t \end{cases} \tag{15}$$

e_t representing a filtered white noise:

$$\begin{cases} x_{v_{t+1}} = A_v x_{v_t} + B_v v_t \\ e_t = C_v x_{v_t} + v_t \end{cases} \tag{16}$$

with v_t a zero mean white noise. The extended model is:

$$\begin{aligned} x_{e_{t+1}} &= A_e x_{e_t} + B_e u_t + K_e v_t \\ y_t &= C_e x_{e_t} + v_t \end{aligned} \tag{17}$$

where:

$$x_e = \begin{bmatrix} x \\ x_v \end{bmatrix} A_e = \begin{bmatrix} A & KC_v \\ 0 & A_v \end{bmatrix} \\ B_e = \begin{bmatrix} B \\ 0 \end{bmatrix} K_e = \begin{bmatrix} K \\ B_v \end{bmatrix} C_e = \begin{bmatrix} C & C_v \end{bmatrix} \tag{18}$$

The system is partially controllable, A_v describing non controllable but observable modes. The predictive control law can be reformulated upon this new prediction model by maintaining the same cost function and constraints. The new pQP is given by:

$$\begin{array}{l} QP_e(x)\text{: } V^*(x_e) = \frac{1}{2}x_e' Y_e x_e + \min_{\mathbf{u}} \frac{1}{2}\mathbf{u}' H_e \mathbf{u} + x_e' F_e' \mathbf{u} \\ \texttt{s.t.} \quad G\mathbf{u} \leq W + S_e x_e \end{array} \tag{19}$$

The different matrices and vectors of (19) can be decomposed in two parts; one dependent on the x, the controllable part, and a second one dependent on x_v, the non controllable part:

$$\begin{array}{l} F_e = [F \quad F_v] \\ H_e = H \quad \texttt{because } A_v \texttt{ is non controllable} \\ S_e = [S \quad S_v] \end{array} \tag{20}$$

With this decomposition, the solution of the pQP can be splited in two parts, one dependent on x another dependent on x_v. It must be noted that the solution dependent on x is the same as the one considered in (9). The optimum without constraints is $u_t = -H^{-1}Fx - H^{-1}F_v x_v = -Lx - L_v x_v = -L_e x_e$. Considering that x and x_v are not measured but observed, the following observer is used:

$$\begin{array}{l} \hat{x}_{e_{t+1/t}} = A_e \hat{x}_{e_{t/t-1}} + B_e u_t + K_1(y_t - C_e \hat{x}_{e_{t/t-1}}) \\ \hat{x}_{e_{t/t}} = \hat{x}_{e_{t/t-1}} + K_2(y_t - C_e \hat{x}_{e_{t/t-1}}) \end{array} \tag{21}$$

We consider the general case of an estimator observer, as the case of one predictor observer is obtained for $K_2 = 0$. The control signal becomes $u_t = -L\hat{x} - L_v \hat{x}_v = -L_e \hat{x}_e$. The obtained controller takes the form:

$$\begin{array}{l} \hat{x}_{e_{t+1/t}} = A_{cn} \hat{x}_{e_t} + B_{cn} y_t \\ u_t = C_{cn} \hat{x}_t - L_e K_2 y_t \end{array} \tag{22}$$

and:

$$\begin{array}{l} A_{cn} = (A_e - K_1 C_e - B_e L_e (I - K_2 C_e)) \\ B_{cn} = (K_1 - B_e L_e K_2) \quad C_{cn} = -(L_e (I - K_2 C_e)) \end{array} \tag{23}$$

The idea thereafter is to find the disturbance model (A_v, B_c, C_v) in order to obtain the equivalence (13)≡(22). This equivalence is obtained by satisfying the following equations:

$$D_Q = L_e K_2 \tag{24}$$

$$C_{c_{QL}} = C_{cn} \tag{25}$$

$$B_{c_{QL}} = B_{cn} \tag{26}$$

$$A_{c_{QL}} = A_{cn} \tag{27}$$

From (25) we obtain $C_Q = L_v - D_Q C_v$, from (26) $K_1 = \begin{bmatrix} K^T & B_Q^T \end{bmatrix}^T$, and from (27) $B_Q = B_v$ and $A_Q = A_v - B_v C_v$.

To resume: the disturbance model (A_v, B_v, C_v) corresponding to a state feedback L, a predictor observer gain K and a **Q** parameter (A_Q, B_Q, C_Q, D_Q) is given by $B_v = B_Q$ and the (A_v, C_v) solution of the following nonlinear equation system:

$$\begin{cases} A_Q - A_v + B_v C_v = 0 \\ C_Q - L_v + D_Q C_v = 0 \\ L_v = H^{-1} F_v(A_v, C_v) \end{cases} \tag{28}$$

Using this principle in the construction of robustified controller one can note that the dependence $F_v(A_v, C_v)$ depends on the nature of the central controller. $F_e = [F \quad F_v]$ in (19) is constructed with A_e, B_e, C_e and thus depends non linearly on matrices (A_v, C_v). This dependence can be further detailed explicitly in certain cases, but, as we don't want to restrict the range of the present study, we remark this non linear dependence (for controllers synthesized with infinite horizon and finite horizon).

The problem (28) can be solved using non linear optimisation techniques. It must be noted that given the nonlinear structure, the existence and uniqueness of (28) are not proved in the general case, but feasibility certificates can be obtained with classical optimization routines.

IV. EXAMPLE

In order to fix the ideas a simple system with constraints on the control action is considered:

$$H(q^{-1}) = \frac{y_t}{u_t} = \frac{0.1q^{-1}}{1 - 0.9q^{-1}} \tag{29}$$

Adding an integral action for step disturbances rejection, the model becomes:

$$H(q^{-1}) = \frac{y_t}{\Delta u_t} = \frac{0.1q^{-1}}{(1 - q^{-1})(1 - 0.9q^{-1})} \tag{30}$$

The state space representation of the model retained is:

$$\begin{array}{l} \begin{bmatrix} x_{1_{t+1}} \\ x_{2_{t+1}} \end{bmatrix} = \begin{bmatrix} 0.9 & 0.1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{1_t} \\ x_{2_t} \end{bmatrix} + \begin{bmatrix} 0.1 \\ 1 \end{bmatrix} \Delta u_t \\ y_t = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_{1_t} \\ x_{2_t} \end{bmatrix} \end{array} \tag{31}$$

We consider the following cost function

$$J = \sum_{k=N_1}^{N_2=5} (x_k' Q x_k) + \sum_{k=1}^{N_u} (\Delta u_k' R \Delta u_k) \quad Q \geq 0 \quad R > 0 \tag{32}$$

With $N_1 = 1, N_2 = 5, N_u = 2, R = 1, Q = C'C$. We take this Q in order to consider the output in the criteria. The constraints are:

$$|u| < u_{max} = 4 \quad |x_1| < 4 \quad |x_2| < u_{max} \tag{33}$$

The matrices of the obtained mpQP problem are:

$$\begin{array}{l} H = 2\Gamma' Q_m \Gamma + R_m \\ F' = 2\Theta' Q_m \Gamma \\ Y = \Theta' Q_m \Theta \\ G = \begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 1 & 1 \\ -1 & -1 \end{bmatrix} W = \begin{bmatrix} u_{max} \\ u_{max} \\ u_{max} \\ u_{max} \end{bmatrix} S = \begin{bmatrix} 0 & -1 \\ 0 & 1 \\ 0 & -1 \\ 0 & 1 \end{bmatrix} \end{array} \tag{34}$$

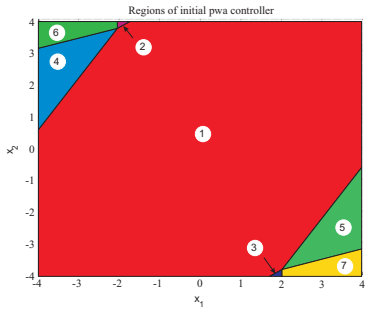


Fig. 3. Regions of initial PWA controller

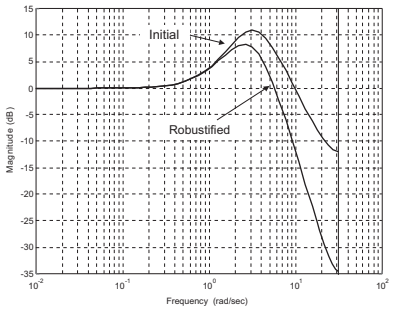


Fig. 4. T_{bu} transfer of initial and robustified central controllers

TABLE I
INITIAL PWA CONTROLLER

Region	L	l
1	$[0.5720 \quad 0.2484]$	0
2	$[0 \quad 1]$	4
3	$[0 \quad 1]$	-4
4	$[0.1516 \quad 0.5084]$	1.8351
5	$[0.1516 \quad 0.5084]$	-1.8351
6	$[0 \quad 1]$	4
7	$[0 \quad 1]$	-4

TABLE II
ROBUSTIFIED PWA CONTROLLER

Region	$L_e = [L \quad L_v]$	l
1	$[L_1 \quad -0.5169]$	l_1
2	$[L_2 \quad 0]$	l_2
3	$[L_3 \quad 0]$	l_3
4	$[L_4 \quad -0.8375]$	l_4
5	$[L_5 \quad -0.8375]$	l_5
6	$[L_6 \quad 0]$	l_6
7	$[L_7 \quad 0]$	l_7

With:

$$\Theta = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^{N_2} \end{bmatrix} \quad \Gamma = \begin{bmatrix} B & 0 \\ AB & B \\ \vdots & \vdots \\ A^{N_2-1}B & A^{N_2-2}B \end{bmatrix} \tag{35}$$
$$Q_m = diag(Q,Q,Q,Q,Q) \quad R_m = diag(R,R)$$

The obtained pQP problem has been solved using MPT Toolbox for Matlab [34]. The obtained PWA controller is shown in Fig. 3. A controller of 7 regions is obtained. Each region and the obtained controller is summarized in table 1.

If the state is not measured, an observer is considered. We have considered a predictor observer, that is (10) with K, in order to place the poles of the observer in $[0.6 \quad 0.7]$. These poles have been chosen in order to have an observer faster than the closed loop with total information. The corresponding transfer T_{bu} is shown in Fig. 4. This transfer represents the sensitivity of the controlled system towards the additive unstructured uncertainties. The lower this transfer is, the bigger the accepted uncertainty without lost of stability is.

The robustification of the obtained central controller towards unstructured uncertainties gives $Q = \frac{-0.5191q^{-1}}{1-0.8q^{-1}}$. That is $A_Q = 0.8, B_Q = 1, C_Q = 0.5191, D_Q = 0$. We have found a **Q** parameter with $D_Q = 0$, in order to keep a predictor observer, and of degree 1 in order to have an easier visualization of regions in the example. The T_{bu} transfer obtained with this Q parameter is shown in Fig. 4. As it can be observed, the robustification towards additive unstructured uncertainties is improved. The disturbance model corresponding to this Q parameter is obtained solving the following optimisation problem:

$$\begin{aligned} &A_Q - A_v + B_v C_v = 0 \\ &C_Q - L_v + D_Q C_v = 0 \\ &L_v = H^{-1} F_v(A_v, C_v) \end{aligned} \tag{36}$$

$F_v(A_v, C_v)$, dependent on A_v and C_v, is obtained as follows:

$$\begin{aligned} &F_e' = 2\Theta_e' Q_{e_m} \Gamma_e = [F' \quad F_v'] \\ &Q_{e_m} = diag(Q_e, Q_e, Q_e, Q_e, Q_e) \\ &Q_e = C_e' C_e \quad C_e = [C \quad C_v] \end{aligned} \tag{37}$$

Θ_e' and Γ_e are obtained with (35) using A_e and B_e of (18).

The solution of this non linear programming gives $A_v = -0.520, B_v = B_Q = 1, C_v = -1.3208$. The initial PWA controller can be modified according to this disturbance model. The obtained PWA controller is shown in Fig. 5 and is summarized in table 2.

The obtained PWA controller is non defined over a convex region of the augmented state space. In Fig. 5 it can be observed that, if the noise state is bigger than 5, two other regions must be generated.

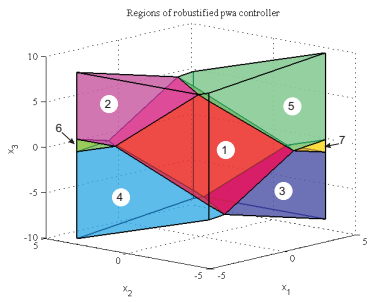


Fig. 5. Regions of robustified PWA controller

V. Conclusions

The paper investigated the robustification methods for the control laws obtained in a constrained predictive control framework. The idea is to design in a first instance a piecewise controller which satisfies the basic demands in terms of tracking performances. In a second stage, the same predictive control structure (prediction horizon, weightings, etc.) is robustified using the model arguments accounting for the noise influence. Its has been shown that the structure of initial PWA controller is maintained. The robustified controller can be obtained from the initial one and the noise model parameters.

The robustification of initial unconstrained controller is made through the Youla-Kučera parametrization, and then this robustification is expanded to all the piecewise structure of the controller. For this, the disturbance model corresponding to the Youla-Kučera parameter is found, and used to regenerate the piecewise controller by preserving the same input/output behavior, but with an increased robustness.

The limitations of the method lay in the existence of the corresponding disturbance model of the Youla-Kučera parameter. This is transparent in the resolution of a non linear equation system. Given that the robustification is done off-line, any infeasibility can be handled by retuning the MPC parameters.

From another point of view, the approach can be seen as an extension of the robustification methods for linear systems to the control laws under constraints.

References

[1] G. Goodwin, M. Seron, and J. DeDona, *Constrained Control and Estimation*. Berlin: Springer-Verlag, 2004.

[2] E. Pistikopoulos, M. Georgiadis, and V. Dua, *Multi-Parametric Model-Based Control: Theory and Applications.* Weinheim, Germany: Wiley-VCH Verlag., 2007.

[3] A. Bemporad, M. Morari, V. Dua, and E. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems." *Automatica*, vol. 38, pp. 3–20, 2002.

[4] S. Olaru and D. Dumur, "A parameterized polyhedra approach for explicit constrained predictive control," *Proceedings of 43rd IEEE Conference on Decision and Control*, 2004.

[5] M. Seron, G. Goodwin, and J. D. Dona, "Characterisation of receding horizon control for constrained linear systems," *Asian Journal of Control*, vol. 5(2), pp. 271–286, 2003.

[6] A. Grancharova, T. Johansen, and P. Tøndel, "Computational aspects of approximate explicit nonlinear model predictive control," in *Assessment and Future Directions of Nonlinear Model Predictive Control*, R. Findeisen, F. Allgöwer, and L. Biegler, Eds. , Germany: LNCIS, Springer-Verlag, 2007, vol. 358, pp. 181–192.

[7] A. Bemporad and C. Filippi, "An algorithm for approximate multiparametric convex programming." *Computational Optimization and Applications*, vol. 35, pp. 87–108, 2006.

[8] E. Kerrigan and J. Maciejowski, "Feedback min-max model predictive control using a single linear program: robust stability and the explicit solution," *International Journal of Robust and Nonlinear Control*, vol. 14, pp. 395–413, 2004.

[9] A. Bemporad, F. Borelli, and M. Morari, "Model predictive control based on linear programming: The explicit solution." *IEEE Transactions on Automatic Control*, vol. 47, pp. 1974–1985, 2002.

[10] S. Olaru and D. Dumur, "A parameterized polyhedra approach for the explicit predicite control," in *Informatics in Control, Automation and Robotics II*, J. Filipe, J.-L. F. J. Cetto, and Carvalho, Eds. Springer, 2007.

[11] A. Grancharova and T. A. Johansen, "Computation, approximation and stability of explicit feedback min-max nonlinear model predictive control," *Automatica*, vol. 45, no. 5, pp. 1134 – 1143, 2009.

[12] C. Løvaas, M. M. Seron, and G. C. Goodwin, "Robust output-feedback model predictive control for systems with unstructured uncertainty," *Automatica*, vol. 44, no. 8, pp. 1933 – 1943, 2008.

[13] P. J. Goulart, E. C. Kerrigan, and J. M. Maciejowski, "Optimization over state feedback policies for robust control with constraints," *Automatica*, vol. 42, no. 4, pp. 523 – 533, 2006.

[14] W. Langson, I. Chryssochoos, S. V. Rakovic, and D. Q. Mayne, "Robust model predictive control using tubes," *Automatica*, vol. 40, no. 1, pp. 125 – 133, 2004.

[15] D. Mayne, S. Rakovic, R. Findeisen, and F. Allgöwer, "Robust output feedback model predictive control of constrained linear systems," *Automatica*, vol. 42, no. 7, pp. 1217 – 1222, 2006.

[16] M. Lazar, "Model predictive control of hybrid systems: Stability and robustness," Ph.D. dissertation, Technische Universiteit Eindhoven, 2006.

[17] D. Limon, T. Alamo, D. Raimondo, D. M. de la Peña, J. Bravo, and E. Camacho, "Input-to-state stability: an unifying framework for robust model predictive control," in *Int. Workshop on Assessment and Future Directions of NMPC*, Pavia, Italy, September, 2008.

[18] S. Olaru and P. Rodriguez-Ayerbe, "Robustification of explicit predicitive control laws," in *Proceedings of IEEE Conference on Decision and Control*, San Diego, 2006, pp. 4556–4561.

[19] L. Zadeh and L. Whalen, "On optimal control and linear programming." *IEEE Trans. Autom. Control*, vol. 7, pp. 45–46, 1962.

[20] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, pp. 789–814, 2000.

[21] S. Olaru and D. Dumur, "On the continuity and complexity of control laws based on multiparametric linear programs," in *Proceedings of IEEE Conference on Decision and Control*, San Diego, 2006, pp. 5465–5470.

[22] J. Spjøtvold, P. Tøndel, and T. A. Johansen, "Continuous selection and unique polyhedral representation of solutions to convex parametric quadratic programs," *Journal of Optimization Theory and Applications*, vol. 134, no. 2, pp. 177–189, 2007.

[23] A. Alessio and A. Bemporad, "A survey on explicit model predictive control," in *Int. Workshop on Assessment and Future Directions of NMPC*, Pavia, September 2008.

[24] S. Olaru and D. Dumur, "Avoiding constraints redundancy in predictive control optimization routines." *IEEE Transactions on Automatic Control*, vol. 50(9), pp. 1459–1466, 2005.

[25] J. Mare and J. DeDona, "Analytical solution of input contrained reference tracking problems by dynamic programming," in *Proceedings of The 44th IEEE Conference on Decision and Control*, Seville, Spain, 2005.

[26] P. Tøndel, T. Johansen, and A. Bemporad, "An algorithm for multiparametric quadratic programming and explicit mpc solutions," *Automatica*, vol. 39(3), pp. 3173–3178, 2003.

[27] T. Perez, H. Haimovich, and G. C. Goodwin, "On optimal control of constrained linear systems with imperfect state information and stochastic disturbances," *Int. J. Robust Nonlinear Control*, vol. 14, pp. 379–393, 2004.

[28] S. Boyd and C. Barratt, *Linear controller desing. Limits of performance.* Prentice Hall, 1991.

[29] B. Kouvaritakis, J. Rossiter, and A. Chang, "Stable generalized predictive control: an algorithm with guaranteed stability." *IEE Proceedings-D*, vol. 139(4), pp. 349–362, 1992.

[30] P. Rodríguez and D. Dumur, "Generalized predictive control robustification under frequency and time-domain constraints." *IEEE Transactions on Control Systems Technology*, vol. 13(4), pp. 577–587, 2005.

[31] J. Rossiter, *Model-based predictive control. A practical approach.* Boca Raton. Florida: CRC Press LLC, 2003.

[32] C. Stoica, P. Rodriguez-Ayerbe, and D. Dumur, "Off-line improvement of multivariable model predictive control robustness," in *46th IEEE Conf. on Decision and Control*, New Orleans, 2007.

[33] K. Åström and B. Wittenmark, *Computer controlled systems. Theory and design (Third Edition).* Englewood Cliffs, N.J.: Prentice Hall, 1997.

[34] M. Kvasnica, P. Grieder, and M. Baotic, *MPT Multi-Parametric Toolbox. Version 2.6.2*. http://control.ee.ethz.ch/ mpt/, 2006.

Modeling and Simulation of a Baker's Yeast Fed-batch Bioprocess

Monica Roman, *Member, IEEE*

***Abstract*— This paper presents the modeling and simulation of a baker's yeast process taking place inside a fed-batch bioreactor. The modeling procedure is based on the pseudo Bond Graph approach, and the simulation is performed by using 20sim modeling and simulation environment. First, a Bond Graph model of a prototype fed-batch bioprocess is obtained, starting from the reactions schemes and taking into account the biochemical phenomena. Then, the method is applied on a complex baker's yeast fed-batch bioprocess, widely used in bio-industry. Several simulation results and comparisons are also presented.**

I. Introduction

THE baker's yeast bioprocess is extensively used in bakery, beer and wine industries. Usually, baker's yeast production is carried out into Continuous Stirred Tank Bioreactors (CSTB) or into Fed-batch Bioreactors (FBB). This bioprocess is highly nonlinear and the reaction rates are quite complicated. However, some nonlinear models were developed in order to describe this process. One of most utilized kinetic model was introduced by Sonnleitner and Käppeli [1], and since then it has been used by many authors [2], [3], [4], [5], [6]. The Sonnleitner and Käppeli kinetic model is based on the well-known bottleneck hypothesis [1], [6], which assumes a limited capacity of yeast, leading to the production of ethanol under conditions of oxygen limitation and/or high glucose concentration. In practice, the analytical models of the reaction rates and/or of the specific growth rates are difficult to obtain.

Modeling of baker's yeast bioprocess is a complex task; however, using the mass balance of the components inside the process and obeying some modeling rules, a dynamical state-space model can be obtained [7], [8]. An alternative to the classical modeling is the pseudo Bond Graph method [9], [10]. This method provides a uniform manner to describe the dynamical behavior of processes. Pseudo Bond Graphs are suitable for chemical systems due to the physical meaning of the effort and flow variables.

The Bond Graph modeling of some biological systems was reported in a few works, such as [11], [12]. Yet, the Bond Graph modeling of bioprocesses is a recent research trend [13], [14], and it is not well exploited. In [14] the Bond graph modeling of a baker's yeast bioprocess which takes place inside CSTB was studied. The present work addresses the pseudo Bond Graph modeling of a baker's yeast bioprocess that is carried out into a FBB. First, the model of a prototype fed-batch bioprocess is obtained using the reaction schemes and taking into account the functioning of the fed-batch bioreactor. Then, in order to outline the applicability of the method, a complex model of baker's yeast bioprocess is developed. This kind of bioreactor is initially partially filled with an amount of some of the needed reactants. The other reactants are then progressively added to the reactor as and when required. The process is stopped when enough products have been accumulated.

The paper is organized as follows. In Section II, the Bond Graph model of a fed-batch prototype bioprocess is developed. Then, Section III deals with the application of the method for the baker's yeast bioprocess. Also, the dynamic models are achieved from the Bond Graph models. Section IV presents several simulation results for both prototype and complex bioprocesses. The Bond Graph models are implemented by using the 20sim modeling and simulation environment (registered trademark of Control-lab Products B.V. Enschede, Netherlands), and time evolution of state variables and reaction kinetic rates are depicted. In order to integrate the systems of differential equations representing the dynamical models, several stiff and non-stiff methods were applied and some comparisons are provided. Finally, in Section V concluding remarks are presented.

II. Pseudo Bond Graph Model of a Prototype Fed-Batch Bioprocess

To model different kind of systems, Bond Graph method uses only nine elements: inertial elements (I), capacitive elements (C), resistive elements (R), effort sources (Se) and flow sources (Sf), transformer elements (TF) and gyrator elements (GY), zero-junctions (J0) and one-junctions (J1). I, C, and R elements are passive elements because they convert the supplied energy into stored or dissipated energy. Se and Sf elements are active elements because they supply power to the system. TF, GY, 0 and 1 junctions are junction elements that serve to connect I, C, R, Se and Sf, and constitute the junction structure of the Bond Graph model. Also, the method uses the effort-flow analogy to describe physical processes. Besides the effort and flow

Manuscript received May 15, 2010. This work was supported by CNCSIS–UEFISCSU, project number PN II-RU PD 108/2010.

M. Roman is with the Department of Automatic Control, University of Craiova, A.I. Cuza no. 13, 200585, Craiova, Romania (phone: +40.251. 438198; fax: +40.251.438198; e-mail: monica@automation. ucv.ro).

variables, two other types of variables are very important in describing dynamic systems: generalized momentum p as time integral of effort and the generalized displacement q as time integral of flow [15]. The pseudo Bond Graph method keeps both the unitary characteristic and some of the basic methodology advantages. Also, it can be adapted to each process particularities.

In biotechnology, pseudo Bond Graph models are accomplished starting with processes reactions schemes and using both base elements of Bond Graph methodology and pseudo bonds with effort and flow variables as concentrations and mass flows.

One of the simplest biological reactions is the micro-organisms growth process, with the reaction scheme [7], [8] given by:

$$S \xrightarrow{\varphi} X, \tag{1}$$

where S is the substrate, X is the biomass and φ is the reaction rate.

This simple growth reaction represents in fact a prototype reaction, which can be found in almost every bioprocess. The dynamics of the concentrations of the components from reaction scheme (1) can be modeled considering the mass balance of the components. The dynamical model of the bioprocess (1) is simple, but if the reaction scheme is more complicated, the achievement of the dynamical model is difficult. In such cases, the Bond Graph method can be a suitable approach. In order to model bioprocesses, pseudo Bond Graph method is more appropriate because of the meaning of variables involved – effort (concentration) and flow (mass flow). From the reaction scheme (1) and taking into account the mass transfer through the fed-batch bioreactor, using the Bond Graph modeling characteristics, the pseudo Bond Graph model of the fed-batch bioprocess is achieved and is depicted in Fig. 1.

The directions of half arrows correspond to the run of the reaction, going out from the substrate S towards biomass X. In the Bond Graph model, the mass balances of the species involved in the bioreactor are represented by two 0-junctions: $0_{1,2,3,4}$ (mass balance for S), $0_{7,8,9}$ (mass balance for biomass X).

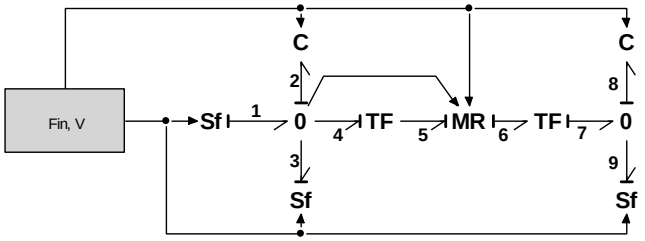


Fig. 1. Pseudo Bond Graph model of the fed-batch prototype bioprocess.

A difficult task is the modeling of the reaction kinetics. The form of kinetics is complex, nonlinear and in many cases unknown. A general assumption [8] is that a reaction can take place only if all reactants are presented in the bioreactor. Therefore, the reaction rates are necessarily zero whenever the concentration of one of the reactants is zero. In order to model the rate of reaction φ, because of the dependency of substrate and biomass concentrations, we have used a modulated two port R element, denoted $MR_{5,6}$. Mass flow of the component entering the reaction is modeled using a modulated source flow element Sf_1 and quantities exiting from the reaction are modeled using modulated flow sources elements Sf represented by bonds 3 and 9. This approach was imposed by the dependency of these elements on F_{in} - the input feed rate (l/h), and on V – volume of the bioreactor (l).

From the constitutive equations of Sf-elements we obtain: $f_3 = e_3 Sf_3$, $f_9 = e_9 Sf_9$.

The accumulations of substrate and biomass in the FBB are represented by bonds 2 and 8, and are modeled using capacitive elements C, with the constitutive equations:

$$e_2 = q_2 / C_2 = (\int_t (f_1 - f_3 - f_4)dt) / C_2, \tag{2}$$

$$e_8 = q_8 / C_8 = (\int_t (f_7 - f_9)dt) / C_8. \tag{3}$$

Using the constitutive relations of transformer elements $TF_{4,5}$ and $TF_{6,7}$ the relations for flows f_4 and f_7 are obtained: $f_4 = k_{4,5} f_5$, $f_7 = f_5 / k_{6,7}$, with $k_{4,5}$ and $k_{6,7}$ the transformers modulus, which are in fact yield coefficients of the bioprocess (their values equal one for this fed-batch bioprocess).

In fact, e_2 is the substrate concentration, which will be denoted with S (g/l), e_8 is the biomass concentration X (g/l), f_5 is proportional to φ and V, $C_2 = C_8 = V$ (l) with $Sf_3 = Sf_9 = F_{in}$.

Therefore, from (2) and (3) we will obtain the dynamical model of the fed-batch bioprocess:

$$\begin{aligned} V\frac{dS}{dt} &= V \cdot \dot{S}(t) = F_{in} S_{in} - F_0 S - \varphi V, \\ V\frac{dX}{dt} &= V \cdot \dot{X}(t) = -F_0 X + \varphi V, \\ \frac{dV}{dt} &= \dot{V}(t) = F_{in}. \end{aligned} \tag{4}$$

The model (4) expresses the equations of mass balance for the reaction scheme (1).

Taking into account that the dilution rate $D = F_{in} / V$, the dynamical behavior of the concentrations can be easily obtained from (4):

$$\dot{S}(t) = DS_{in} - DS - \varphi, \quad \dot{X}(t) = -DX + \varphi, \quad \dot{V}(t) = F_{in}. \tag{5}$$

The reaction rate can be expressed as [7]:

$$\varphi(S,X) = \mu(S,X) \cdot X \,, \tag{6}$$

where μ is the specific growth rate [8]. We will consider the specific growth rate modeled as a Haldane law [16]:

$$\mu(S) = \mu_0 \left(S / (K_m + S + S^2 / K_i) \right), \tag{7}$$

where μ_0 represents the maxim specific growth rate, K_m the Michaelis-Menten constant and K_i the inhibition constant.

III. Modeling of Baker's Yeast Bioprocess

Next, the same procedure as in previous section will be used in order to obtain the Bond Graph model of a baker's yeast fed-batch bioprocess. Living cells of *Saccharomyces cerevisiae*, which form baker's yeast, are predominantly used in bakery, beer and wine industries. Nowadays, with the achievement of modern gene technology, *S. cerevisiae* are increasingly used as host organisms for producing recombinant proteins (production of insulin for diabetics, vaccines, etc.) [6]. Baker's yeast production is carried out into fed-batch fermenters with inoculums of *S. cerevisiae* culture and a glucose solution as substrate feed. Three metabolic pathways can be detected: respirative growth on glucose, fermentative growth on glucose and respirative growth on ethanol. Respirative pathways occur in presence of oxygen and the fermentative one in its absence, associated with ethanol production [1].

The reaction scheme of the baker's yeast production process was introduced by Sonnleitner and Käppeli [1], and since then it has been used by many authors [3], [4], [5], [6]:

$$S + C \xrightarrow{\mu_s^o} X + G \,, \quad S \xrightarrow{\mu_s^r} X + E + G \,, \quad E + C \xrightarrow{\mu_e^o} X + G \,. \tag{8}$$

In the reaction schemes (8), S is the glucose, X the biomass, E the ethanol, C is the dissolved oxygen, and G the dissolved carbon dioxide.

The first reaction scheme represents the respiratory growth on glucose; the second reaction scheme the fermentative growth on glucose, and finally the third reaction represents the respiratory growth on ethanol. μ_s^o, μ_s^r, and μ_e^o are three specific growth rates that reflect the capacity of the baker's yeast to exploit three catabolic pathways for energy and material sources.

In the following, we will consider the fed-batch operation of the bioprocess. From the reaction scheme (8), and considering the mass transfer through the bioreactor, using the modeling procedure described in the previous section, the pseudo Bond Graph model of the bioprocess is achieved.

This model is presented in Fig. 2.

The directions of the half arrows in the Bond Graph correspond to the progress of the reactions, going out from the components S and C towards X and G for the first reaction, from S towards X, E and G for the second reaction, and from E and C towards X and G for the third reaction.

In Bond Graph terms, the mass balances of the species involved in the bioreactor are represented by five 0-junctions: $0_{1,2,3,4,33}$ (mass balance for C), $0_{6,7,8,9,20}$ (mass balance for S), $0_{13,14,15,24,37}$ (mass balance for X), $0_{17,18,19,26,39}$ (mass balance for G), and $0_{28,29,30,31}$ (mass balance for E).

The accumulations of species C, S, X, G and E in the bioreactor are represented by bonds 2, 7, 14, 18 and 29, and are modeled using capacitive elements C.

The constitutive equations of C-elements are obtained using constitutive relations of 0-junction, and they have the following form:

$$\begin{aligned}
e_2 &= q_2 / C_2 = (\int_t (f_1 - f_3 - f_4 - f_{33})dt) / C_2 \,, \\
e_7 &= q_7 / C_7 = (\int_t (f_6 - f_8 - f_9 - f_{20})dt) / C_7 \,, \\
e_{14} &= q_{14} / C_{14} = (\int_t (f_{13} - f_{15} + f_{24} + f_{37})dt) / C_{14} \,, \\
e_{18} &= q_{18} / C_{18} = (\int_t (f_{17} - f_{19} + f_{26} + f_{39})dt) / C_{18} \,, \\
e_{29} &= q_{29} / C_{29} = (\int_t (f_{28} - f_{30} - f_{31})dt) / C_{29} \,,
\end{aligned} \tag{9}$$

where C_2, C_7, C_{14}, C_{18} and C_{29} are the parameters of C-elements: $C_2 = C_7 = C_{14} = C_{18} = C_{29} = V$, with V being the bioreactor volume (l).

The components flows exiting from the reaction are modeled using flow sources elements Sf represented by bonds 3, 8, 15, 19 and 30; the constitutive equations of these elements are:

$$f_3 = e_3 Sf_3 \,, \; f_8 = e_8 Sf_8 \,, \; f_{15} = e_{15} Sf_{15} \,, \tag{10}$$

$$f_{19} = e_{19} Sf_{19} \,, \; f_{30} = e_{30} Sf_{30} \,, \tag{11}$$

with Sf_3, Sf_8, Sf_{15}, Sf_{19}, Sf_{30} parameters of Sf-elements:

$$Sf_3 = Sf_8 = Sf_{15} = Sf_{30} = F_{in} \,, \; Sf_{19} = K_V K_L a V + F_{in} \tag{12}$$

where F_{in} is the input feed rate (l/h), and K_V is a transfer coefficient.

Mass flows of the components entering the reaction are modeled using two flow sources elements Sf_1 and Sf_6, and the transformer elements $TF_{4,5}$, $TF_{9,10}$, $TF_{12,13}$, $TF_{16,17}$, $TF_{20,21}$, $TF_{23,24}$, $TF_{25,26}$, $TF_{27,28}$, $TF_{31,32}$, $TF_{33,34}$, $TF_{36,37}$, $TF_{38,39}$ were introduced to model the yield coefficients $k_i, i = \overline{1,12}$.

For the modeling of the reaction rates φ_1, φ_2, φ_3 we used three modulated multiport R-element, $MR_{11,12}$, $MR_{22,23}$, and $MR_{35,36}$.

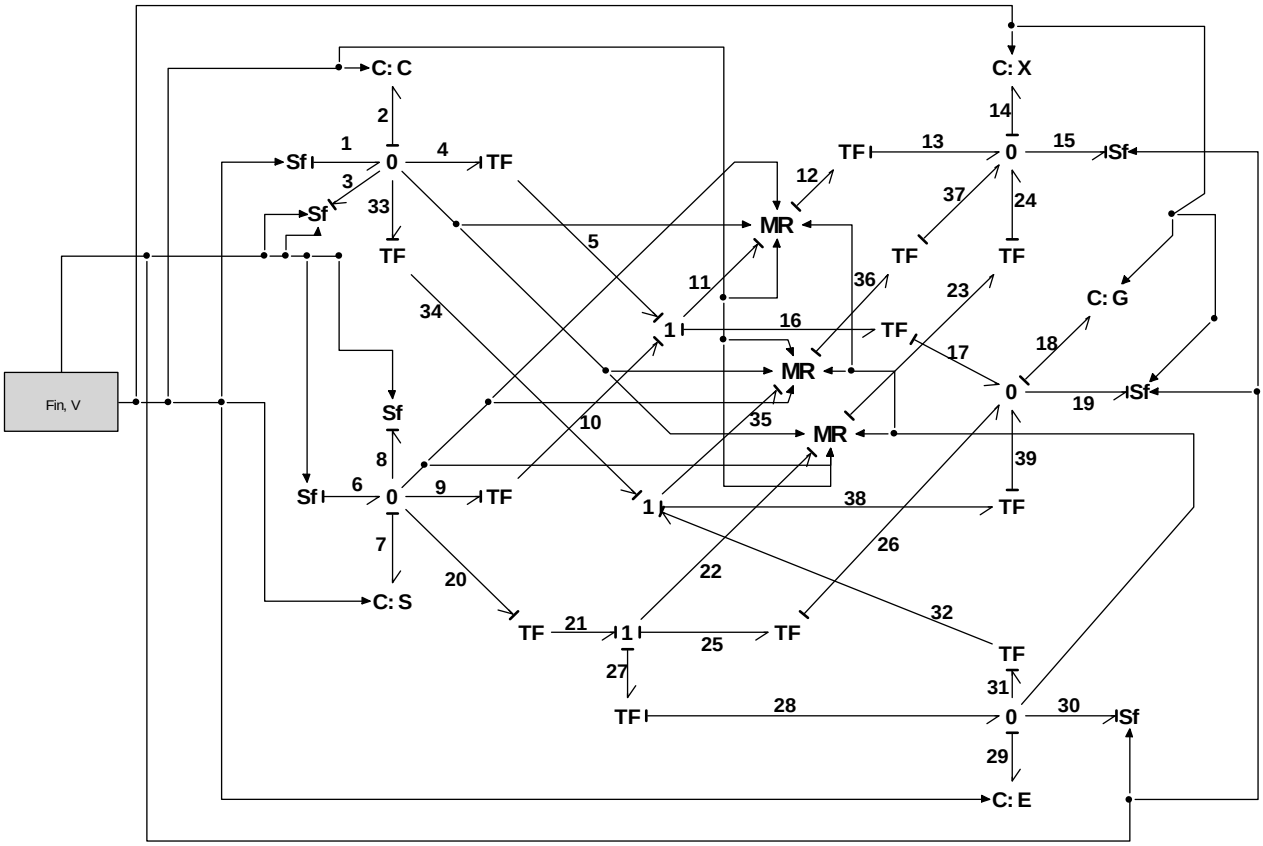


Fig. 2. Pseudo Bond Graph model of the baker's yeast bioprocess.

The constitutive relations of MR elements imply that: $f_{12} = \mu_s^0 Ve_{12}$, $f_{23} = \mu_s^r Ve_{23}$, $f_{36} = \mu_e^0 Ve_{36}$.

The signification of Bond Graph elements is as follows: e_2 is the substrate concentration C (g/l), e_7 - the glucose concentration S (g/l), e_{14} - the biomass concentration X (g/l), e_{18} is the product concentration G (g/l), e_{29} is the ethanol concentration E (g/l), $f_1 = K_L a C^* V$ (where $K_L a$ is the mass transfer coefficient and C^* the equilibrium concentration of the dissolved oxygen), $f_6 = F_{in} S_{in}$ where F_{in} is the input feed rate (l/h), S_{in} is the glucose concentration on the feed (g/l).

Taking into account all these aspects, from (9) we will obtain the dynamical model of the baker's yeast bioprocess:

$$\begin{aligned}
&V \cdot \dot{C}(t) = K_L a (C^* - C) V - F_{in} C - k_1 \varphi_1 V - k_2 \varphi_3 V ,\\
&V \cdot \dot{S}(t) = F_{in} S_{in} - F_{in} S - k_3 \varphi_1 V - k_4 \varphi_2 V ,\\
&V \cdot \dot{X}(t) = k_5 \varphi_1 V - F_{in} X + k_6 \varphi_2 V + k_7 \varphi_3 V ,\\
&V \cdot \dot{G}(t) = k_8 \varphi_1 V - (K_V K_L a V + F_{in}) G + k_9 \varphi_2 V + k_{10} \varphi_3 V ,\\
&V \cdot \dot{E}(t) = k_{11} \varphi_2 V - F_{in} E - k_{12} \varphi_3 V .
\end{aligned} \tag{13}$$

Taking into account that the dilution rate $D = F_{in}/V = 1/t_r$, with t_r - medium residence time, the above equations become:

$$\begin{aligned}
&\dot{C}(t) = K_L a (C^* - C) - DC - k_1 \varphi_1 - k_2 \varphi_3 ,\\
&\dot{S}(t) = DS_{in} - DS - k_3 \varphi_1 - k_4 \varphi_2 ,\\
&\dot{X}(t) = k_5 \varphi_1 - DX + k_6 \varphi_2 + k_7 \varphi_3 ,\\
&\dot{G}(t) = k_8 \varphi_1 - (K_V K_L a + D) G + k_9 \varphi_2 + k_{10} \varphi_3 ,\\
&\dot{E}(t) = k_{11} \varphi_2 - DE - k_{12} \varphi_3 .
\end{aligned} \tag{14}$$

For the reaction rates of the baker's yeast bioprocess there exist many possible models [1], [3], [5], [6]. However, the process of yeast growth on glucose with ethanol production is generally described by the next three metabolic reactions [1]. First, the reaction rate of respiratory growth on glucose, and the associated specific growth rate are [6]:

$$\varphi_1 = \mu_s^0 \cdot X , \quad \mu_s^0 = \min\left(\mu_s , \mu_{c\max} / k_5\right), \tag{15}$$

where the kinetic terms associated with the glucose consumption μ_s and with the respiratory capacity $\mu_{c\max}$ are modeled by Monod-type laws: $\mu_s = q_{s\max} S/(S + K_s)$, $\mu_{c\max} = q_{c\max} C/(C + K_c)$, with $q_{s\max}$ and $q_{c\max}$ the maximal values of the specific growth rates of glucose and oxygen, and K_s and K_c the saturation parameters for glucose and oxygen uptake, respectively.

Second, the reaction rate of the fermentative growth on glucose, and the associated specific growth rate are [6]:

$$\varphi_2 = \mu_s^r \cdot X\ ,\quad \mu_s^r = \max\left(0, \mu_s - \mu_{c\max} / k_5\right). \tag{16}$$

Finally, if the oxidation capacity is sufficiently high to oxidize both ethanol and glucose, then their consumption is possible [5]. Then, the reaction rate of the respiratory growth on ethanol, and the specific growth rate are [6]:

$$\varphi_3 = \mu_e^0 \cdot X\ , \tag{17}$$

$$\mu_e^0 = \max\left(0, \min\left(\mu_e, (\mu_{c\max} - k_5\mu_s) / k_6\right)\right), \tag{18}$$

where μ_e is the potential ethanol oxidative rate, modeled by Monod-type law: $\mu_e = q_{e\max} E/(E + K_e)$, with $q_{e\max}$ the maximal value of the ethanol specific growth rate, and K_e the saturation parameter for growth on ethanol.

IV. Simulation results

The obtained Bond Graph models of the prototype and baker's yeast bioprocesses were simulated using 20sim environment.

For the simulation of the fed-batch operation a typical time profile of the input feed rate is considered - see Fig.3.

In the same picture, the time profile of the bioreactor volume was depicted. The simulation results of the time evolution of reaction rate, and of substrate and biomass concentrations respectively are shown in Figs. 4 and 5.

For the simulation of the prototype fed-batch bioprocess the following parameters were used: $\mu_0 = 6$ (h^{-1}), $K_M = 10$ (g/l), $K_i = 100$ (g/l), $S_{in} = 100$ (g/l).

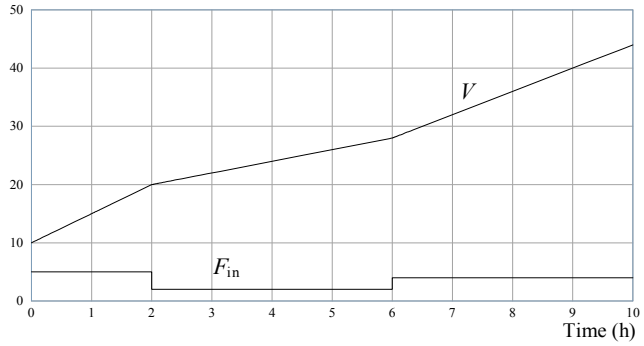


Fig. 3. Time profiles of the input feed rate (l/h) and of the volume (l).

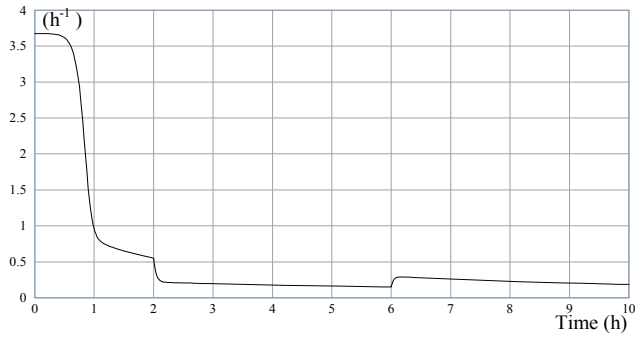


Fig. 4. Evolution of specific growth rate μ.

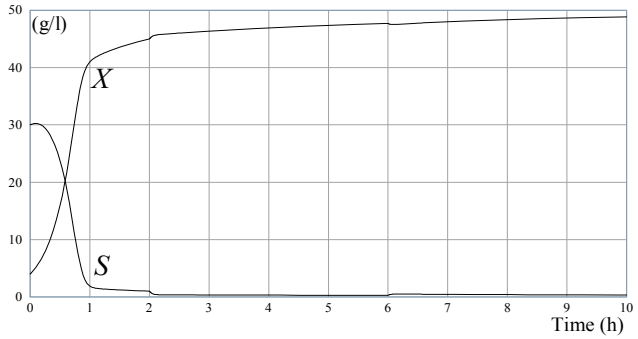


Fig. 5. Time evolution of concentrations - prototype fed-batch bioprocess.

For the case of baker's yeast bioprocess the simulations were performed for the parameters given by [1]. The values of these parameters were obtained from real experiments. However, the obtained Bond Graph model is a qualitative one, which allows the development of some simulation experiments.

Fig. 6 shows the feed rate profile imposed to the process. As shown in Fig. 7, the specific growth rates related to ethanol production and consumption switch alternatively from zero to positive values and are not at any instant simultaneously positive. Fig. 8 depicts the time evolution of specific growth μ_s^r. The profiles the state variables are plotted in Figs. 9, 10 and 11.

For the simulation of Bond Graph models, several non-stiff and stiff integration methods were used. After performing a lot of numerical simulations, the conclusion is that the non-stiff methods do not provide good results.

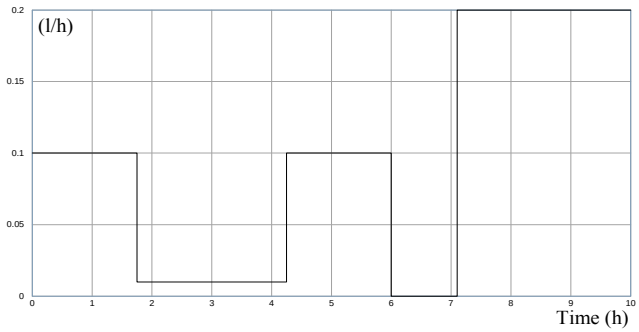


Fig. 6. Time profiles of the input feed rate – baker's yeast bioprocess.

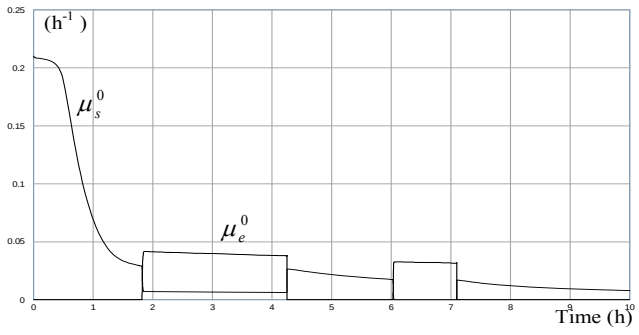


Fig. 7. Evolution of specific growth rates μ_s^0, and μ_e^0.

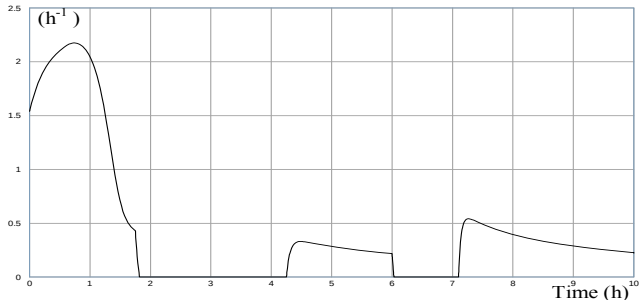


Fig. 8. Evolution of specific growth rate μ_s^r

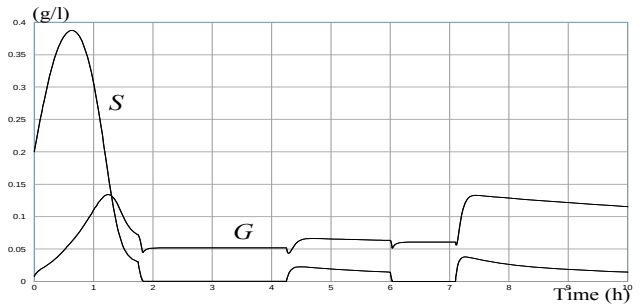


Fig. 9. Substrate and dissolved carbon dioxide concentrations.

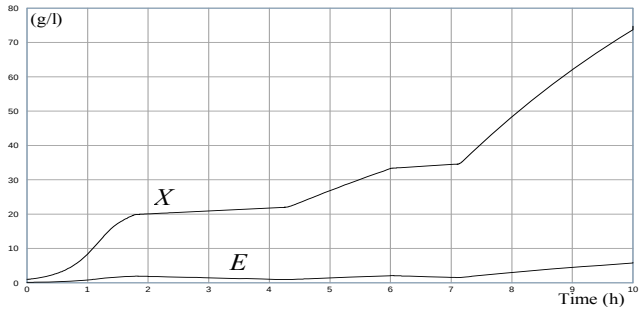


Fig. 10. Evolutions of biomass and ethanol concentrations.

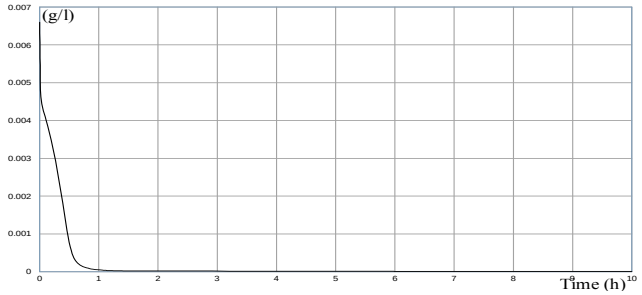


Fig. 11. Time profile of the dissolved oxygen concentration.

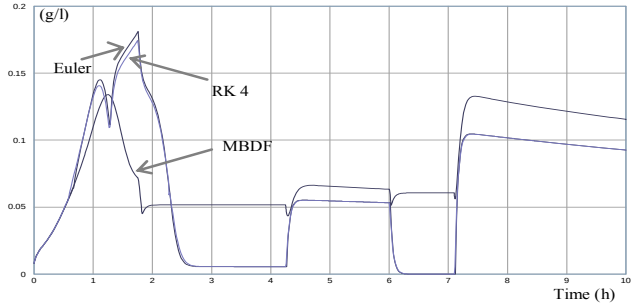


Fig. 12. The profiles of dissolved carbon dioxide concentration – different integration methods.

For example, in Fig. 12 the time profiles of one of the concentrations (dissolved carbon dioxide) obtained with three different methods were provided (non-stiff: Euler, Runge-Kutta 4, and stiff: Modified Backward Differentiation Formula).

V. Conclusion

In this paper, a systematic procedure was used in order to develop the pseudo Bond Graph models of a prototype fed-batch bioprocess and of a nonlinear baker's yeast process, widely used in bioindustry. Also, starting from the pseudo Bond Graph models, the dynamical models were obtained. The simulation of Bond Graph models by comparing the results obtained using different integration methods showed good results for stiff methods.

References

[1] B. Sonnleitner, and O. Käppeli, "Growth of Saccharomyces cerevisiae is controlled by its limited respiratory capacity: Formulation and verification of a hypothesis," *Biotechnology and Bioengineering,* vol. 28, no. 6, pp. 927-937, 1986.

[2] E. C. Ferreira, "Identification and Adaptive Control of Biotechnological Processes," Ph.D. Thesis, Univ. of Porto, 1995.

[3] R. Oliveira, E. C. Ferreira, F. Oliveira, and S. Feyo de Azevedo, "A study on the convergence of observer-based kinetic estimators in fed-batch fermentations," *J. Proc. Contr.*, vol. 6, pp. 367-371, 1996.

[4] V. Lubenova, and E. C. Ferreira, "Adaptive algorithms for estimation of multiple biomass growth rates and biomass concentration in a class of bioprocesses," in Proc. *of 4th Portuguese Conf. on Automatic Control*, 2000, pp. 289-294.

[5] P. Georgieva, A. Ilchmann, and M. F. Weiring, "Modelling and adaptive control of aerobic continuous stirred tank reactors," *European Journal of Control*, vol. 7, no. 5, pp. 476-491, 2001.

[6] F. Renard, and A. Vande Wouwer, "Robust adaptive control of yeast fed-batch cultures," *Computers and Chemical Engineering*, vol. 32, no. 6, pp. 1238-1248, 2008.

[7] G. Bastin, and D. Dochain, *On-line Estimation and Adaptive Control of Bioreactors*. New York: Elsevier, 1990.

[8] D. Dochain (Ed.), *Automatic Control of Bioprocesses*. ISTE Publ. and Wiley & Sons, 2008.

[9] F. Couenne, C. Jallut, B. Maschke, P. C. Breedveld, and M. Tayakout, "Bond graph modelling for chemical reactors," *Math. and Comp. Model. Dynamical Syst.*, vol. 12, no. 2, pp. 159–174, 2006.

[10] C. Heny, D. Simanca, and M. Delgado, "Pseudo-bond graph model and simulation of a continuous stirred tank reactor," *Journal of the Franklin Institute*, vol. 337, pp. 21–42, 2000.

[11] J. Schnakenberg, *Thermodynamics Network Analysis of Biological Systems*. Berlin: Universitext, 1981.

[12] D. A. Linkens, "Bond graphs for an improved modelling environment in the lifesciences," in *Proc. of IEE Colloquium on Bond Graphs in Control*, London, UK, 1990, pp. 3/1–3/4.

[13] M. Roman, E. Bobaşu, E. Petre, D. Selişteanu, and D. Şendrescu, "Pseudo Bond Graph modelling of some prototype bioprocesses," *Control Eng. & Appl. Informatics*, vol. 11, no. 2, pp. 22–29, 2009.

[14] M. Roman, D. Selisteanu, E Bobasu, and D. Sendrescu, "Bond Graph Modeling of a Baker's Yeast Bioprocess," *Int. Conf. on Model., Identif. and Contr.*, Okayama, Japan, July 2010. (accepted)

[15] D. Karnopp and R. Rosenberg, *System Dynamics: A Unified Approach*. New York: John Wiley, 1974.

[16] D. Selişteanu, E. Petre, and V. Răsvan, "Sliding mode and adaptive sliding-mode control of a class of nonlinear bioprocesses," *Int. J. of Adaptive Contr. & Signal Proc.*, vol. 21, no. 8-9, pp. 795-822, 2007.

Train Scheduling with Delay Time Petri Nets and Genetic Algorithms

M. M. Santa, O. Cuibus and T. Leţia

Dept. of Automation Technical University of Cluj-Napoca, Daicoviciu St., *15, 40020, Cluj-Napoca, Romania* (e-mail: Maria.SANTA@aut.utcluj.ro; Octavian.CUIBUS@aut.utcluj.ro; Tiberiu.LETIA@aut.utcluj.ro;)

***Abstract* — The train traffic control problem is approached from the point of view of dynamic scheduling of resource allocation. The asynchronous train arrival times due to unexpected delays or previously unscheduled (charter) trains is considered. The train scheduling problem is solved using genetic algorithms. The chromosome specifies direction and waiting time of each train at its entrance in each interlocking. The railway network model is obtained by modeling the interlockings with Delay Time Petri Nets (DTPNs). The genetic algorithms are used for off-line scheduling of a set of trains in an empty railway structure, and for on-line scheduling adding a new set of trains in a crowded railway structure.**

I. INTRODUCTION

The railway traffic control system is a dynamic one that operates in cooperation with an environment with uncertain properties that include transient and resource overloads, arbitrary arrivals, arbitrary failures and decreases of traffic parameters. Unlike classical real-time control applications that usually concern only the response times to meet the deadlines, railway traffic involves the reasoning about end-to-end timelines and the reaction to events such that the global traffic system fulfills the time requirements.

Despite many uncertainties, the control system is expected to guarantee that all the trains behave according to timelines. Due to environment circumstances (weather conditions, unexpected maintenances required, faults etc.) some trains are delayed. A used method to avoid the train deadline missing is to add to each train laxity times that are used for unexpected delays recovery. These *extra* added times lead to a slower transportation system. Often, even if the train laxity times exist, they are not long enough to avoid the train deadline missing because the delays exceed the reasonable expected necessary safety times.

The current paper solves the Railway Traffic Control (RTC) problem using the dynamic resource allocation. The railway resources are lines, interlockings (combination of switches and traffic lights) and platforms (a special case of lines). They can be allocated *synchronously* (for a specified period of time), *asynchronously* (until the occurrence of an event of release). From another point of view, the allocation an be performed off-line, before a train starts, or on-line, during the system evolution when a train reaches some points. Periodical trains as well as charter trains are taken into account in the current study. The main control problems of the train traffic are analyzed and modeled by Delay Time Petri Nets (DTPN; see [1]) and finally solved using genetic algorithms.

In many application areas, including control systems, fully exploiting the available system resources is crucial to maximizing application performance. Most traditional resource management techniques for real-time systems with multiple resources and multiple users are based on *a priori* characterizations of the expected workload. The algorithms use parameters that are loaded before traffic system starts. At runtime, resources are shared by all tasks according to the pre-established allocations regardless of the dynamics of the control applications, thus implementing an *open loop control*.

II. TRAIN SCHEDULING PROBLEMS

Figure 1 presents a subsystem of a railway network that links 3 railway stations A, B and C. The following specified elements are used: traffic lights (denoted by s1,…,sm), lines (L1-Lj), platforms (special lines, denoted P1-P3, P19-P24), detectors (on each line, in front of each interlocking), interlockings (denoted by I0, …, I*n*), platforms (denoted by P1, …, P*i*) and trains (denoted by T1, …, T*k*). The entrance, the presence or the leaving of trains on the line is signalized by detectors. The railway network is composed of a set of linked resources (lines, platforms, interlockings etc.). The state of a resource can be *reserved*, *occupied* or *released*.

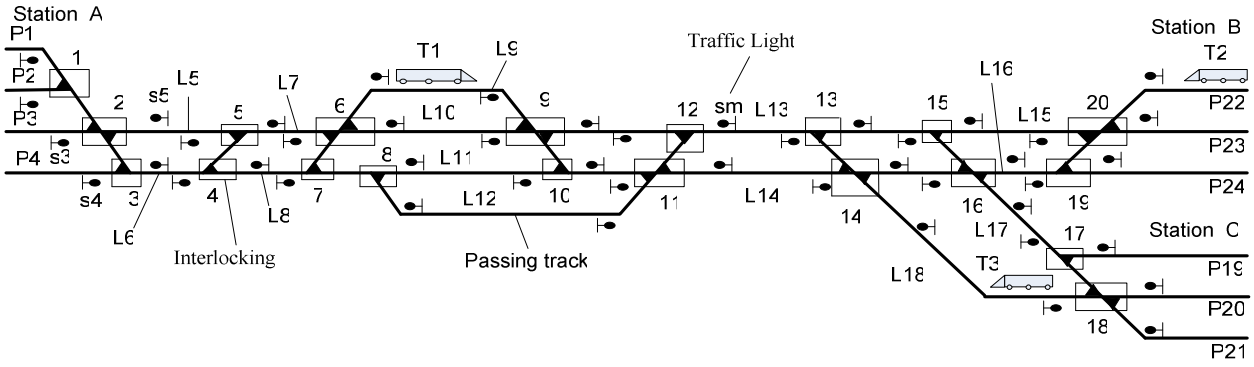


Figure 1. Railway network structure

An *interlocking* is an arrangement of neighboring interconnected sets of switch points and traffic lights such that the train movements through them are performed in a proper and safe sequence. The interlockings are used here to control the train moves in the network and particularly the train entrances on lines. An interlocking cannot be occupied by a train more than it is necessary to cross it.

Once it enters a line, the train moves freely towards the next interlocking. The journey on a line takes a specified duration. If the next line is occupied, the control system delays the train entrance using the appropriate traffic light.

Generally, a *train schedule* specifies time, place and deadlines, both for departure and arrival. The problem considered is defined by the railway layout (Figure 1) and the train table (Table 1).

TABLE 1.
TRAIN TABLE

Train	Departure			Arrival		
	Place	Time	Deadline	Place	Time	Deadline
T1	A.P1	0	9	B.P22	38	40
T3	A.P3	9	18	B.P24	53	55
T5	B.P22	1	10	A.P1	33	40
T6	B.P23	4	13	A.P4	43	47
T9	C.P20	8	17	A.P2	42	45
#T1	A.P1	20	29	B.P22	59	60
#T3	C.P21	15	24	A.P3	55	56

The scheduling problem consists of a set of trains with different departure times. The control systems allocate the resources such that the trains are able to leave the departure places and arrive to destination before their corresponding deadlines. The train scheduling can be performed off-line or on-line. The *off-line scheduling* involves that all the trains are considered unscheduled and the algorithm searches a feasible solution such that all the trains fulfill the timing requirements. If a feasible solution does not exist, then the lower priority trains must wait on passing tracks to permit the higher priority trains to arrive at destination in time. The trains remained on the side tracks are scheduled (without fulfilling the time requirements) when train density has diminished.

The *on-line scheduling* problem considers the case when a train set is already scheduled; they are in movement when one or more trains demand to be scheduled. The new schedule demand can be performed in the off-line manner (considering all the trains unscheduled) searching a feasible solution for all the trains, or the previous scheduled trains move according to the allocated resources and the algorithm search to allocate the available resources to the newly arrived trains without disturbing the routing of previously scheduled trains.

Both approaches use the assumption that the trains can release the tracks at the end of the granted duration. The approach that uses the dynamic resource allocation can lead to better resource utilization, but it is prone to deadlock of trains. This approach can be used only by an algorithm that uses additional strong non-blocking rules [2]. A relevant requirement for any solution is to avoid the train deadlocks. Any solution that leads to a deadlock has to be rejected.

III. RELATED WORKS

New trends of train traffic control and management started since 1997 [3]. An autonomous decentralized train control and management system is proposed to attain both the real-time properties for train control such as the real-time traffic and non-real-time properties for train management. The basic principles of railway traffic control are given in [4]. These include the interlocking utilization, resource management and the partitioning of the railway network into different parts. The train deviations from the scheduled timetable should be removed during the operation [5].

Formal development and verification of a distributed railway control system are performed applying a series of refinement and verification steps in [6]. The distributed train scheduling problem has some similarities with distributed software job scheduling [7], [8]. The concept of collaborative scheduling is also applicable. Each node constructs its local schedule using only local information. The lack of global information makes it impossible for a node to make a globally optimal decision. Thus it is possible for a node to make a scheduling decision that is locally optimal in terms of the utility that can be accrued to the node, but compromises global optimality [2].

An improvement can be obtained using the GPS and wireless communication between train engine and local control center [9]. Some distributed signal control systems based on the Internet technology are also used [10]. A single delayed train can cause a domino effect of secondary delays over the entire network, which is the main concern of planners and dispatchers [11].

The approach in [12] targets a wide class of control systems (linear systems). The optimal resource allocation policy for control tasks that they present in this paper optimally solves the Quality of Control (QoC) scheduling problem formulated by Martí, P [13], but in terms of resource management. The dynamics of the controlled systems as suggested by Yépez, J. [14] and Velasco, M. [15] are the key to better exploiting system resources and improving control systems performance in resource constrained control systems. [15] presents a control-based model for control tasks that allows each control task to trigger itself optimizing computing resources and control performance. Preliminary results of the feedback approach to resource management were reported also by Lin, C. [16].

Tormos, P. [17] presented an application of evolutionary algorithms to solve very complex real-world problems. For this purpose a Genetic Algorithm is designed to solve the Train Timetabling Problem. The railway scheduling problem considered in this work implies the optimization of trains on a railway line that is occupied by other trains with fixed timetables. The timetable for the new trains is obtained with a Genetic Algorithm (GA) that includes a guided process to build the initial population.

The timetable given to each new train must be feasible and it must satisfy a given set of constraints. Among the constraints resulting in this problem, sometimes the

periodicity of the timetables is required. Periodicity leads to the classification as *periodic* and *non periodic* train timetabling. In *periodic timetabling* each trip is operated in a periodic way (each period of the timetable is the same). The mathematical model is called Periodic Event Scheduling Problem (PESP) by Serafini and Ukovich [18]. In PESP a set of repetitive events is scheduled under periodic time window constraints. The PESP model has been used by Kroon and Peeters [19], and Liebchen [20].

Non periodic train timetabling is especially relevant on heavy-traffic, long-distance corridors where the capacity of the infrastructure is limited due to great traffic densities. Many references consider Mixed Integer Problem formulations in which the arrival and departure times are represented by continuous variables and there are binary variables expressing the order of the train departures from each station. The non periodic train timetabling problem has been considered by several authors [21] – [24].

IV. DELAY TIME PETRI NETS MODEL OF RAILWAY TRAFFIC

The DTPN presented in [1] was chosen for modeling the railway traffic. Beside the possibility of modeling the concurrency of train behaviors, the DTPN can completely model the timing features as well.

The formalism of DTPN is presented in detail by Juan et al. in [1]. DTPN were introduced to overcome the problem of preservation of timing constraints during net reduction. It is shown in [1] that DTPN are much more suitable for net reduction. At the simulation stage, net reduction is very important because the number of states explode when interconnecting DTPN models of the interlockings. The novelty here is the use of DTPN for interlocking modeling.

The DTPN represented in Figure 2 models the behavior of a Y shape interlocking, represented in Figure 1, like I1, I3, I4, I5 etc. An Y shape interlocking allows the train to move from *c* to *a* or *b* or from *a* or *b* to *c*.

The following interpretation is associated to the Petri nets nodes and transitions:

- *a*, *b* or *c* – the train is on line *a*, *b* or *c* and waits for the traffic lights *Sa*, *Sb* or respectively *Sc* to be ON;
- I0 – the train is on interlocking I0. By convention, the time needed for a train to pass any interlocking type Y is 1 time unit (this corresponds to location I0 having delay 1);
- *Sa*, *Sb*, *Sc* – the traffic lights that allow trains to leave lines *a*, *b* or *c* and enter the interlocking *I0*. These locations may have tokens only if there is no train in I0.
- *Sa'*, *Sb'*, *Sc'* – the traffic lights that allow trains to leave interlocking I0 and enter lines *a*, *b* or *c* respectively (line must be free !). Note that pairs (*Sa*, *Sa'*) and others are disjunctive. In real life, one traffic light that indicates both permission and direction of travel is modeled by *Sa* and *Sa'* together.
- *Sab* – an additional location, used for storing the direction of travel, to prevent a train from one line to return to the same line after passing interlocking I0; *Sab* also prevents trains from *a* to travel to *b* or vice versa;
- *Ta*, *Tb*, *Tc* – the trains goes from line *a*, *b* or *c* to interlocking I0. Note that a token in *Sa*, *Sb* or respectively *Sc* is needed to execute the transition.
- *Ta'*, *Tb'*, *Tc'* – this transition occurs when a train in interlocking I0 is moving to line *a*, *b* or *c* respectively. Note that a token in *Sa'*, *Sb'* or respectively *Sc'* is needed to execute the transition. In addition, in order to execute *Ta'* or *Tb'*, but not *Tc'*, a token is needed in location *Sab*.
- *Tai*, *Tbi*, *Tci* – the train moves from a previous interlocking (not drawn here) to lines *a*, *b* or *c*. Similarly, *Tao*, *Tbo* and *Tco* corresponds to a train moving from line a, b or c to the next interlocking (not drawn here).

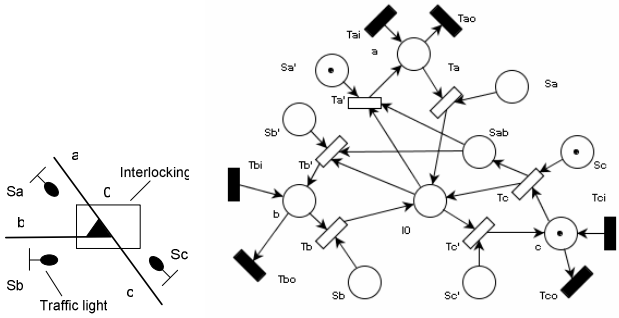


Figure 2. Type Y interlocking and its DTPN model

For example, when a train enters line *c* from a previous interlocking (not drawn here), *Tci* is executed and a token is put in location *c*. Delay of location *c* corresponds to the time that a train needs to travel the length of line *c*. When the traffic light *Sc* goes ON, transition *Tc* becomes executable, which corresponds to the token being extracted from *c* (train leaves line *c*) and injected into I0 (train enters interlocking I0). Also, *Sab* receives a token. The delay of location I0 is 1 time unit, corresponding to the time needed for a train to travel through the interlocking. After that, depending on which traffic light is on, *Sa'* or *Sb'*, transition *Ta'* or *Tb'* is executed (note that *Sab* has a token). This corresponds to the train leaving interlocking I0 and entering line *a* or *b*. At the end, transition *Tao* or *Tbo* is executed – the train leaves the line for the next interlocking, which is not represented here.

The DTPN represented in Figure 3 models the behavior of an X type interlocking, represented in Figure 3, like I2, I6, I9, I11 etc. An X type interlocking makes it possible for the train to move from *a* or *b* to *c* or *d*, or from *c* or *d* to *a* or *b*. By convention, the time needed for a train to pass any interlocking type X is 1 time unit (this corresponds to location I1 having delay 1). The interpretation of locations and transitions of the DTPN is analogue to the type Y interlocking. The locations *Sab* and *Scd* have the following interpretation:

- *Sab* – an additional location that stores the information that the train is traveling from *c* or *d* to *a* or *b*. Note that here, *Sab* is ON when the train comes from *c* or *d* and travels towards *a* or *b*. *Sab* prevents trains coming from *d* (for example) to return to *d* or go towards *c*;
- *Scd* – similar to *Sab*, stores the information that the train is traveling from *a* or *b* towards *c* or *d*;

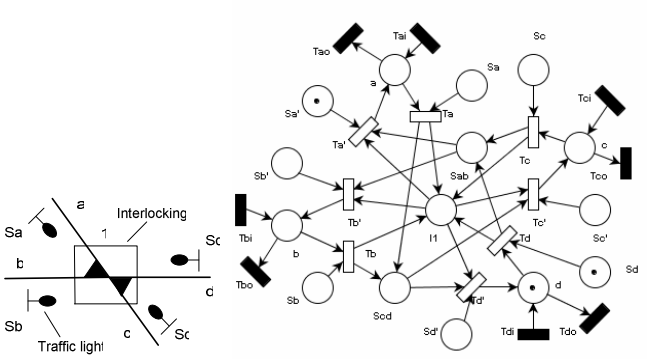


Figure 3. Type X interlocking and its DTPN model

The railway network model was obtained by interconnecting the DTPNs models of the interlockings.

As one can see from Figure 2 and Figure 3, the models of the interlockings are deadlock-free. However, care must be taken when injecting tokens into locations that model the traffic lights (*Sa*, *Sa'*, etc). The control system must control *Sa*, *Sa'*, etc disjunctively, in order to avoid deadlocks.

V. GENETIC ALGORITHMS FOR TRAIN SCHEDULING

A solution to the proposed problem (defined in Figure 1 and Table 1) consists of information about routing for each train, and the amount of delay before entering each line. Table 2 represents the form of the solution on the general case. The table contains a line for each train that has to be scheduled. One column contains the information related to one interlocking. The element *delay* specifies the amount of time a train has to wait before entering the interlocking. The element *selection* expresses the position of switches and traffic lights such that the direction of travel from the interlocking onward is determined. The trains that travel to different destinations may have different selection sets.

The delay has values from 0 to *j*, where *j* is the maximum amount of time that a train can wait on a line. In our case, we considered *j*=7 for all lines and all trains. The selection set contains a set of numbers, corresponding to the direction of the switch that can lead to the destination of the train. If one particular direction cannot lead to the destination, it is not included in the selection set. The direction set can be either: {1,2} if both output directions from the interlocking can lead to the destination, {1} or {2} if only one direction leads to the destination, or even non selection {} when the train is not supposed to arrive at that interlocking. As an example of non selection, train T6 from Table 1, traveling from B.P23 to A.P4 is not supposed to arrive at interlocking I1. Hence, the selection set for T6 of I1 is empty {}.

TABLE 2.

SCHEDULER SOLUTION FOR THE GENERAL CASE

Train	I1		I2		…	In	
	delay	selection	delay	selection		delay	selection
T1	[0,j]	{1,2}	[0,j]	{1}		[0,j]	{2}
T2	[0,j]	{}	[0,j]	{1,2}		[0,j]	{}
Tk	[0,j]	{2}	[0,j]	{}		[0,j]	{1}

In our example, there are 20 interlockings (n=20) and 7 trains to be scheduled (k=7). The maximum number of possible solutions (N) can be calculated as:

$$N = [2 \cdot (j+1)]^{n \cdot k} = 16^{140} \cong 4 \cdot 10^{168}$$

To avoid the long search of a feasible scheduling solution, the genetic algorithms were used. The DTPN model was used to evaluate the fitness function. The genotype is an adaptation of the general solution (Table 2) for our problem. Table 3 represents the part of the genotype corresponding to the trains T1, T6 and #T3.

TABLE 3.

THE CHROMOSOME IN OUR EXAMPLE

Train	I1		I2		…	I20	
	delay	selection	delay	selection		delay	selection
T1	[0,7]	{1}	[0,7]	{1}		[0,7]	{1}
T6	[0,7]	{}	[0,7]	{}		[0,7]	{2}
#T3	[0,7]	{}	[0,7]	{2}		[0,7]	{}

The genetic crossover operation is performed by cutting two chromosomes between genes and binding the opposite parts. The mutation operation is performed by acting on a selected gene (interlocking) by modifying the delay and/or the selected path, but only in the available selection set. In this way, from each interlocking, a train can move only towards the destination. Therefore, each train always arrives at the correct destination. Moreover, mutation is only used on those genes that represent interlockings crossed by trains.

The main objective of the genetic algorithm is to generate a schedule (train/job timetable) where each train/job satisfies time and resource constraints with the lowest computational effort and optimizing a measure of performance.

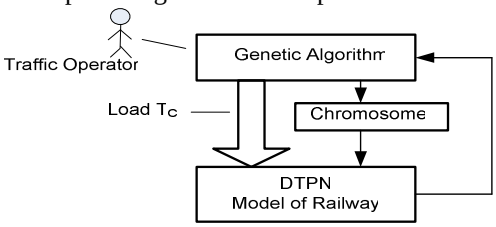


Figure 4. Scheduler structure

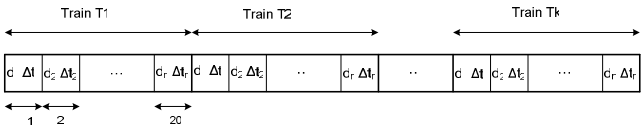


Figure 5. Chromosome representation

The evaluation of a solution is achieved by the fitness function using the scheme presented in Figure 4. This is implemented by the evaluation algorithm presented below, using the notations:

- *resourceTable* is a table storing the states of each resource (line, interlocking) during the horizon time intervals,
- T_C is the set of currently circulating trains, (which cannot be re-scheduled)
- T_N is the set of the new trains that need to be scheduled,
- T_N*_chromosome* is the chromosome to be evaluated,
- *R* is the resource set,
- *F* is the fitness function,
- *H* is the length of the scheduling horizon,
- *v* and *w* are two neighboring concatenated resources,
- *train.resource*(*r*,*t*) refers to the resource *r* required by the mentioned train during the time interval *t*.

Evaluation algorithm for the proposed schedules of the T_N train set:

1: *input:* T_C train set, T_N train set, T_C schedules, *T_N_chromosome*, *resourceTable*;
2: *output:* F(*T_N_chromosome*);
3: initialize *resourceTable* as not reserved;
4: F(*T_N_chromosome*) = 0;
5: simulate the T_N train set evolution using DTPN and mark the required resources in *resourceTable* as reserved
//evaluation of train collision
6: **for** all the trains *x* and *y* from T_C and T_N **do**
7: **for** all the resources *r* from *R* **do**
8: **for** all the time intervals *t* from *H* **do**
9: **if** (*x.resource*(*r*,*t*) == *y.resource*(*r*,*t*)) **then** F(*T_N_chromosome*) = F(*T_N_chromosome*) + collision_penalization;
10: **if** (x.resource(v,t) == y.resource(w,t-1)) & (x.resource(v,t-1) == y.resource(w,t)) **then** F(*T_N_chromosome*)= F(*T_N_chromosome*) + collision_penalization;
//evaluation of train arrival times
11: **for** all the trains *x* from T_N **do**
12: **if** (*x.arrival_time* > *x.deadline*) **then** F(*T_N_chromosome*) = F(*T_N_chromosome*) + *P*·(*x.arrival_time* – *x.deadline*);
else F(*T_N_chromosome*) = F(*T_N_chromosome*) + (*x.deadline* – *x.arrival_time*);
13: **return** F(*T_N_chromosome*);

The algorithm uses the simulation of the DTPN model of train traffic network (line 5). It penalizes the situation in which two trains collide (collision_penalization) (line 9 and 10). If the train arrives at the destination later than the time specified in the train table, it is also penalized (line 12). Deadline missing is penalized much harder, by factor *P*.

VI. TEST AND RESULTS

Two tests using the above-described algorithm have been carried out to solve the train scheduling problem.

The first test performed offline scheduling, as described in chapter 2. It was considered that no trains have been yet scheduled, and the set of trains to be scheduled is {T1, T3, T5, T6, T9}, with their specifications as presented in Table 1. As one can see from Figure 6, the trains were scheduled in such a way as to meet their deadlines. On the vertical axis time units are represented, and on the horizontal axis, the resources (lines – L1-L24 and interlockings – I1-I20) are represented. The continuous oblique line represents the trains traveling across the resource with the specified speed according to the resource specifications. The interlockings are crossed in 1 time unit, whereas lines are covered in 2 to 5 time units (depending on line length). Vertical continuous lines represent trains waiting on a line to travel across an interlocking. Dotted lines are used when a train is not using a resource, only to provide a visual uninterrupted line for each train. The total amount of time that the trains are traveling is 195 time units, out of which 20 are delays.

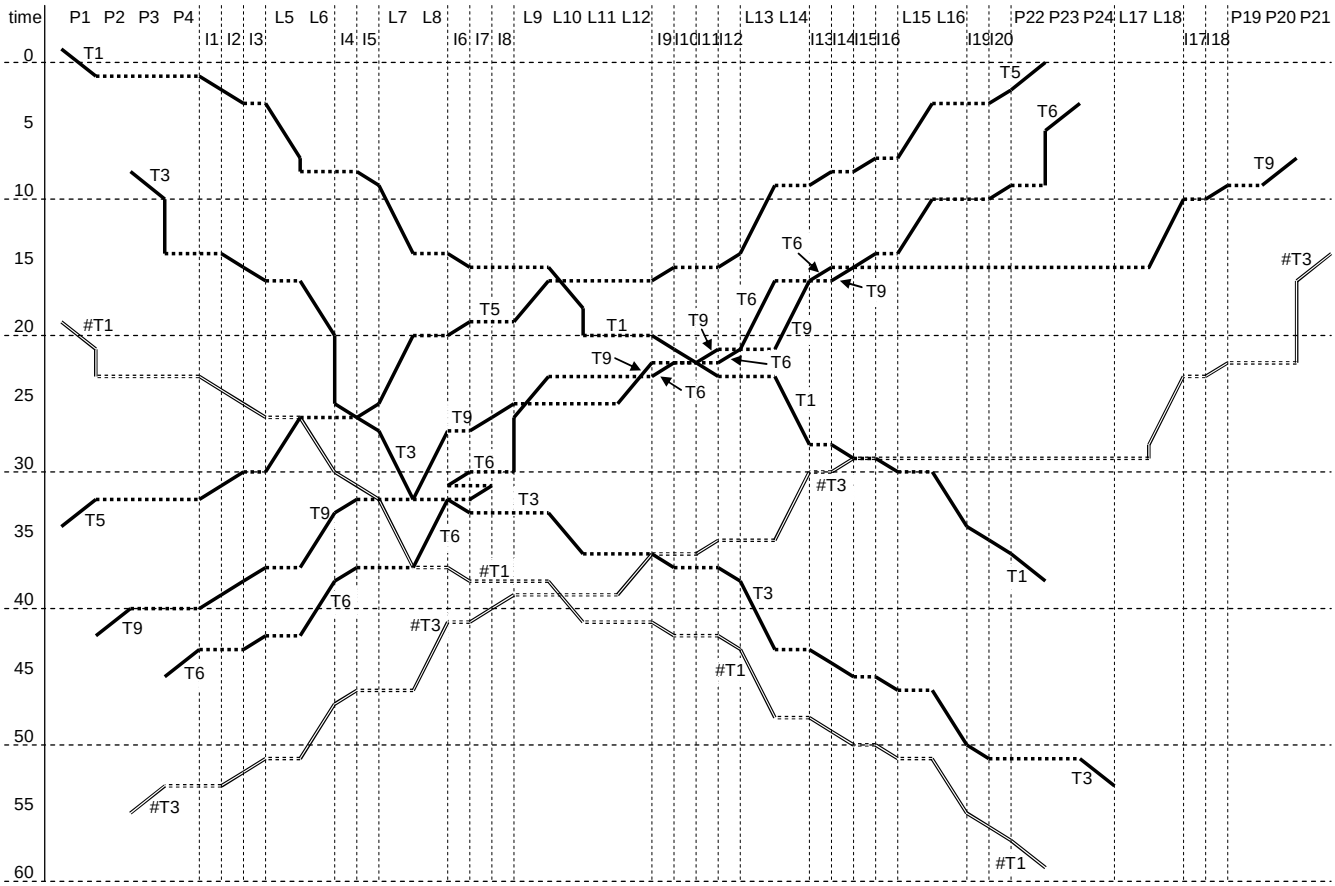


Figure 6. The solution (scheduled trains) found by the genetic algorithm.

The computational process required 68 generations and took 53 minutes on an Intel Dual Core processor to complete. The chromosome had 200 genes.

The second test carries out online scheduling: it considers the circulating trains set T_C = {T1, T3, T5, T6, T9} already scheduled, and uses genetic algorithms to schedule the new trains set T_N = {#T1, #T3}. It was observed that the new trains are scheduled according to their deadlines. The schedules are shown in Figure 6 as well. New trains are represented by doubled lines. The total train traveling time is 81 time units, out of which 9 are delays. One can observe there are more delays per train in this second case, when some of the resources are already occupied.

The computational process required 37 generations and took 33 minutes to complete. The chromosome had 80 genes.

Depending on the train table, in some cases, new trains may not be scheduled to meet their deadlines. If the necessary resources are already occupied, the problem may not have a feasible solution.

VII. CONCLUSIONS

The use of the heuristic approach of the scheduling problem is justified by the search space dimension. Due to the short durations necessary to execute the genetic algorithms, these can be used for on-line train scheduling. The execution durations are significantly dependent on the genotype length. Taking into account that the genotype contains a gene for each train and each interlocking, the gene number is $m \cdot n$ (where m is the number of the new trains that have to be scheduled and n is the interlocking number). The number of trains that are already scheduled can slightly influence the execution durations. The genotype and the railway model were chosen such that to diminish the search space of the solution and to simplify the implementation.

The problem of train scheduling to fulfill a given timetable is very similar to real-time task scheduling on a processor. The main parallel lines in Figure 1 can be considered 2 processors and each train is similar to an execution thread or task. Each task has a starting time and a deadline by which it should finish. Task scheduling and processor allocation can therefore be done by genetic algorithms.

REFERENCES

[1] Juan, E. Y. T., Tsai, J. J. P., Murata, T. and Zhou, Y. (2001). *Reduction methods for real-time systems using delay time Petri nets. IEEE Trans. on Soft. Eng., Vol. 27, No.5*, pp 422-448.

[2] Leţia, T., Hulea, M., Miron, R. (2008). *Distributed Scheduling for Real-Time Railway Traffic Control*. IMCSIT 2008, Real-Time Workshop RTS'08, Wisla, ISSN 1896-7094, Polonia, 2008. pp. 679-685.

[3] Shoji, S., Igarashi, A., (1997). *New trends of train control and management systems with real-time and non-real-time properties*, Proceedings of the 3rd International Symposium on Autonomous Decentralized Systems (ISADS'97), pp.319-326.

[4] Pachl, J., (2004). *Railway Operation and Control*, VTD Rail Publishing, Mountlake Terrace WA 98043 USA.

[5] Tornquist, J., and Persson, J.A., (2005). *Train traffic deviation handling using tabu search and simulated annealing,* Proceeding of the 38th Annual Hawaii International Conference on System Science, pp. 73a.

[6] Hauxthausen, A. and Peleska, J., (2000). *Formal development and verification of a distributed railway control system*, IEEE Trans. on Software Engineering, volume:26.

[7] Fahmy, S.F., Ravindran, B., and Jensen, E.D., (2008a). *On collaborative scheduling of distributable real-time threads in dynamic*, Networked Embedded Systems, Proceedings of the 11th IEEE Symposium on Object Oriented Real-Time Distributed Computing (ISORC), pp. 485-491.

[8] Fahmy, S.F., Ravindran, B., and Jensen, E.D., (2008b). *Scheduling distributable real-time threads in the presence of crash failures and message losses*, ACM SAC, Track on Real-Time Systems.

[9] Zimmermann, A. and Hommel, G., (2003). *A train control system case study in model-based real-time system design*, Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS'03), Nice, France, pp. 118b.

[10] Fukuta, Y., Kogure, G., Kunifuji, T., Sugahara, H., Ishima, R. and Matsumoto, M., (2007). *Novel railway signal control system based on the internet technology and its distributed control architecture*, Proceedings of the Eighth International Symposium on Autonomous Decentralized Systems (ISADS'07).

[11] Goverde, R.M.P., (2006). *A delay propagation algorithm for large-scale scheduled rail traffic*, Preprints of 11th IFAC Symposium on Control in Transportation Systems, Delft, Netherlands, pp. 169-175.

[12] Martí, P., Lin, C., Brandt, S.A., Velasco, M. and Fuertes, J.M.(2004), *Optimal State Feedback Based Resource Allocation for Resource-Constrained Control Tasks*, The 25th IEEE International Real-Time Systems Symposium (RTSS'04), pp.161-172.

[13] Martí, P., Fohler, G., Ramamritham, K. and Fuertes, J.M., (2002). *Improving quality-of-control using flexible time constraints: Metric and scheduling issues*. In Proceedings of the 23nd IEEE Real-Time Systems Symposium (RTSS 2002).

[14] Yépez, J., Fuertes, J. and Marti, P., (2003). *The large error first (LEF) scheduling policy for real-time control systems*. In Work in Progress Proceedings of the 24th IEEE Real-Time Systems Symposium (RTSS WIP 2003), pp. 63-66.

[15] Velasco, M., Fuertes,J. and Marti, P., (2003). *The self triggered task model for real-time control systems*. In Work in Progress Proceedings of the 24th IEEE Real-Time Systems Symposium (RTSS WIP 2003), pp. 67-70.

[16] Lin, C., Marti, P., Brandt, S.A., Banachowski, S., Velasco, M. and Fuertes, J.M., (2004). *Improving control performance using adaptive quality of service in a real-time system*. In Work in Progress Session of the IEEE Real-Time and Embedded Technology and Applications Symposium (UC Santa Cruz Technical Report UCSC-CRL-04-04), Toronto, Canada.

[17] Tormos, P., Lova, A., Barber, F., Ingolotti, L., Abril, M. and Salido, M.A., (2008), *A Genetic Algorithm for Railway Scheduling Problems*, Studies in Computational Intelligence(SCI)128, pp.255–276, www.springerlink.com, (Springer-Verlag Berlin Heidelberg).

[18] Serafini, P. and Ukovich, W., (1989). *A mathematical model for periodic scheduling problems*. SIAM J. on Discrete Mathematics, 2, pp.550–581.

[19] Kroon, L. and Peeters, L., (2003). *A variable time model for cycling railway timetabling*. Transportation Science, 37(2), pp.198–212.

[20] Liebchen, C., (2006). *Periodic Timetable Optimization in Public Transport*. dissertation. de - Verlag im Internet GmbH.

[21] Silva de Oliveira, E., (2001). *Solving Single-Track Railway Scheduling Problem Using Constraint Programming*. PhD thesis, The University of Leeds, School of Computing.

[22] Kwan, R.K.S. and Mistry, P., (2003). *A co-evolutionary algorithm for train timetabling*. In IEEE Press, editor, Congress on Evolutionary Computation, pp. 2142–2148.

[23] Caprara, A., Monaci, M., Toth, P. and Guida, P., (2006). *A lagrangian heuristic algorithm for a real-world train timetabling problem*. Discrete Applied Mathematics, 154, pp.738–753.

[24] Ingolotti, L., Lova, A., Barber, F., Tormos, P., Salido, M.A. and Abril, M., (2006). *New heuristics to solve the csop railway timetabling problem*. Advances in Applied Artificial Intelligence. LNAI, Subseries of Lecture Notes in Computer Science.

Complex negotiations in multi-agent systems

Mihnea Scafeş, Costin Bădică
University of Craiova, Software Engineering Department
Bvd.Decebal 107, Craiova, 200440, Romania
{scafes_mihnea,badica_costin}@software.ucv.ro

***Abstract*— This paper presents an overview of existing models of automated negotiation in multi-agent systems and then focuses on more complex negotiations involving non-linear utility functions.**

I. Introduction

Negotiation is used in business (agree over the price of an item), politics (negotiation between countries over some regional resources they want to use) and various other domains. As the information systems became more and more advanced, negotiation started to be used between such systems, using computers. Over the last years researchers tried to automate the negotiation process and they used computer science for implementation and analysis of negotiation algorithms. They spent considerable effort trying to find better negotiation models that lead to better outcomes. Automated negotiation is often studied in the field of multi-agent systems (MAS), a research field in which researchers combine techniques from distributed systems and artificial intelligence.

Generally, negotiation brings together three topics [1]: negotiation protocols, negotiation subject and negotiation strategies. The protocols are the rules that negotiation participants must obey when negotiating. It describes the steps of a negotiation, what messages can be sent, what actions participants are allowed to take during each phase of the negotiation. There are various types of protocols, for different types of negotiation: Contract Net Protocol (CNP) [2] for distributed problem solving, Rubinstein's alternating offers protocol [3] for bargaining problems, Monotonic Concession Protocol (MCP) [4], auctions.

The negotiation subject or negotiation object, as it is also called, describes what is being negotiated between the partners. It can be a state of the environment the participants what to reach, an action they would like to perform or an item they would like to have. If the subject is composed of multiple attributes it is called *multi-issue*, otherwise it is called *single-issue*. For example, when a car dealer and a client are negotiating about a car, they might negotiate about the price of the car, the engine and other options the client might want, which means they are negotiating about a multi-issue subject. Further more, the issues are *divisible* if they can be split between the participants (i.e. if all the participants can get a share of the issue, e.g. money) and *indivisible* if they cannot (e.g. a house).

The strategy of the agents is strongly dependent on the problem domain, the protocol, the subject and the information the agents have. The strategy represents how agents make decisions during negotiation.

Typically, there are two types of negotiation problems: bargaining and task allocation [5].

The bargaining problem has been studied in game theory and can be applied to automated negotiation between self-interested agents. In bargaining problems, each agent tries to maximize its own preference measure. *Utility functions* are generally used to represent preferences. Agents use a certain interaction protocol when exchanging proposals and a strategy when computing proposals.

In a task allocation problem, agents are able to perform tasks with an associated cost and sometimes want to delegate tasks to other agents. An agent might choose to delegate a task either because it might not be able to perform it or because it might cost less than performing it itself.

The Contract Net Protocol (CNP) [2] is usually studied for task contracting.

This paper presents an overview of the state-of-the-art in negotiations with a focus on complex negotiations and their challenges. The paper is structured as follows. Section II consists of related work in negotiation models with linear utility functions. This section provides a background in negotiation, emphasizes achieved results and problems that researchers have encountered. Section III provides guidelines for developing negotiation models. These guidelines result from the related work and consist of the key elements researchers should take into account when developing negotiation models. Section IV discusses negotiations with complex utility functions. Section V contains conclusions and directions for future work.

II. Existing negotiation models

For the rest of this section we will discuss the work of various researchers in the field of automated negotiation, and emphasize main ideas of their research.

[1] discusses agent organization (cooperation and coordination) by means of automated negotiation. The authors discuss existing methods at the time and emphasize challenges encountered by researchers. They present the three

Mihnea Scafeş was supported by IOSUD-AMPOSDRU contract 109/25.09.2008. Costin Bădică was supported by DIADEM project. Diadem project is funded by European Union under Information and Communication Technologies (ICT) theme of the 7th Framework Programme for R&D, ref. no: 224318.

main components of negotiation: protocols, objects and decision making models (strategies). During negotiation, agents search in their private deal spaces (the space of potential outcomes) for offers to make to the other agents. Thus, negotiation is presented as a distributed search problem. The deal spaces might change as the negotiation progresses, as a result of context changes or persuasion. Agents can critique the received offers and can make counter-offers. Various negotiation techniques bring particular elements to this framework. Game-theoretic techniques can help design negotiation protocols and strategies and provide strong theoretical methods for the analysis of negotiations. Using such techniques, a negotiation is modeled as a strategic game. But they assume complete knowledge of information, an assumption which rarely holds in real-world situations. Moreover, searching for solutions is often an intractable problem (e.g. finding equilibria). The authors claim that game-theoretic techniques are much harder to use for multi-issue negotiations. Heuristic approaches attempt to overcome the disadvantages of game-theoretic approaches. Mainly, they tend to find acceptable solutions, rather than optimal solutions. Apart from the non-optimal solutions that they often provide, another disadvantage of heuristic approaches is that they often need experimental evaluation. Argumentation-based approaches offer the possibility for agents to exchange more information than just proposals or counter-proposals – they can offer some details about their decisions (why has an agent rejected a proposal).

In a related paper, Kraus discusses automated negotiation for various domains [6]. The concept of *equilibrium* (used in game-theory) is explained in the context of strategic negotiation. The author also discusses other negotiation methods such as auctions, task allocation, coalition formation and argumentation.

A. Task allocation

The Contract Net Protocol (CNP) [2] is a well known mechanism for contracting. An agent called *manager* makes tasks available to *contractor* agents. After a selection process, the manager awards one or more tasks to a contractor, supervises the execution of tasks and processes task results. The contractor is in charge of task execution. The original specification of the CNP does not contain any strategy model. The decision making models for bidding, bid processing and task awarding phase are not specified. Sandholm proposes a study of the CNP based on marginal cost calculations [7]. Agents use local utility functions for computing costs that are taken into account during the bidding and awarding phase. This model formalizes the decision steps of CNP, but the local decision models (the utility functions for example) are not specified. This type of negotiation has been applied to a vehicle routing problem, where dispatch centers negotiate in order to route vehicles efficiently and reduce costs.

Because in the CNP the manager tries to optimally select contractors, it would be nice to have some information about the performance and efficiency about contractors prior to negotiations. [8] uses this approach in the context of task allocation among robots. They record successes and failures of contractors and use this information to select potential contractors. The manager uses credibility and relevance in order to select the contractor after the bidding phase. They also discuss commitment and permit breach of contract in their negotiations. A contractor is allowed to breach a contract by paying a penalty to the manager. The experiments prove that the negotiation mechanism is feasible.

Researchers have tried to extend the CNP according to their needs, for various negotiation problems. A survey of extensions to the CNP, as well as a theoretical evaluation of these extensions can be found in [9].

A scalability analysis of the CNP is done in [10] experimentally, the authors concluding that the performance depends on the system load and that the CNP cannot be used without a deadline.

The CNP has been studied for task allocations in the context of an insurance application [11], where the authors characterize the negotiation issues with several properties (preferred value, reserved value, utility and weight) and implemented various strategies for generating proposals and calls for proposals.

An experimental comparison of three different methods for task allocation (sequential auctions, multi-issue MCP and mediator-based simulated annealing) has been done in [12], for the scope of allocating tasks for monitoring the environment using negotiations between ground control stations and orbiting space probes.

B. Bargaining

One of the first models of bargaining is Rubinstein's alternating offers game with an infinite horizon (no deadlines) and complete information [13]. In this game, two players are bargaining on how to split a pie. Each player proposes a partition of a pie at each turn. The other player must either accept the offer or reject it and propose another partition. The game takes the time preferences into account as fixed bargaining costs or fixed discounting factors. Rubinstein later extended this model [14] for incomplete information. He introduces an uncertainty element in his previous model, namely one of the two agents can be of two types: *strong* or *weak*. The *weak* player is more "impatient", i.e. it loses utility more rapidly with time than the *strong* player. The scenario presented in both models of Rubinstein is that of two players trying to split a pie, that is, negotiation about a single issue.

Another interesting model of bilateral single-issue negotiation is that of [15]. The authors prove that in incomplete information settings with time effects (deadlines and discount factors), the bargainers prefer to wait until the earlier deadline. That is, agreement is reached at the latest possible step.

Multi-issue negotiations are more interesting to study mainly because different procedures can be used to negotiate the issues. They can be negotiated all together (*bundled*, in package) or sequentially (*issue-by-issue*). It is also more difficult to design negotiation models with multiple issues using game-theoretic techniques [1]. Using the *issue-by-issue*

approach, it has been shown that the order in which the issues are negotiated (the agenda) influences the negotiation outcome [16]. When the decision of what issue to negotiate next is taken during negotiation, the agenda is *endogenous*, otherwise it is *exogenous*.

An agenda-based approach to negotiations has been proposed in [17]. It is a multi-issue, issue-by-issue negotiation model with incomplete information. Agents (two agents, a buyer and a seller) use utility functions with time discounts to evaluate received proposals. Depending on the discount factors, they can be *patient* (gain utility with time) or *impatient* (lose utility with time). Three different types of tactics are used to generate counter-offers [18]: (i) *boulware* – the agent concedes very slowly during the negotiation, but near the deadline it concedes rapidly to the reservation price; (ii) *conceder* – the agent concedes rapidly at the beginning of the negotiation to the reservation value; (iii) *linear* – the agent concedes linearly. Agents have incomplete information about their opponents in the form of two probability distributions over many values, one for the opponent's deadline and the other for the opponent's reservation price. The relationships between agent deadlines and discount factors lead to six negotiation scenarios that are then analyzed. For each one of the six scenarios the authors determine the optimal strategy, i.e. the strategy that gives the maximum expected utility. The authors also determine conditions for convergence of the optimal strategies, study the properties of the equilibrium and determine conditions under which the equilibrium is Pareto optimal. This model is extended to multiple issues by extending the information model of the agents. Te analysis of negotiation with multiple issues includes a comparison of two implementations: exchange of an issue takes place immediately after the issue has been negotiated – *sequential implementation*, or after all the issues have been negotiated – *simultaneous implementation*. The negotiation agenda is endogenous, the order in which the issues are negotiated is determined by the equilibrium.

A heuristic approach is presented in [19], [20]. The authors have created a *System for Analysis of Multi-Issue Negotiation (SAMIN)* for the purpose of testing and improving negotiations among humans. Negotiation processes are formalized using a *temporal trace language (TTL)* which makes it easy to add useful properties that the system should be tested against (e.g. Pareto monotony). Their model also uses incomplete information: one party can estimate the other's issue weights. This guessing is based on the assumption that an agent concedes more on a less important issue. Time effects are not present in the model. Also, the model is not theoretically analyzed.

The authors of [21] present a negotiation model with applications to an international crisis. They take time into account, describe the equilibrium and show that, in theory, if an agreement can be reached, it will be reached in the first or the second time step of negotiation.

In [22], the authors study bilateral multi-issue negotiation with indivisible issues, incomplete information and time constraints in the form of deadlines and discount factors. For indivisible issues, finding the equilibrium is an NP-hard problem (similar to the 0-1 knapsack problem). Thus, the authors find approximate strategies and show that they form equilibria (approximate). The study is extended to online negotiation, when issues become available later in the negotiation and the agents are unsure about their preferences. This situation also leads to an approximate equilibrium.

In [23], [24], the same authors study the negotiation procedure. They compare the *package deal* (bundled issues, negotiated together), *simultaneous* (negotiated at the same time, but independent of each other) and *sequential* (independently and one after another) procedures in bilateral, multi-issue negotiations with incomplete information. The difference between the two papers is that they focus on different types of uncertainties, but the key result of both is that the optimal procedure is package deal.

The same researchers also study optimal agendas. In [25] they study the optimal agenda for bilateral, 2-issue negotiations with incomplete information. Four negotiation scenarios are defined, depending on the zone of agreement for the two issues. Then they identify the optimal agenda for each one of the two procedures. In [26], the authors decompose the negotiation issues into a number of stages. At each stage, the issues to be negotiated are determined exogenously and the order is determined endogenously. Therefore, the authors use study an agenda that is both exogenous and endogenous. They determine negotiation scenarios that give agents higher utilities when the issues are negotiated in stages, but the agents are not able to identify the scenarios, since they do not have complete information. A mediator is used to help the agents identify those scenarios.

III. Developing negotiation models

There are lots of things to be taken into consideration when designing negotiation models. Most often the protocols, strategies and negotiation objects are domain dependent. That means that the negotiation designer first analyzes the problem to be resolved (the environment, the available resources) and then chooses the most appropriate negotiation protocol, strategies and subject.

When there are only two agents that will interact, a model for bilateral negotiation will be used, like [17], [15]. When more that two agents will interact, multilateral negotiation models will be used. Multilateral negotiation models can be divided into two categories: (i) one-to-many negotiations in which one of the agents has a special role (e.g. *coordinator*, *beneficiary* in service contracting) and (ii) many-to-many negotiations, where all the agents have the same privileges and negotiate for the partitions of an object. One-to-many negotiations are typically modeled using auctions.

The negotiation subject is one of the most important things to take into consideration, as it is the element that drives the negotiation (e.g. a pie that will be split [13], a house that will be sold) and gives the outcome structure. If there

is only one atomic item/issue[1] that will be negotiated, the subject is said to be *single-issue*. If the negotiation subject can be split into multiple atomic items/issues, the subject is said to be *multi-issue*. Single-issue negotiations are easier to study using game-theoretic techniques [1] because the preferences over a single issue can be represented easier. Humans usually prefer one issue to another, and one value of an issue to another. In economics, these preferences are represented using *indifference curves* [27], but in this form they cannot be compared. Therefore, the concept of utility [27] has been introduced. Utility functions are largely used to represent and compare preferences in multi-agent negotiation. In negotiation, a utility function $u(x) : \mathcal{X} \rightarrow \mathcal{R}$ maps a set of outcomes to real numbers. It shows how much an agent prefers an outcome using a real number. Usually this function is bounded to a positive interval of the real numbers, e.g. $[0, 1]$. A utility of 0 denotes no valuation. Usually 0 utility is assigned also to the conflict deal (i.e. no deal), this having the effect that the agent does not want to reach a conflict but tries to reach an agreement and gain an outcome. In multi-agent negotiations, agents use utility to represent their preferences and evaluate outcomes and try to maximize their utilities [28] (Von Neumann-Morgenstern utility maximizers). If the agents value the issues independently (issues do not depend on one another), a weighted-sum utility function can be used to evaluate outcomes [29].

If the agent considers dependencies between agents, the utility function has more complex forms, depending on how these interdependencies are modeled [30], [31], [32].

The interaction protocols define the conditions under which agents exchange messages (offers, counter-offers). For bargaining problems the most encountered protocol is the alternating offers [3] protocol and variations. Using this protocol, agents start by offering deals that give maximum utility and then continue making concessions to the other agent using different tactics (for an example of tactics, including time-dependent, see [18]) until an agreement or the conflict deal is reached. A typical flow of this protocol can be seen in Figure 1, where two agents *a* and *b* reach an agreement after their utilities dropped from maxima to the scoring of the agreement outcome. After receiving an offer, an agent evaluates the offer using its utility function and decides whether to accept the offer or make a counter-offer. The process of evaluating the offers and making concessions are influenced by the preferences the agents have about the negotiation subject, the environment (e.g. time) and the other agents. For example, time can influence an agent's utility function and how it makes proposals. The agent can evaluate more the negotiation subject as time passes (gains utility with time), or the negotiation subject might loose importance as time passes (looses utility with time). These time constraints are usually modeled using *deadlines* and/or *discount factors* [13], [14], [16], [17], [15], [18], [30]. Bargaining problems (taking the model from game-theory) have been divided

[1]Please note that atomic here means that the negotiator does not have any interest to divide the issue more, i.e. he does not have different preferences for smaller parts of the issue

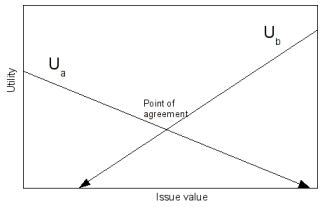


Fig. 1. Bargaining with linear functions

into: negotiation with complete information and negotiation with incomplete information [28]. Information completeness means agents completely know their and the other agents' preferences. Information incompleteness refers to the lack of knowledge about some information, either about themselves or about the other agents and it is the most encountered situation in real world. However, agents can still try to model these parameters, often by using probability distributions (see for example the model of [17]). An interesting fact has been pointed out by [15]: an agent reveals information about itself when making offers or counter-offers. Negotiation with incomplete preferences is harder as compared to negotiation with complete information. With complete information means that at least theoretically, if the point of agreement exists then it can be computed (even if sometimes this computation is too expensive). Note than knowing if a solution exists is a different thing from knowing the actual solution.

Having a mechanism is not sufficient. It must be stable, i.e. the agents must not have the incentive to deviate from their strategies. Game-theory provides the concept of equilibria [3]. Various types of equilibria are available, depending on the situation. Sometimes equilibria might be hard to compute, even though it can be proven they exist. If there are equilibria, the optimal one (i.e. the one that gives agents maximum utility) is desirable. But in some cases the existence of equilibria is not easy to prove and the performance of the negotiation model is evaluated through experiments.

IV. Advanced negotiations

Bargainers typically start by proposing the deal that gives them maximum utility and then make concessions following a certain strategy. They seek to maximize their utility functions. Figure 1 shows a simplified process of this kind. This hill-climbing approach works perfectly as long as the utility functions are monotonic. As monotonic functions have only one local optimum (which is also the global optimum), the hill-climbing method stops at the global optimum. But this situation changes when the utility functions are not monotonic, as the hill-climbing method is not able to get past local optima.

For example, the non-monotonic utility functions of two agents, *a* and *b*, are represented in Figure 2[2]. It can be observed that the utility function of agent *a* has 3 local

[2]The functions are depending on only one issue for ease of representation, as multiple issues require multiple dimensions

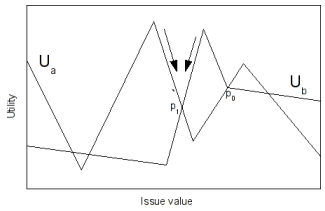


Fig. 2. Bargaining with non-monotonic functions

maxima (one of which is the global maximum). The utility function of agent *b* has two local maxima.

When they are brought together in a bargaining process (Figure 2), the agents use the hill-climbing method and reach an agreement point, p_1, but the outcome is not optimal. Instead, they should have reached the agreement in p_0 in order to get the maximum outcome.

This particular type of problem arises when agents negotiate about interdependent issues. The utility functions become non-monotonic in complex situations. In order to use hill-climbing, one must design methods for detection of local optima and take actions accordingly. One possibility to do that is to explore the entire deal space and mark the points that give local optima. But if the number of issues is very large, the deal space becomes too complex to be easily explored. Researchers have tried to overcome this problem. All the known proposed solutions are heuristic. Many of them make use of the simulated annealing optimization algorithm [29]. This algorithm, might accept, with a certain probability, solutions that might not be better than the current solution. It has been shown in practice that simulated annealing can get past local optima and reach global optimum in many situations, unlike hill-climbing, which gets stuck in local optima.

The problem of complex negotiations is described in [32]. The authors develop mediated bilateral negotiations about interdependent boolean issues. The preference model is simple, but it defines a very large space that cannot be easily explored. They study the outcome of negotiations between two types of agents: hill-climbing (accepting only contracts better than the last accepted contracts) and annealing (can accept worse contracts with a certain probability). When pairing hill-climbers only, the outcome is poor (they get stuck in local optima), while when pairing annealers, the outcome is good (as they can get past local optima). However, when pairing one hill-climber with one annealer, the hill-climber does very well because the annealer, mostly at the beginning, accepts even non-beneficial contracts. By improving their model, they come up with a final solution in which they put the annealing part inside the mediator and extend the negotiation protocol to allow agents to vote the contracts proposed by the mediator. The solution works very well and the work is very valuable.

[31] studies a model of multi-issue negotiation with non-linear and non-monotonic utilities and interdependent issues. The interdependencies between the issues are represented with constraints. As it is very hard to completely explore the deal space in this situation, the agents first take random samples from the deal space, then adjust the samples to find local optima using simulated annealing and then make bids. The model limits the number of bids the agent can make to make computations finish in a reasonable amount of time. A mediator computes bid intersection and determines the outcome. The performance drops exponentially with the number of issues. A comparison with a hill-climbing method is provided, showing that the proposed protocol performs better, especially in complex deal spaces with local optima, where the hill-climbing method blocks. They somehow improve their model in [33] by using a three-stage protocol which reduces complexity, but the disadvantages of the previous model, namely the presence of a mediator, a bid limit per agent and comparison with hill-climbing only, still remain. The presence of a mediator is disadvantageous because agents have to give their private preferences to a third-party agent. The model has not been analyzed using game-theoretic tools.

Another negotiation model with simulated annealing is [34]. This work does not involve a mediator, but the authors do not compare their approach against the hill-climbing method.

There are, however, other methods that do not use simulated annealing and focus on the approximation of the utilities, or try to reduce the complexity of the problem.

Results of finding the optimal procedure in case of nonlinear utility functions are shown in [35]. Computing equilibrium for the package deal procedure if utility functions are nonlinear is hard. The authors compare the package deal procedure (when issues are bundled together) for a linear approximation of the utility functions with the simultaneous procedure (when issues are negotiated independently of each other, in parallel), resulting that an equilibrium can be computed in polynomial time for both procedures and although the package deal procedure leads to Pareto optimality, the simultaneous procedure can give better outcomes to one of the agents.

Another model that works with interdependent issues, [30], considers more complex interdependencies with the help of the Choquet integral, which has a higher representation power than the utilities in [31]. They study a cooperative protocol and show that the protocol has subgame perfect equilibria.

By modeling agent preferences using utility graphs and trying to decompose them, the complexity of the problem can be exponentially reduced (although it remains exponential) [36]. The outcomes are close to maximum efficiency.

V. Conclusions and future work

Negotiation with non-linear utility functions is a complex problem. Such situations are very common in the real world. Little research has been done in this area, the typical methods that have been used are the presence of a mediator and simulated annealing. More research should be done to find other approaches to this problem, better methods to explore

these complex deal spaces and reach agreements in a finite amount of time. Designing such methods would imply a complete design of the agent preferences. They might be extended with methods for representing and learning the opponent agents' preferences (mainly the interdependencies between issues) and finding optimal strategies. If the solutions cannot be theoretically verified various experiments will be performed to measure their performance, efficiency and other characteristics.

REFERENCES

[1] N. R. Jennings, P. Faratin, A. R. Lomuscio, S. Parsons, C. Sierra, and M. Wooldridge, "Automated negotiation: Prospects, methods and challenges," *International Journal of Group Decision and Negotiation*, vol. 10, no. 2, pp. 199–215, 2001. [Online]. Available: http://eprints.ecs.soton.ac.uk/4231/

[2] R. G. Smith, "The contract net protocol: High-level communication and control in a distributed problem solver," *IEEE Trans. Comput.*, vol. 29, no. 12, pp. 1104–1113, 1980.

[3] M. Osborne and A. Rubinstein, *A Course in Game Theory*. Cambridge, MA: MIT Press, 1994.

[4] J. S. Rosenschein and G. Zlotkin, *Rules of Encounter: Designing Conventions for Automated Negotiation Among Computers*. Cambridge, Massachusetts: MIT Press, 1994.

[5] J. M. Vidal, *Fundamentals of Multiagent Systems: Using NetLogo Models*. Unpublished, 2006. [Online]. Available: http://www.multiagent.com/fmas

[6] S. Kraus, "Automated negotiation and decision making in multiagent environments," *Mutli-agents systems and applications*, pp. 150–172, 2001.

[7] T. W. Sandholm, "An implementation of the contract net protocol based on marginal cost calculations," in *Proceedings of the 12th International Workshop on Distributed Artificial Intelligence*, Hidden Valley, Pennsylvania, 1993, pp. 295–308. [Online]. Available: citeseer.ist.psu.edu/sandholm93implementation.html

[8] H. Yu-Sheng, X. Mo-Gen, and Z. Pu-Cheng, "Extended contract net protocol for multi-robot dynamic task allocation," *Information Technology Journal*, vol. 6, no. 5, pp. 733–738, 2007.

[9] E. Bozdag, "A survey of extensions to the contract net protocol," CiteSeerX - Scientific Literature Digital Library and Search Engine [http://citeseerx.ist.psu.edu/oai2] (United States), Tech. Rep., 2008. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary10.1.1.100.567

[10] Z. Juhasz and P. Paul, "Scalability analysis of the contract net protocol," in *CCGRID '02: Proceedings of the 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid*. Washington, DC, USA: IEEE Computer Society, 2002, p. 346.

[11] S. Paurobally, V. Tamma, and M. Wooldrdige, "A framework for web service negotiation," *ACM Trans. Auton. Adapt. Syst.*, vol. 2, no. 4, p. 14, 2007.

[12] D. Chakraborty, S. Saha, S. Sen, and B. Clement, "Negotiating monitoring task allocation for orbiters," in *ICDCN*, ser. Lecture Notes in Computer Science, S. Chaudhuri, S. R. Das, H. S. Paul, and S. Tirthapura, Eds., vol. 4308. Springer, 2006, pp. 282–287.

[13] A. Rubinstein, "Perfect equilibrium in a bargaining model," *Econometrica*, vol. 50, no. 1, pp. 97–109, January 1982. [Online]. Available: http://ideas.repec.org/a/ecm/emetrp/v50y1982i1p97-109.html

[14] ——, "A bargaining model with incomplete information about time preferences," *Econometrica*, vol. 53, no. 5, pp. 1151–72, 1985. [Online]. Available: http://econpapers.repec.org/RePEc:ecm:emetrp:v:53:y:1985:i:5:p:1151-72

[15] T. Sandholm and N. Vulkan, "Bargaining with deadlines," in *AAAI '99/IAAI '99: Proceedings of the sixteenth national conference on Artificial intelligence and the eleventh Innovative applications of artificial intelligence conference innovative applications of artificial intelligence*. Menlo Park, CA, USA: American Association for Artificial Intelligence, 1999, pp. 44–51.

[16] C. Fershtman, "The importance of the agenda in bargaining," *Games and Economic Behavior*, vol. 2, no. 3, pp. 224–238, September 1990. [Online]. Available: http://ideas.repec.org/a/eee/gamebe/v2y1990i3p224-238.html

[17] S. S. Fatima, M. Wooldridge, and N. R. Jennings, "An agenda-based framework for multi-issue negotiation," *Artif. Intell.*, vol. 152, no. 1, pp. 1–45, 2004.

[18] P. Faratin, C. Sierra, and N. R. Jennings, "Negotiation decision functions for autonomous agents," *Int. Journal of Robotics and Autonomous Systems*, vol. 24, no. 3 - 4, pp. 159–182, 1998. [Online]. Available: http://eprints.ecs.soton.ac.uk/2117/

[19] T. Bosse, C. M. Jonker, L. Meij, V. Robu, and J. Treur, "A system for analysis of multiissue negotiation," in *Software Agent-Based Applications, Platforms and Development Kits*. Birkhaeuser Publishing Company, 2005, pp. 253–280.

[20] T. Bosse, C. M. Jonker, and J. Treur, "Experiments in human multi-issue negotiation: Analysis and support," *Autonomous Agents and Multiagent Systems, International Joint Conference on*, vol. 2, pp. 671–678, 2004.

[21] S. Kraus and J. Wilkenfeld, "A strategic negotiations model with applications to an international crisis," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 23, 1993.

[22] S. Fatima, M. Wooldridge, and N. R. Jennings, "Approximate and online multi-issue negotiation." in *6th International Joint Conference on Autonomous Agents and Multi-agent Systems.*, 2007, pp. 947–954. [Online]. Available: http://eprints.ecs.soton.ac.uk/14219/

[23] ——, "Multi-issue negotiation with deadlines." *Journal of Artificial Intelligence Research*, vol. 27, pp. 381–417, 2006. [Online]. Available: http://eprints.ecs.soton.ac.uk/13079/

[24] ——, "On efficient procedures for multi-issue negotiation," in *8th Int. Workshop on Agent-Mediated Electronic Commerce*, 2006, pp. 71–84. [Online]. Available: http://eprints.ecs.soton.ac.uk/12582/

[25] ——, "Optimal negotiation of multiple issues in incomplete information settings," in *AAMAS '04: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*. Washington, DC, USA: IEEE Computer Society, 2004, pp. 1080–1087.

[26] ——, "Optimal agendas for multi-issue negotiation," in *2nd International Joint Conference on Autonomous Agents and Multi-Agent Systems*, 2003, pp. 129–136. [Online]. Available: http://eprints.ecs.soton.ac.uk/8618/

[27] H. R. Varian, *Intermediate Microeconomics: A Modern Approach*, 7th ed. W. W. Norton & Company, 2005.

[28] J. von Neumann and O. Morgenstern, *Theory of Games and Economic Behavior*. Princeton, NJ: Princeton University Press, 1947.

[29] S. J. Russell and P. Norvig, *Artificial Intelligence - A Modern Approach*. Prentice Hall, 2003.

[30] M. Hemaissia, A. El Fallah Seghrouchni, C. Labreuche, and J. Mattioli, "A multilateral multi-issue negotiation protocol," in *AAMAS '07: Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*. New York, NY, USA: ACM, 2007, pp. 1–8.

[31] T. Ito, H. Hattori, and M. Klein, "Multi-issue negotiation protocol for agents: Exploring nonlinear utility spaces," in *IJCAI*, M. M. Veloso, Ed., 2007, pp. 1347–1352.

[32] M. Klein, P. Faratin, H. Sayama, and Y. Bar-Yam, "Negotiating complex contracts," *Group Decision and Negotiation*, vol. 12, pp. 111–125, March 2003. [Online]. Available: http://jmvidal.cse.sc.edu/library/klein03a.pdf

[33] H. Hattori, M. Klein, and T. Ito, "A multi-phase protocol for negotiation with interdependent issues," in *IAT '07: Proceedings of the 2007 IEEE/WIC/ACM International Conference on Intelligent Agent Technology*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 153–159.

[34] A. Kardan and H. Janzadeh, "A multi-issue negotiation mechanism with interdependent negotiation issues," in *ICDS '08: Proceedings of the Second International Conference on Digital Society*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 55–59.

[35] S. S. Fatima, M. Wooldridge, and N. Jennings, "An analysis of feasible solutions for multi-issue negotiation involving non-linear utility functions," in *Proc. 8th Int. Conf on Autonomous Agents and Multi-Agent Systems*, 2009, pp. 1041–1048. [Online]. Available: http://eprints.ecs.soton.ac.uk/17065/

[36] V. Robu, D. J. A. Somefun, and J. A. La Poutré, "Modeling complex multi-issue negotiations using utility graphs," in *AAMAS '05: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*. New York, NY, USA: ACM, 2005, pp. 280–287.

Improved power optimization method for squirrel-cage-induction-generator-based wind energy conversion systems

A. Scarlat, I. Munteanu, A.I. Bratcu, and E. Ceangă

Abstract—**The aim of this paper is to propose an improved power optimization control method of a squirrel-cage-induction-generator–based wind energy conversion system. It uses a vector control scheme and allows imposing both the slope and the zero-torque point of a generator forced mechanical characteristic. A better dynamic, along with extended domain of stable operation and controllable generator torque variations, is thus obtained. The closed-loop dynamics are analyzed through numerical simulation.**

I. Introduction

VARIABLE-SPEED wind energy conversion system (WECS) is a highly nonlinear time-variant system excited by stochastic inputs which significantly affect its reliability and lead to non negligible variations in the dynamic behaviour of the system over its operating range. This is a reason for which variable speed wind turbines control has not yet converged to a classical widely-accepted solution.

Usually the WECS control deals with a multicriteria objective, with various components, expressing energy and performance optimization problems, such as wind turbine power limitation in full-load operating mode, the alleviation of wind-turbulence-generated mechanical fatigue, maximization of captured energy in partial-load mode, the reduction of mean squared error between the actual operating point and the optimal power regime *etc.* [1]-[3]. The WECS control aiming at reducing the mechanical fatigue due to generator torque variations along with minimizing the mean squared error of the operating point around the optimal regime characteristic, ORC, is still a problem of significant interest. It can be stated as a dynamic optimization problem with the following performance criterion [4], [5]:

$$I = E\left[\alpha \cdot \left(\lambda - \lambda_{opt}\right)^2 + \Delta T_{em}^2\right], \tag{1}$$

where E is the statistical average, λ is the tip speed ratio defined as ratio between the peripheral speed of the blades and the wind speed, λ_{opt} is the optimal tip speed ratio that grants captured power maximization, ΔT_{em} is the electromagnetic torque variation and α is a weighting coefficient.

In this paper, the case of a high-power SCIG-based WECS, operating in the partial-load regime, is considered. The wind turbine is controlled to extract maximum power available in the wind. An improved control method is proposed that overcomes the main drawback of the classical control solution: slow dynamic response with accentuated overshoot [6]. The proposed method allows imposing both the slope and the zero-torque point of the SCIG mechanical characteristic. In this way good dynamic with acceptable electromagnetic torque variations is ensured. The presented results allow a comparison with the classical vector-controlled system.

This paper is structured as follows. Section II presents the statement of the improved control method. Section III presents the basic wind conversion system modelling; Section IV presents the vector control principle, the block defining the desired torque-versus-rotational-speed dependence within the proposed method, the closed-loop analysis and some design issues. Matlab®/Simulink® simulation results for the classical control and for the improved control are presented in Section V. Section VI presents the main conclusion regarding the obtained results.

II. Control Problem Statement

A large horizontal-axis wind turbine with a back-to-back converter and a vector-controlled SCIG has a slow dynamic. Its output energy efficiency is given by the first term of the performance criterion (1), whose minimization reflects good dynamic behavior. However, the two terms of the performance criterion have to be reasonably traded-off, to ensure system reliability also [4]. In the following, different possibilities of driving the SCIG generator at variable speed will be analyzed.

Using a scalar control, one can control generator speed in a wide range by varying the stator voltage and the frequency such as their ratio to be kept constant [7]. The generator mechanical characteristics in this case are presented in Fig. 1, case (a). The mechanical time constant of the system linearized around a typical operating point is [8]:

$$T = J_h \Big/ \left(\frac{\partial T_{em}}{\partial \Omega_h} - \frac{\partial T_w}{\partial \Omega_h}\right), \tag{2}$$

where T_{em} is the induction generator torque, T_w is the wind

A. Scarlat, I. Munteanu, A.I. Bratcu, and E. Ceangă are with Dunărea de Jos University of Galati, Faculty of Electrical and Electronics Engineering (email: adriana.scarlat@ugal.ro, iulian.munteanu@ugal.ro, antoneta.bratcu@ugal.ro, emil.ceanga@ugal.ro)

turbine torque, and J_h and Ω_h are the high-speed shaft turbine inertia and rotational speed.

Knowing that the induction generator characteristic slope, $\left(\partial T_{em}/\partial\Omega_h\right)$, is always larger than the wind turbine characteristic slope, $\left(\partial T_w/\partial\Omega_h\right)$, the time constant will always be positive and the system will always be stable. The major drawback of a scalar control is that the variations of the generator torque cannot be limited in absence of a torque control loop. Hence, the drive train mechanical loads cannot be conveniently controlled.

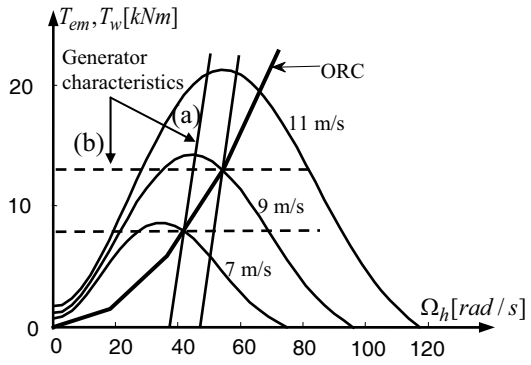


Fig. 1. Generator and wind turbine (high speed shaft) characteristics: a) for scalar-controlled system, b) for torque-controlled system

Using a torque vector control, the generator characteristics obtained for a constant torque set-point are presented in Fig.1, case (b). This intrinsically overcomes the drawback of the scalar control, but has other disadvantages. As the generator characteristics slopes are zero, the system is stable only on the descending side of the wind turbine characteristic. The stability may be obtained by employing an outer speed control loop. The other drawback is a large time constant (see (2)).

This paper proposes a new control method that combines the advantages of the two above-presented methods. Based on the torque vector control, the generator mechanical characteristic is forced to have a controllable slope.

III. WECS Modeling

The case of a back-to-back 2 MW rigid-drive-train SCIG-based WECS is considered in this paper. The average turbine (wind) torque is given by the expression:

$$T_w = 0.5\pi\rho \cdot v^2 R^3 C_T\left(\lambda\right), \tag{3}$$

where v is the wind speed, R is the blade length, ρ is the air density, and $C_T\left(\lambda\right)$ is the torque coefficient, which depends on the wind turbine tip speed ratio, λ. This dependence can be approximated by a polynomial and gives the wind torque shapes in Fig. 1. The tip speed ratio is given below:

$$\lambda = R\cdot\Omega_h \big/ \left(i\cdot v\right), \tag{4}$$

where i is the drive train multiplication ratio.

The rigid drive train, with the parameter values given in Appendix, is characterized by the following motion equation [9]:

$$J_h \frac{\mathrm{d}\Omega_h}{\mathrm{d}t} = \frac{T_w}{i}\left(\Omega_h, v\right) - T_{em}\left(\Omega_h, \Omega_c\right), \tag{5}$$

where $J_h = J_g + J_w \big/ i^2$ is the high-speed shaft inertia, J_g and J_w are the generator and turbine rotor inertias respectively.

The studied machine is a 2-pole-pairs SCIG. The generator feeds a power grid via a back-to-back converter [7]. The converter has two inverters decoupled by a DC-link. The grid-side inverter is dedicated to the DC-link power transfer to the grid. In order to achieve this target, it maintains the DC-link voltage constant. The machine-side inverter is employed for the SCIG speed control. It allows the turbine operation at variable speed. Based on the DC-link constant voltage, this power structure allows fast SCIG torque adjustment using the vector control scheme [7].

IV. WECS Control

A. Vector Control

The generator torque control is achieved by considering a SCIG modeling in the rotor (*d*,*q*) frame [7]. The associated vector control structure separately drives the machine flux and torque, as they correspond to axis *d* and *q*, respectively. This control structure will be not further detailed as it is beyond the scope of this paper. Their dynamic performances are very good as the torque settling time is usually rated at hundred of milliseconds for the considered machine. Therefore, this low-level electrical dynamic is neglected in relation to the mechanical dynamic exhibited by the rotational speed evolution.

B. Speed Control and Torque-Speed Dependence Block (TSDB)

Wind turbines speed control is essential as it allows not only captured power control, but also over-speed protection and mechanical loads alleviation. The speed control in this paper is ensured by a PI controller. This configuration is classical, being employed in the majority of wind turbine control applications. Considering a measure of the wind speed, the rotational speed set-point corresponding to the optimal tip speed (maximum power capture) is given by the following relation:

$$\Omega_h^*\left(t\right) = \left(\lambda_{opt}\cdot i / R\right)\cdot v(t) = k_\lambda \cdot v(t). \tag{6}$$

The PI controller parameters influence on the controlled system dynamics will be analyzed in the following section. As already sated in [6], the wind speed has a double influence on the system. Beside of imposing the rotational speed set-point, it also influences directly the plant by changing the wind torque and consequently the rotational speed. Thus, if the PI controller is tuned to ensure a good tracking, then the closed-loop system dynamic behavior results far from being optimal, having an important overshoot, as described in [6].

In order to improve the WECS behavior, this paper proposes the employment of a so-called *torque-speed dependence block*, TSDB, who generates the electromagnetic torque reference as a linear dependence of the high-speed shaft rotational speed. Thus, by forcing the dependence between the electromagnetic torque and its rotational speed, the generator mechanical characteristic $T_{em}(\Omega_h)$ can be rotated with an arbitrarily imposed angle.

In this way, a supplementary degree of freedom is added to the system as its dynamic can be conveniently modified. This is done by changing the generator characteristic slope:

$$m = \tan\alpha = \Delta T_{em} / \Delta\Omega_h . \tag{7}$$

The equation that generates the linear dependence is:

$$T_{em} = m \cdot (\Omega_h - \Omega_c), \tag{8}$$

where Ω_c is the zero-torque rotational speed. This point is outputted by the PI speed controller, where the slope *m* is given as a user-supplied value. According to (2), a large value of *m* will speed up the plant.

The proposed control structure is given in Fig. 2. One can remark the wind speed double influence over the system and the TSDB presence that affects the electromagnetic torque reference.

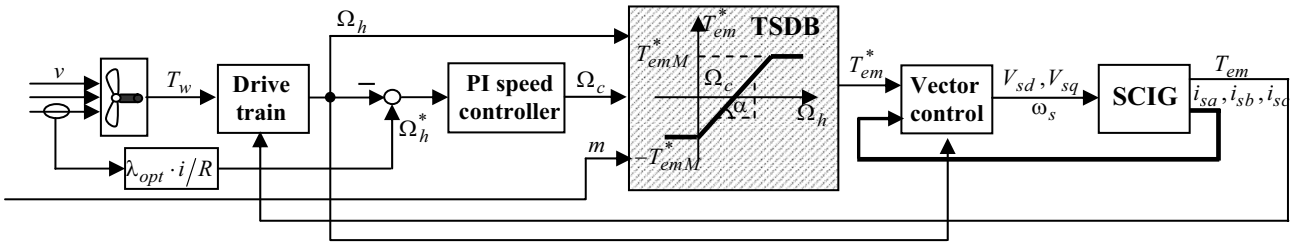


Fig. 2. WECS control block structure containing the TSDB.

C. Closed-loop System Analysis

The plant main dynamic is given in (5). By differentiating this equation around a steady-state operation point one obtains:

$$J\frac{\mathrm{d}}{\mathrm{d}t}\Delta\Omega_h = \frac{\partial T_w}{\partial\Omega_h}\Delta\Omega_h + \frac{\partial T_w}{\partial v}\Delta v - \frac{\partial T_{em}}{\partial\Omega_c}\Delta\Omega_c - \frac{\partial T_{em}}{\partial\Omega_h}\Delta\Omega_h \tag{9}$$

The following notations are used next. The wind torque and electromagnetic torque variation with rotational speed are denoted as $\partial T_w/\partial\Omega_h = k$ and $\partial T_{em}/\partial\Omega_h = m$ respectively, while the wind torque variation with respect to the wind speed is noted as $\partial T_w/\partial v = k_v$. The generator torque variation with the control input, Ω_c, is denoted by $\partial T_{em}/\partial\Omega_c = -m$. Note that k and k_v depend on the steady-state operating point.

Simple algebra leads to the rotational speed variation dependence on the wind speed and control input variations:

$$\Delta\Omega_h = k_v \cdot P \cdot \Delta v - m \cdot P \cdot (\Delta\Omega_h - \Delta\Omega_c), \tag{10}$$

where: $P = 1/(J_h \cdot s - k)$. (11)

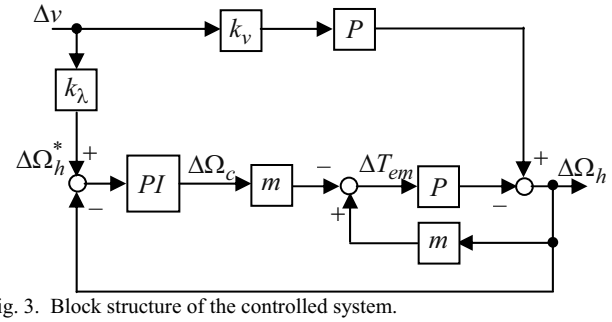


Fig. 3. Block structure of the controlled system.

In closed loop, the rotational speed variation is (see Fig. 3):

$$\Delta\Omega_h = k_v \cdot P \cdot \Delta v - m \cdot P\left[\Delta\Omega_h - H_{PI} \cdot (\Delta\Omega_h^* - \Delta\Omega_h)\right], \tag{12}$$

where $H_{PI} = K_P(1 + 1/(sT_I))$ is the controller transfer function.

Further, one can extract the rotational speed dependence on the wind speed:

$$\frac{\Delta\Omega_h}{\Delta v}=\frac{k_v\cdot P+m\cdot P\cdot H_{PI}\cdot k_\lambda}{1+m\cdot P\cdot\left(1+H_{PI}\right)}. \tag{13}$$

One aims at obtaining system dynamic performance of the maximum power tracking. This can be translated in good tracking of the wind speed variations. Thus, the rotational speed response to wind speed variations is imposed as being a standard second-order response:

$$\frac{\Delta\Omega_h}{\Delta v}=k_\lambda\frac{T_1 s+1}{T_0^2 s^2+2\xi T_0 s+1}, \tag{14}$$

where the closed-loop time constant, T_0, the damping, ξ, and the system's zero time constant, T_1, depend on the generator torque slope, *m*, through the following equations:

$$T_0=\sqrt{\frac{T_I\cdot J_h}{K_P\cdot m}},\ \xi=\frac{m\left(1+K_P\right)-k}{2}\sqrt{\frac{T_I}{K_P\cdot m\cdot J_h}}, \tag{15}$$

$$T_1=T_I\left(k_v/\left(K_P\cdot m\cdot k_\lambda\right)+1\right). \tag{16}$$

D. Design Issues

As it has already been seen, the generator torque slope determines the closed-loop system dynamic behavior. An increased value of *m* leads the decrease in both T_0 and T_1 time constants and the increase of the damping, ξ. This implies faster settling time of the wind turbine rotational speed together with lower overshoot. The dependences of the above listed variables on *m* for a typical operating point are given in Fig. 4.

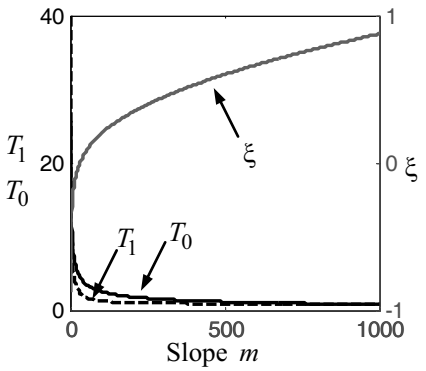


Fig. 4. Closed-loop time constant, T_0, damping, ξ and the system's zero time constant T_1 versus the generator torque slope, *m*.

The curves represented exhibit some kind of "saturation", meaning that large values of *m* will not significantly influence the system dynamics. Also, when *m* increases, the electromagnetic torque variations induced by the wind speed tracking process increase also. Using the transfer function from the wind speed to the electromagnetic torque given below (see Fig. 3):

$$\frac{\Delta T_{em}}{\Delta v}=m\cdot\frac{\Delta\Omega_h}{\Delta v}+m\cdot H_{PI}\cdot\left(\frac{\Delta\Omega_h}{\Delta v}-k_\lambda\right)=m\frac{\Delta\Omega_h}{\Delta v}+$$
$$+m\cdot\frac{K_P}{T_I}\cdot k_\lambda\cdot\left(T_1-2\xi T_0\right)\cdot\left(T_I s+1\right)\cdot\frac{\frac{T_0^2}{\left(2\xi T_0-T_1\right)}s+1}{T_0^2 s^2+2\xi T_0 s+1}, \tag{17}$$

and applying the final value theorem, one can estimate more accurately the torque variations dependence on parameter *m*, when a step variation of the wind speed is considered, *i.e.*, $\Delta v(s)=1/s$:

$$\Delta T_{em\_st}=\lim_{t\to\infty}\Delta T_{em}(t)=\lim_{s\to 0}s\Delta T_{em}(s)=$$
$$=m\cdot k_\lambda\left[1+\left(K_P/T_I\right)\cdot\left(2\xi T_0-T_1\right)\right]. \tag{18}$$

Relation (18) shows that the torque amplitude is proportional with the value of *m*.

Now, one can analyze the transfer function from the wind speed to the tip speed ratio variations. Starting from (4), one can obtain by differentiation around the optimal operating point (determined by λ_{opt}):

$$\Delta\Omega_h=\left(v\cdot i/R\right)\cdot\Delta\lambda+\left(\lambda\cdot i/R\right)\cdot\Delta v \tag{19}$$

Then, $\frac{\Delta\lambda}{\Delta v}=\left.\frac{R}{v\cdot i}\right|_{opt}\cdot\frac{\Delta\Omega_h}{\Delta v}-\frac{\lambda_{opt}}{v}$, and if employing (14), one obtains:

$$\frac{\Delta\lambda}{\Delta v}=\frac{\lambda_{opt}}{v}\cdot\left(\frac{T_1 s+1}{T_0^2 s^2+2\xi T_0 s+1}-1\right). \tag{20}$$

This latter relation shows that the tip speed variations will be reduced as the closed-loop system time constant becomes smaller (the term in the brackets approaches zero).

To conclude, the user-supplied parameter *m* significantly affects the closed-loop system dynamical behavior. It can be seen as a trade-off parameter between the tracking fidelity of the wind speed variations (power maximization purpose) and the alleviation of the electromagnetic torque high-frequency components (reliability purpose). From this point of view it plays the role of the weighting parameter α introduced in (1).

V. Simulation Results

The controlled WECS behavior is analyzed by numerical simulation using Matlab®/Simulink® software. The simulation block diagram corresponds to Fig. 2.

Two situations have been analyzed: first, when the electromagnetic torque reference is obtained directly as

output of the PI speed controller (classical WECS speed control structure) and a second one, when the electromagnetic torque is imposed by the above-presented dependence block (TSDB – see Fig. 2).

Concerning the first situation, the PI speed controller has been tuned around the optimal operating point at 9 m/s. Its parameters, $K_p = 733$ and $T_I = 2s$, correspond to a closed-loop time constant of about 2 seconds.

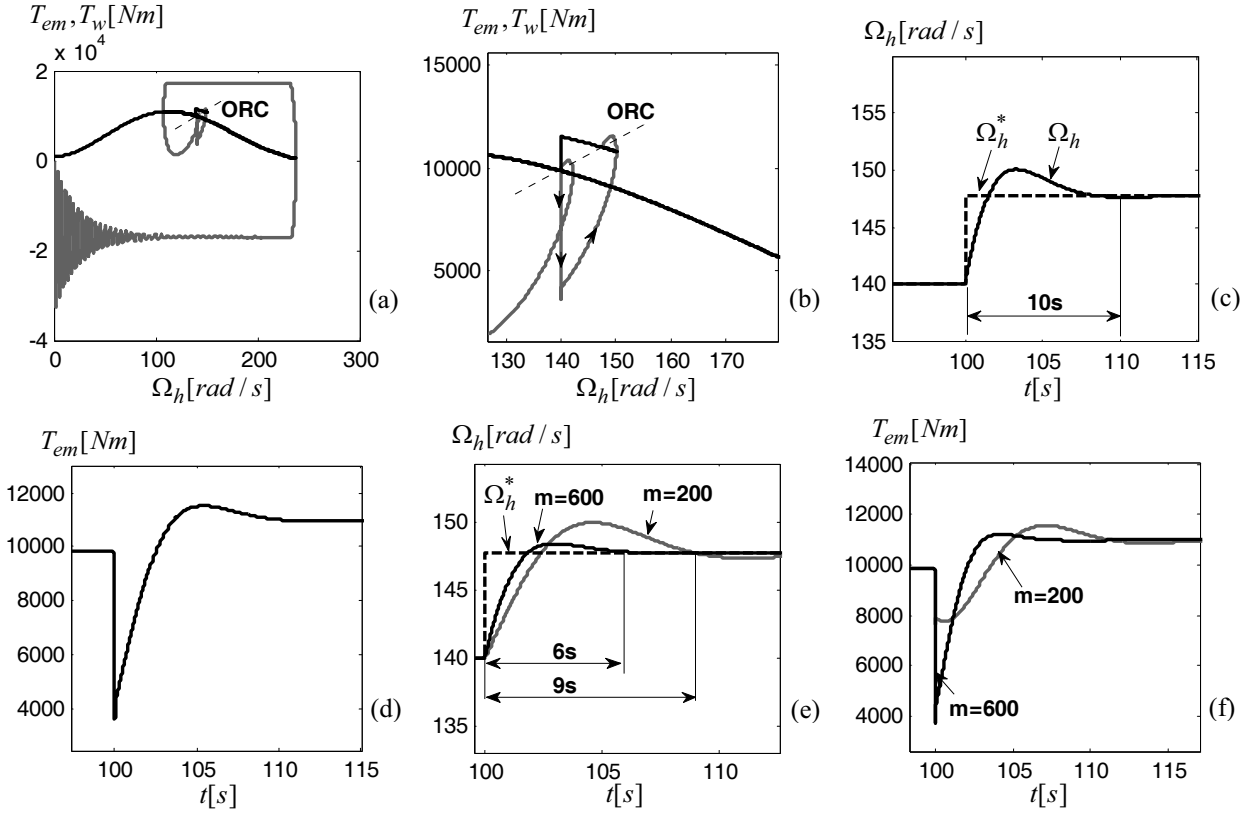


Fig. 5. Simulation results for a classical speed control structure with a vector controlled SCIG: a) trajectory of the operating point; b) zoom of a) around ORC; c) high-speed shaft rotational speed; d) electromagnetic torque variation. Dynamic closed-loop responses for slopes *m*=200 and *m*=600: e) high-speed shaft rotational speed; f) electromagnetic torque.

Fig. 5a presents the turbine start-up and its evolution towards ORC when the wind speed is constant at 9 m/s. The corresponding operating point trajectory on the $T(\Omega_h)$ plane indicates, first, a motoring regime. Then, when the generation regime is reached, the operating point evolves towards ORC. Fig. 5b contains the detailed system response to a positive step change in the wind speed (of 0.5 m/s). The dynamic response lasts for about 10 seconds – see Fig. 5c. One can note the quite large generator torque variations that correspond to significant control effort – see Fig. 5d.

Now, let us consider the second situation, when the speed controller output is passed through the TSDB, as indicated in Fig. 2. Two values of the characteristic slopes, *m*=200, and *m*=600, have been chosen. Because *m* parameter strongly influences the system dynamics, the parameters values of the PI speed controller have to be changed in order to obtain the same dynamic response. The new values for the speed controller are: $K_P = 1.2$ and $T_I = 0.8$.

The results are shown for the same operating point as in the first situation (positive wind speed step transition from 9 m/s to 9.5 m/s). As the system is nonlinear, for negative wind speed steps, the response differs slightly.

One can see that the obtained dynamic is better as the slope is larger (see Fig. 5e), but the inconvenient is the increase of electromagnetic torque variation (Fig. 5f). The overshoot is also significantly reduced. These results are coherent with the discussion in Section IV.D.

Fig. 6 shows the system response to stochastic wind speed. The turbine is fed with a stationary band-limited wind signal [10].

Large slope values lead to smaller variance of the speed error (better wind speed tracking performance) – see Fig.6a. Large slopes also lead to large SCIG torque variations, therefore to larger mechanical stress. This shows that the slope parameter *m* can be effectively used as a trade-off parameter between energy efficiency and mechanical stress alleviation.

Furthermore, for *m* values larger than 600, the generator torque variations become so large that the system operates in saturation. An upper limit value of *m* should be identified in order to avoid nonlinear behavior of the controlled system.

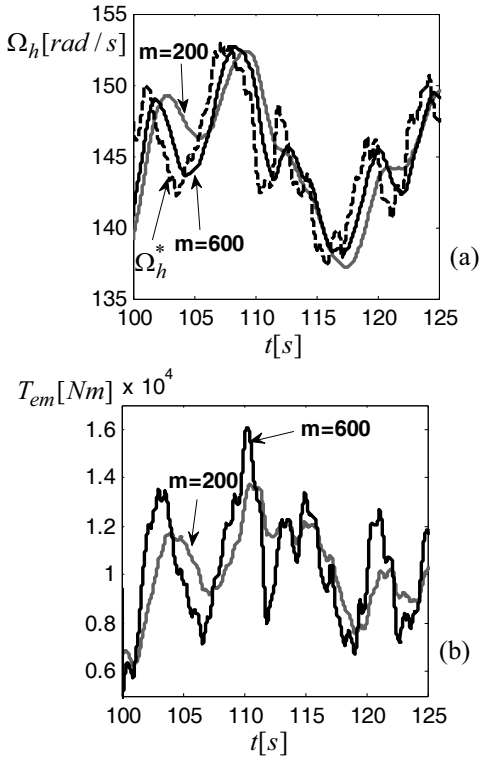


Fig. 6. System response to stochastic wind speed for slopes m=200 and m=600: a) high-speed shaft rotational speed; b) electromagnetic torque.

VI. Conclusions

This paper deals with dynamic power optimization for high-power WECS with a vector-controlled SCIG. To ensure faster dynamics, the electromagnetic torque is linked to the rotational speed by a linear dependence, similar to the one exhibited by the SCIG natural characteristic. This is achieved by passing the speed controller output through a torque-speed dependence block. The main difference with respect to the classical control structure is that the speed controller output is not the electromagnetic torque reference, but the zero-torque speed of the generator mechanical characteristic.

The influence of the torque-speed slope on the closed-loop dynamics has been thoroughly studied. Numerical simulations show that the controlled system dynamics can be conveniently adjusted using a single parameter: the torque-speed slope. It not only diminishes the speed overshoot, but also provides a mean of achieving simply a trade-off between the energy efficiency and mechanical efforts in the wind turbine partial-load region. The user-supplied value of the torque-speed slope can be conveniently adjusted corresponding to the wind speed value and turbulence. Small values can be provided by a supervisor that is concerned with the wind turbine reliability. Although high torque-speed slope values ensure faster turbine response, one must ensure their limitation in order to avoid excessive motoring regimes or too strong nonlinear behavior.

Future work will aim at finding a suitable gain scheduling law for the torque-speed slope with respect to the wind speed variation. The influence of this parameter on the power regulation loop in full-load regime should also be studied.

Appendix

Blade length R=45m, rotor inertia $J_w = 9 \cdot 10^6$ kg·m^2, rated wind speed 10.5 m/s, air density $\rho = 1.25$Kg/m^3, optimal tip speed ratio $\lambda_{opt} = 7$, SCIG rated power 2MW, maximal torque 17000 N·m, $J_g = 135$ kg·m^2, multiplier ratio $i = 100$.

References

[1] F. Bianchi, H. De Battista, and R.J. Mantz, "Wind turbine control systems – Principles, modelling and gain scheduling design," Springe–Verlag, London, 2006.

[2] T. Burton, D. Sharpe, N. Jenkins, and E. Bossanyi, "Wind energy handbook," *John Wiley & Sons*, New-York, 2001.

[3] H. Camblong, I. Martinez de Alegria, M. Rodriguez, and G. Abad, "Experimental evaluation of wind turbines maximum power point tracking controllers," *Energy conversion and Management* 47(18-19), 2006, pp. 2846-2858.

[4] T. Ekelund, "Modeling and linear quadratic optimal control of wind turbines," Ph.D. Thesis, Chalmers University of Göteborg, Sweden, 1997.

[5] I. Munteanu, A.I. Bratcu, N.A. Cutululis, and E. Ceangă, "Optimal Control of Wind Energy Systems – Towards a global approach.," *Springer-Verlag,* London, 2008.

[6] C. Vlad, I. Munteanu A.I. Bratcu, and E. Ceangă, "Anticipative Control of Low-Power Wind energy Conversion Systems for Optimal Power Regime," *Control Engineering and Applied Informatics*, 11(4), 2009, pp. 26-35.

[7] W. Leonhard, "Control of electrical drives," 3rd edition. Springer, Berlin Heidelberg, New-York, 2001.

[8] A.D. Diop, "Contribution au développement d'un simulateur électromécanique d'aérogénérateur : simulation et commande en temps réel d'une turbine éolienne de puissance moyenne à angle de calage variable," Ph.D. Thesis, Université du Havre, France (in French), 1999.

[9] I. Munteanu, "Contributions to the optimal control of wind energy conversion systems," Ph.D. Thesis, "Dunărea de Jos" University of Galați, Romania, 2006.

[10] C. Nichita, D. Luca, B. Dakyo, and E. Ceangă, "Large band simulation of the wind speed for real time wind turbine simulators," *IEEE Transactions on Energy Conversion* 17(4), 2002, pp. 523-529

An Overview of Temporal Coherence Methods in Real-Time Rendering

Daniel Scherzer

***Abstract*— Most of the power of modern graphics cards is put into the acceleration of shading tasks because here lies the major bottleneck for most sophisticated real-time algorithms. By using *temporal coherence*, i.e. reusing shading information from a previous frame, this problem can be alleviated. This paper gives an overview of the concepts of temporal coherence in real-time rendering and should give the reader the working practical and theoretical knowledge to exploit temporal coherence in his own algorithms.**

I. INTRODUCTION

One of the driving forces of computer graphics is to render physically correct images with rich visual effects. This usually requires large scenes with highly detailed models, as well as computationally intensive shading work to be incorporated in a modern rendering system. On the other hand, real-time rendering has the conflicting goal of creating a sequence of such images fast enough to still allow for continuous animation and user interaction. Here a limit of at least 60 frames per second is considered as sufficiently smooth for the human observer, which means the time available for one frame is about 16 milliseconds. All calculations necessary to create a frame have to fit into this time budget. This not only includes all the rendering algorithms we are concerned with in computer graphics, but may also contain the domain specific code of an application, artificial intelligence, input processing and sound rendering.

Although computer graphics hardware has made staggering advances in terms of speed and programmability, there still exist a number of algorithms that are too expensive to be computed in this time budget. A few important examples include physically correct shadows, depth of field and motion blur effects, or even an ambient occlusion approximation to the exact global illumination solution. The situation becomes worse when these effects are combined with large and complex scenes, in which the hidden geometry often consumes a significant portion of render time but contributes nothing to the final images.

One way to circumvent this hard time limit is to capitalize on *Temporal Coherence* (TC) and avoid redundant computations over time. TC is hereby defined as the existence of a correlation in time of the output of a given algorithm. For example, in a scene rendered at high frame rates, there is usually very little difference in the shading over visible surfaces between two consecutive frames, and the majority of surfaces are mutually visible (see Figure 1). Therefore, computing everything from scratch in every frame is potentially wasteful. Exploiting the coherence between adjacent frames and reuse intermediate or final shading result can therefore reduce the average shading cost of generating a single frame.

Daniel Scherzer is an Assistant Professor at the Institute of Computer Graphics and Algorithms, Vienna University of Technology
scherzer@cg.tuwien.ac.at

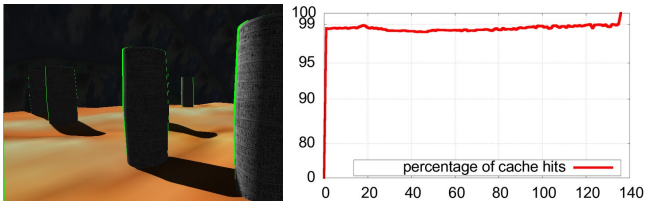


Fig. 1. Temporal coherence that exists in a game-like scene. *Left:* For a strafe-left movement the cache misses are shown in green. *Right:* Plot of the percentage of pixels found in the cache for each frame of the animation sequence.

In general, TC can be applied for achieving either of the following goals:

- *Acceleration*: A given algorithm can be accelerated by reformulating it as incremental in time, thereby amortizing the total workload over several frames. The output quality may be marginally degraded but the overall speed improvement is often promising.
- *Quality improvement*: The results of a given algorithm can be augmented by taking into account results computed in previous frames. By a slightly increase in render time, the quality of the result can often be significantly improved.
- *Reducing temporal aliasing*: When rendering frames, for each frame independent rasterizations are produced. This causes temporal aliasing and can result in strong flickering artifacts. TC can be applied to avoid this by introducing knowledge of previous rasterizations into the calculations for the current one, thereby allowing for temporal smoothing by dissallowing sudden changes in coherent regions.

These goals have in common that for a drastic change in the input some latency in the output may be introduced. In the acceleration case this requires a major refresh in the previously computed results, which may cause a sudden drop of framerate. In the quality improvement case, this means that over several frames only an approximate solution can be displayed before the algorithm converges. Fortunately with relatively high framerates and careful algorithmic designs, these problems can often be handled smoothly and unnoticed by the viewer. In addition, ongoing animation may also cause information from previous frames to be outdated. This has to be accounted for in order to avoid temporal artifacts such as after-images or tailing.

Aside from the fact that the redesign of algorithms to account for TC can be challenging, special care has to be taken to fit these algorithms to the massively parallel nature of modern graphics architectures.

II. BACKGROUND

The term *frame-to-frame coherence* was first introduced by Sutherland et al. [19] in his seminal paper "Characterization of Ten Hidden-Surface Algorithms", in which he describes various versions of coherence, like scan-line or area coherence that allow for more efficient rendering.

In *ray-tracing* reprojection of object space information can be used to allow to reuse information from the previous frame to accelerate animations [2], [1].

In *image-based rendering* TC allows to replace parts of a scene with image-based representations. Complex distant geometry can be replaced by so called impostors [6], [13]. A scene can also be factorized into multiple layers by accounting for differences in perception of fore-/background objects, as well as differences in the motion of objects [8]. The extreme case is frameless rendering, which only relies on TC instead of spatial coherence [3]. Each pixel is rendered independently based on the most recent input. Pixels stay visible for a random time-span. To avoid image tearing pixels are rendered in a random order.

In *image warping* images are used as a cache to be reused and warped into different views. In-between views can be calculated by morphing a number of reference images [4], [10], [9]. For each view color and depth are stored. Both images are warped into the new view to allow for small camera movements. Then the two images are composited together to compensate for most disocclusions [9]. A more involved representations is the *render cache* [22]. It is intended as an acceleration data structure for renderers that are too slow for interactive use. It is a point based structure, which stores previous results, namely 3d coordinates and shading information. By using reprojection, image space sparse sampling heuristics and by exploiting spatio-temporal image coherence these results can be reused in the current frame. Progressive refinement allows decoupling the rendering and display frame rates, enabling high interactivity. This approach was later extended with predictive sampling and interpolation filters [21] and finally accelerated on the GPU [20], [24].

III. IMAGE-SPACE REAL-TIME REVERSE REPROJECTION

Similar to Walters' render cache idea is Nehab et al.'s [11], [12] so-called *reprojection cache*, which is introduced as a way to accelerate real-time pixel shading in hardware rasterization renderers (see Figure 2). The main difference to the *render cache* is that the *reprojection cache* does not contain points but visible pixels in screen space (with additional data like depth). This is a very hardware friendly approach as this cache is just a viewport-sized off-screen buffer and can therefore reside in graphics memory without causing traffic between GPU and CPU. Another difference that fits perfectly to hardware is that this method uses reverse reprojection and

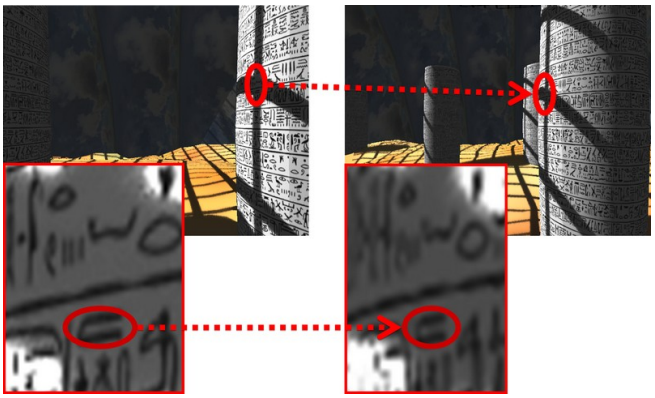


Fig. 2. Fast reprojection on the hardware is achieved by back-projecting each fragment from the current frame *(left)* into the cache *(right)* and incorporating the information found there to shade the current fragment.

therefore can use hardware texture filtering capabilities for sample retrieval. The *reprojection cache* approach described in this paper is similar to our concurrent work [14], where we specialized in improving shadow quality.

Reprojection is achieved by back-projecting each fragment $\mathbf{p}$ into the coordinate space of the previous frame – the space in which the reprojection cache was created. Consequently, if the camera moves, for every currently rendered fragment we have to find the corresponding position in the reproojection cache. Since we have the 3D position of our current fragment (in the post-perspective space of the current view), we can simply use the view ($\mathbf{V}$) and projection ($\mathbf{P}$) matrices and their inverses of the current and the last frame to do the transformation (back into the post-perspective space of the previous frame):

$$\mathbf{p}_{prev} = \mathbf{P}_{prev} * \mathbf{V}_{prev} * \mathbf{V}^{-1} * \mathbf{P}^{-1} * \mathbf{p} \qquad (1)$$

Here $\mathbf{p}$ is the fragment in the post-perspective space of the current frame. This fragment is transformed by $\mathbf{P}^{-1}$, the inverse projection matrix of the current frame, $\mathbf{V}^{-1}$, the inverse view matrix of the current frame (we are now in world space), $\mathbf{V}_{prev}$, the view matrix of the previous frame and finally by $\mathbf{P}_{prev}$, the projection matrix of the previous frame. After homogenization we are at the position the fragment would have had in the previous frame $\mathbf{p}_{prev}$. For moving objects, we can additionally store the object space transformation matrices or skinning matrices to do the backprojection step. Please note that all matrix transformations can be performed in the vertex shader and only the homogenization (division by the w-coordinate) has to happen in the fragment shader.

The obtained position will normally not be at an exact fragment center in the history buffer except for the special case that no movement has occurred. Consequently, filtering the history buffer for the lookup should be done. In practice, the bilinear filtering that graphics hardware offers shows good results.

For distinguishing between a cache hit and miss when reprojecting current fragments into the cache, the depth is used. If a cache value has a depth equal ($\pm\epsilon$) to the current

fragment's depth, a cache hit is assumed (see Figure 1) and information from the cache can be reused. Otherwise no temporally coherent information for this fragment is available.

Due to its speed and versatility this approach is the de facto standard for using TC in real-time rendering. In the following sections we will therefore discuss a selection of algorithms that are based on this method.

A. Discrete LOD Interpolation

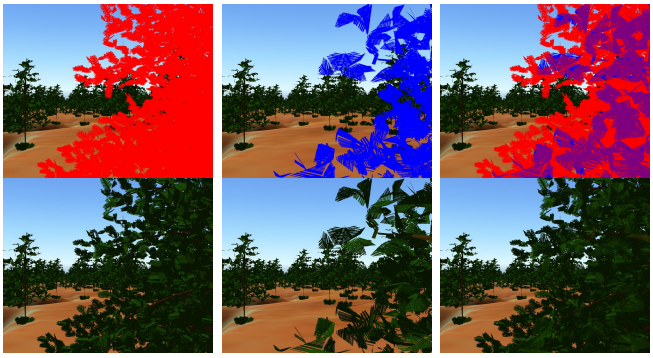


Fig. 3. LOD interpolation combines two buffers containing the discrete LODs to create smooth LOD transitions. *First and second column*: buffers; *last column*: combination. The top row shows the two LODs in red and blue respectively.

The idea of discrete level-of-detail (LOD) techniques is to use a set of representations with different levels of detail for one model and select the most appropriate representation for rendering at runtime. Due to memory constrains only a small number of LODs is being used and therefore switching from one representation to another can lead to noticeable popping artifacts. A solution to this problem is to replace the hard switch by a transition phase, in which the two representations are alpha blended in screen space [5].

Apart from other problems, this approach requires that the geometry (and the shaders) of both representations have to be rendered in this transition phase, thereby generating a higher rendering cost than the higher quality level alone would incur. To circumvent this Scherzer and Wimmer introduce LOD interpolation (see Figure 3) [16]. The idea is that by using TC the two LODs required during an LOD transition can be rendered in *subsequent frames*. Two separate render passes are used to achieve the transition phase between adjacent LOD representations: Pass one renders the scene into an off-screen buffer (called *LOD buffer*). For objects in transition we use one of the two LOD representations and render only a certain amount of its fragments (see Figure 4), depending one were in the transition (i.e., how visible) this object currently is. This is done via so-called *visibility textures*, which represent a visibility function for an object. In the next frame the same is done the other LOD representation and rendered into a second *LOD buffer*. The second pass combines these two *LOD buffers* (one from the current and one from the previous frame) to create the desired smooth transition effect.

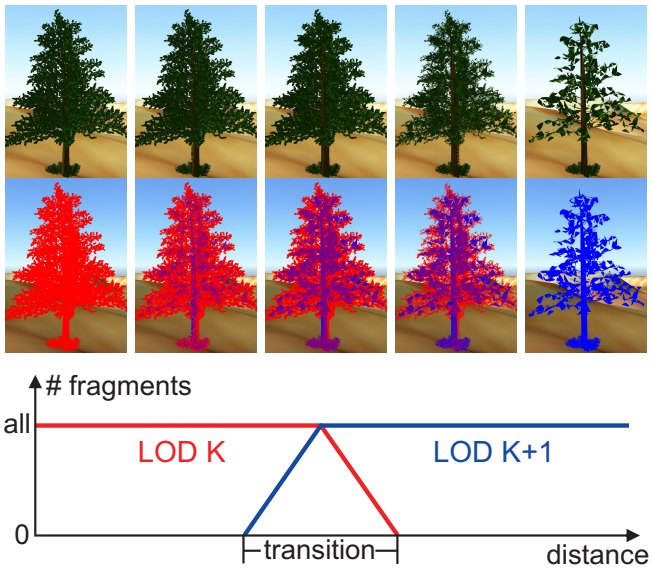


Fig. 4. Transition phase from LOD_k to LOD_{k+1}: *left:*LOD_k; *middle:* midway in the transition all fragments of both LODs are drawn; *right:* LOD_{k+1}; *Below:* First LOD_{k+1} is gradually introduced till all its fragments are drawn. Then LOD_k is gradually removed by rendering fewer and fewer fragments. The top two rows show the result of our method and a false color illustration.

We store a 3D visibility function per object (the *visibility texture*) and compare it to a visibility threshold to decide which fragments to discard. The visibility threshold $\tau \in [0..1]$ is given by the function depicted in Figure 4. Written as an equation: $\lambda : R^3 \times [0..1] \rightarrow \{true, false\}$

$$\lambda : (\mathbf{p}, \tau) \mapsto visTex(\mathbf{p}) > \tau \qquad (2)$$

is the function that evaluates for each fragment if it should be discarded. Here $\mathbf{p}$ is the object-space coordinate (before any animation is applied) of the fragment, τ the visibility threshold and $visTex$ is the lookup into the visibility texture. Note that even though the visibility function may be continuous, the thresholding operation gives a binary result and therefore no semi-transparent pixels appear.

By using different visibility textures, one can control in which way the fragments become visible. Examples include a uniform noise pattern, a function that decreases from the center outward, or any other function best suited to a given object. This has the effect that the amount and distribution of the visible fragments of an object can be controlled (see Figure 5).

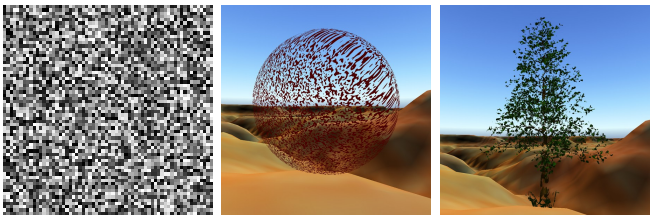


Fig. 5. A uniform noise visibility texture (*left*) applied to two different models with visibility $\tau = 0.5$).

B. Hard Shadows

Shadows are widely acknowledged to be one of the global lighting effects with the most impact on scene perception. They are perceived as a natural part of a scene and give important cues about the spatial relationship of objects.

Due to its speed and versatility, shadow mapping is one of the most used real-time shadowing approaches. The idea is to first create a depth image of the scene from the point of view of the light source (shadow map). This image encodes the front between lit and unlit parts of the scene. On rendering the scene from the point of view of the camera each fragment is transformed into this space. Here the depth of each transformed camera fragment is compared to the respective depth in the shadow map. If the depth of the camera fragment is nearer it is lit otherwise it is in shadow (See Figure 6).

The most concerning visual artifacts of this method originate from aliasing due to undersampling. The cause for undersampling is in turn closely related to rasterization that is used to create the shadow map itself. Rasterization uses regular grid sampling for rasterization of its primitives. Each fragment is centered on one of these samples, but is only correct exactly at its center. If the viewpoint changes from one frame to the next, the regular grid sampling of the new frame is likely to be completely different than the previous one. This frequently results in artifacts, especially noticeable for thin geometry and the undersampled portions of the scene called *temporal aliasing*.

This is especially true for shadow maps. Due to shadow map focusing, a change in the viewpoint from one frame to the next also changes the regular grid sampling of the shadow map. Additionally the rasterized information is not accessed in the original light-space where it was created, but in eye-space, which worsens these artifacts. This frequently results in temporal aliasing artifacts, mainly flickering (See Figure 6).

The main idea in [14] is to jitter the shadow map viewport differently in each frame and to combine the results over several frames, leading to a higher effective resolution.

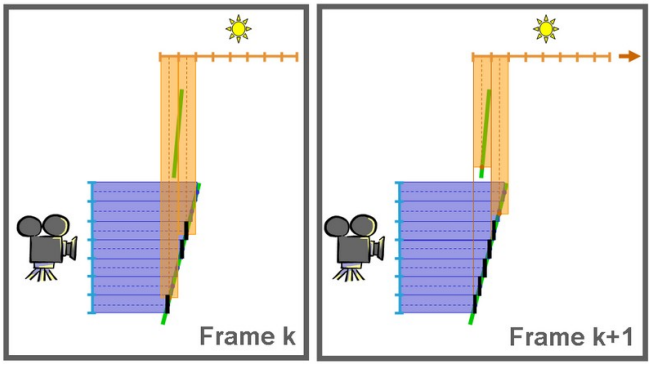


Fig. 6. If the rasterization of the shadow map changes (here represented by a right shift), the shadowing results may also change. On the *left* three fragments are in shadow, while on the *right* five fragments are in shadow. This results in flickering or swimming artifacts in animations.

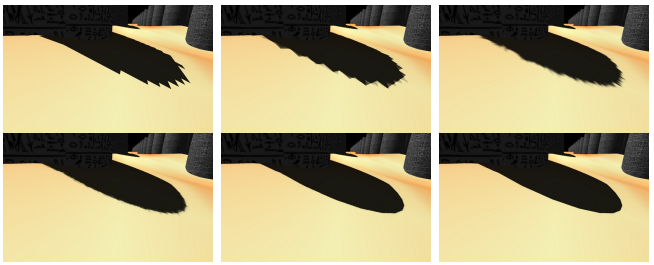


Fig. 7. Shadow adaption over time of an undersampled uniform shadow map after 0 *(top-left)*, 1 *(top-middle)*, 10 *(top-right)*, 20 *(bottom-left)*, 30 *(bottom-middle)* and 60 *(bottom-right)* frames.

In order to reduce temporal aliasing, each pixel is interpreted as a separate function $f(n)$ with the time as the input domain (usually represented by a frame number n). Temporal anti-aliasing is then done by smoothing this function. Smoothing itself is done by employing *exponential smoothing*:

$$s(n) = w * f(n) + (1 - w) * s(n - 1) \qquad 0 < w \leq 1 \quad (3)$$

Here w is a weight and $s(n-1)$ is the result of the previous evaluation. w allows balancing fast adaption of s to changing input parameters against temporal noise of the function. With increasing w, $s(n)$ depends more on the result of the current frame function and less on older frames and vice versa.

The shadow quality in this approach can actually be made to converge to a pixel-perfect result by optimizing the choice of the weight between the current and the previous frame result (stored in a reprojection cache). The weight is determined according to the *confidence* of the shadow lookup. The confidence is higher if the lookup falls near the center of a shadow map texel, since only near the center of shadow map texels it is very likely that the sample actually represents the scene geometry (see Figure 7 and 8).

C. Soft Shadows

In reality most light sources are area light sources and hence most shadows exhibit soft borders. *Light source sampling* introduced by Heckbert and Herf [7] creates a shadow map for every sample (each on a different position on the light source) and calculates the average (= soft shadow) of the shadow map test results for each fragment. The primary problem here is that the number of samples (and therefore shadow maps) to produce smooth penumbrae is huge and

Fig. 8. LiSPSM *(left)* gives good results for a shadow map resolution of 1024^2 and a viewport of 1680×1050, but temporal reprojection *(middle)* can still give superior results because it uses shadow test confidence *(right)*: $\mathrm{conf}_{x,y} = 1 - \max\left(|x - \mathrm{center}_x|, |y - \mathrm{center}_y|\right) \cdot 2$.

therefore this approach is slow. Typical methods for real-time applications approximate an area light by a point light located at its center and use heuristics to estimate penumbrae, which leads to soft shadows that are not physically correct. Here overlapping occluders can lead to unnatural looking shadow edges, or large penumbrae can cause single sample soft shadow approaches to either break down or become very slow

The main idea of our algorithm [15] is to formulate light source area sampling in an iterative manner, evaluating only a single shadow map per frame. We start by looking at the math for light source area sampling: Given n shadow maps, we can calculate the soft shadow result for a given pixel $\mathbf{p}$ by averaging over the hard shadow results s_i calculated for each shadow map. This is given by

$$\psi_n(\mathbf{p}) = \frac{1}{n}\sum_{i=1}^{n} s_i(\mathbf{p}). \tag{4}$$

We want to evaluate this formula iteratively by adding a new shadow map each frame, combining its shadow information with the data from previous frames stored in a so-called shadow buffer B_{prev}, and storing it in a new shadow buffer B_{cur}. With this approach, the approximated shadow in the buffer improves from frame to frame and converges to the true soft shadow result (see Figure 9).

Our approach has the following steps:

- The area sampling is done one sample per frame by creating a shadow map from a randomly selected position on the area light. For each screen pixel the hard shadow results obtained from this shadow map are combined with the results from previous frames (accumulated in the reprojection cache) to calculate the soft shadow for each pixel.
- When a pixel becomes newly visible and therefore no previous information is available in the reprojection cache, we use a fast single sample approach (PCSS with a fixed 4x4 kernel) to generate an initial soft shadow estimation for this pixel.
- To avoid discontinuities between sampled and estimated soft shadows, all the estimated pixels are augmented by using a depth-aware spatial filter to take their neighborhood in the *shadow buffer* into account.

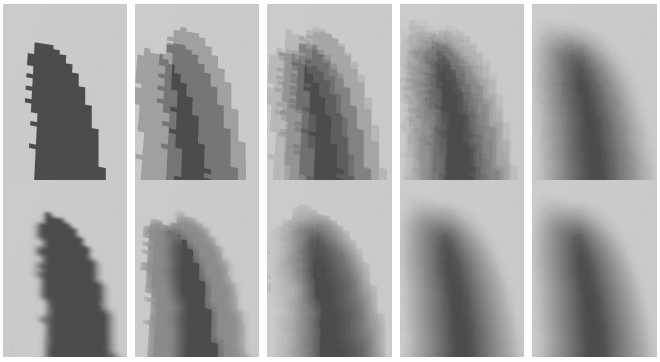


Fig. 9. Convergence after 1,3,7,20 and 256 frames. *Upper Row*: Sampling of the light source one sample per frame; *Lower Row*: Our new algorithm.

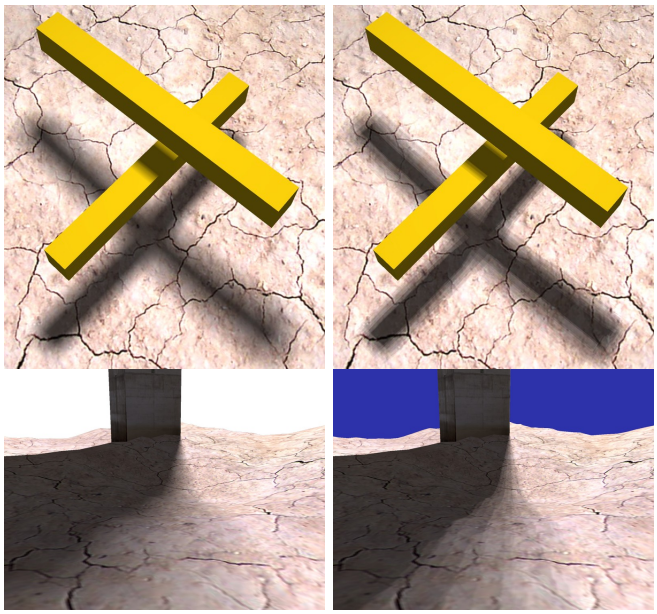


Fig. 10. *Left side*: Our Method. *Right Side*: PCSS 16/16. Overlapping occluders (upper row) and bands in big penumbras (lower row) are known problem cases for single sample approaches.

This results in a very fast soft shadow approach based on shadow maps that uses temporal reprojection for converging to the physical correct result (see also Figure 10).

IV. Conclusions

This paper has given a short introduction into the field of TC in real-time rendering. TC is of course also used in many other areas of real-time rendering.

Yang et al. [23] use it for amortizing supersampling: They maintain several samples from previous frames and combine them in the current frame using reprojection. In the majority of cases they can thereby avoid the computational cost of calculating multiple samples for each fragment.

Also analysis papers of the reprojection cache approach exist. Sitthi-Amorn et al. [17] analyse the potential performance gain achievable by introducing the reprojection cache. They find that a 3-pass algorithm (in contrast to the single pass or 2-pass algorithms used before) is more efficient to execute on current hardware. The problem when and how to update the *reprojection cache* (refresh policy) is investigated in [18]. They present automatic methods to select when and which samples to refresh using a parametric model that describes the way possible caching decisions affect the visual fidelity and the shader's performance.

But due to place constrains many interesting algorithms had to be omitted. This area is a very active field of research and there are still many algorithms in real-time rendering that could also benefit from accounting for TC in the algorithm design.

REFERENCES

[1] Stephen J. Adelson and Larry F. Hodges. Generating exact ray-traced animation frames by reprojection. *IEEE Comput. Graph. Appl.*, 15(3):43–52, 1995.

[2] S. Badt Jr. Two algorithms for taking advantage of temporal coherence in ray tracing. *VC*, 4:123–132, 1988.

[3] Gary Bishop, Henry Fuchs, Leonard McMillan, and Ellen J. Scher Zagier. Frameless rendering: double buffering considered harmful. In *SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 175–176, New York, NY, USA, 1994. ACM.

[4] Shenchang Eric Chen and Lance Williams. View interpolation for image synthesis. In *SIGGRAPH '93: Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 279–288, New York, NY, USA, 1993. ACM.

[5] Markus Giegl and Michael Wimmer. Unpopping: Solving the image-space blend problem for smooth discrete lod transitions. *Computer Graphics Forum*, 26(1):46–49, March 2006.

[6] Meister Eduard Gröller. *Coherence in Computer Graphics*. PhD thesis, Institute of Computer Graphics and Algorithms, Vienna University of Technology, Favoritenstrasse 9-11/186, A-1040 Vienna, Austria, 1992.

[7] Paul S. Heckbert and Michael Herf. Simulating soft shadows with graphics hardware. Technical Report CMU-CS-97-104, CS Dept., Carnegie Mellon U., Jan. 1997. CMU-CS-97-104, http://www.cs.cmu.edu/ ph.

[8] Jed Lengyel and John Snyder. Rendering with coherent layers. In *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 233–242, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co.

[9] William R. Mark, Leonard McMillan, and Gary Bishop. Post-rendering 3d warping. In *SI3D '97: Proceedings of the 1997 symposium on Interactive 3D graphics*, pages 7–ff., New York, NY, USA, 1997. ACM.

[10] Leonard McMillan and Gary Bishop. Head-tracked stereoscopic display using image warping. In *Proceedings SPIE, volume 2409*, pages 21–30, 1995.

[11] Diego Nehab, Pedro V. Sander, and John R. Isidoro. The real-time reprojection cache. In *ACM SIGGRAPH Sketch*, page 185, 2006.

[12] Diego Nehab, Pedro V. Sander, Jason Lawrence, Natalya Tatarchuk, and John R. Isidoro. Accelerating real-time shading with reverse reprojection caching. In *Graphics Hardware*, pages 25–35, 2007.

[13] Gernot Schaufler. Exploiting frame to frame coherence in a virtual reality system. In *VRAIS '96: Proceedings of the 1996 Virtual Reality Annual International Symposium (VRAIS 96)*, page 95, Washington, DC, USA, 1996. IEEE Computer Society.

[14] Daniel Scherzer, Stefan Jeschke, and Michael Wimmer. Pixel-correct shadow maps with temporal reprojection and shadow test confidence. In *Eurographics Symposium on Rendering*, pages 45–50, 2007.

[15] Daniel Scherzer, Michael Schwärzler, Oliver Mattausch, and Michael Wimmer. Real-time soft shadows using temporal coherence. *Lecture Notes in Computer Science (LNCS)*, November 2009.

[16] Daniel Scherzer and Michael Wimmer. Frame sequential interpolation for discrete level-of-detail rendering. *Computer Graphics Forum (Proceedings EGSR 2008)*, 27(4):1175–1181, June 2008.

[17] Pitchaya Sitthi-amorn, Jason Lawrence, Lei Yang, Pedro V. Sander, and Diego Nehab. An improved shading cache for modern GPUs. In *Proc. of Graphics Hardware*, pages 95–101, 6 2008.

[18] Pitchaya Sitthi-amorn, Jason Lawrence, Lei Yang, Pedro V. Sander, Diego Nehab, and Jiahe Xi. Automated reprojection-based pixel shader optimization. *ACM Trans. Graph.*, 27(5):127, 12 2008.

[19] Ivan E. Sutherland, Robert F. Sproull, and Robert A. Schumacker. A characterization of ten hidden-surface algorithms. *ACM Comput. Surv.*, 6(1):1–55, 1974.

[20] Edgar Velázquez-Armendáriz, Eugene Lee, Kavita Bala, and Bruce Walter. Implementing the render cache and the edge-and-point image on graphics hardware. In *GI '06: Proceedings of Graphics Interface 2006*, pages 211–217, Toronto, Ont., Canada, Canada, 2006. Canadian Information Processing Society.

[21] Bruce Walter, George Drettakis, and Donald P. Greenberg. Enhancing and optimizing the render cache. In *EGRW '02: Proceedings of the 13th Eurographics workshop on Rendering*, pages 37–42, Aire-la-Ville, Switzerland, Switzerland, 2002. Eurographics Association.

[22] Bruce Walter, George Drettakis, and Steven Parker. Interactive rendering using the render cache. In D. Lischinski and G.W. Larson, editors, *Rendering techniques '99 (Proceedings of the 10th Eurographics Workshop on Rendering)*, volume 10, pages 235–246, New York, NY, Jun 1999. Springer-Verlag/Wien.

[23] Lei Yang, Diego Nehab, Pedro V. Sander, Pitchaya Sitthi-amorn, Jason Lawrence, and Hugues Hoppe. Amortized supersampling. *ACM Trans. Graph.*, 28(5):135, 2009.

[24] Tenghui Zhu, Rui Wang, and David Luebke. A gpu-accelerated render cache. *Pacific Graphics, (Short Paper Session)*, October 2005.

Primary Investigation of Sound Recognition for a domotic application using Support Vector Machines

M. A. Sehili[1,2], D. Istrate[1], and J. Boudy[2]
[1]LRIT-ESIGETEL, 1 Rue du Port de Valvins,77210 Avon, France
[2]Telecom SudParis, 9 Rue Charles Fourier, 91000 Evry, France

***Abstract*— The advent of modern communications and the low cost of some kinds of devices have resulted in a desire to equip elderly peoples' homes with sensors to monitor their activities and be forewarned of abnormal situations. In such an environment, sound may represent a rich source of information that can be exploited and this is considered as one of the most ergonomic and least intrusive solutions. However, this solution is often adversely affected by noise that is to say, mostly sounds of a type not taken into account in the creation of this system. Several methods were used to make it possible to classify sounds. In this work we tested Support Vector Machines to classify sounds in a domotic environment.**

I. Introduction

Sound classification is a problem of pattern recognition where one aims to distinguish the class of a given sound from other classes. In a domotic environment, there are many kinds of daily sounds which require detection in order to obtain information about the status of elderly people and their activities. There are also some sounds considered as noise that the system should ignore. Speech is considered as one of the most informative sounds, it is by far the most important class for a telemonitoring system. In fact, a speech signal can carry useful information like emotions and may contain a distress expression. This is what has motivated researchers to attempt sound classification in a hierarchical fashion as in [7] where speech was first distinguished from other sounds before being transmitted to a second classification engine.

This research work take place in the framework of the Sweet-Home project which search to provide a domotic HMI based on direct/indirect Speech/Sound recognition. The aim of this project is the safety of the personns and of goods using audio techniques. The interesting sound classes for this project are everyday life sounds (door clap, phone ring, dshes sounds,...) and abnormal sounds (screams, glass breaking, object falls,...).

The problem of sound classification can be compared to that of speaker identification as both are a multiclass pattern recognition task and rely on extracting and modeling the relevant features from the signal in order to differentiate them. In recent years, several statistical methods which have been successfully used for speaker identification, for example: Hidden Markov Models (HMMs), Gaussian Mixture Models (GMMs) [4] and Dynamic Time Warping (DWT), were used for sound classification. Previous work of the ANASON team applied GMMs to sound classification following the model described above. A combination of two or more classification methods was also used like in [12] and [5].

Support Vector Machines (SVMs) is a hyperplane based method that has gained increasing attention in the pattern recognition community over the last few years and has been successfully applied to tasks like speaker identification and verification, and face recognition. From a theoretical point of view, this discrimination method is quite robust. For a linear classification problem, it attempts to choose a hyperplane that best separates data points from two classes. Moreover, it has been shown to perform a non-linear classification with accuracy via the use of appropriate Kernel functions. This makes the SVMs extremely valuable for the task of sound classification.

II. Support Vector Machines

SVMs belong to the family of binary classifiers. That means that an SVM attempts to assign one of two labels to data points from two distinct classes. The goal is to assign the exact label to each point given a set of labeled examples used to train the classifier.

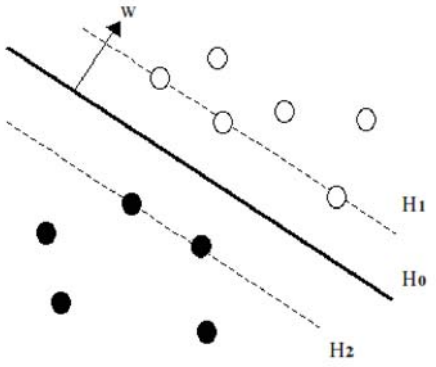


Fig. 1. Example of a linear classifier

The basic idea behind this method is to find a decision surface hyperplane which maximizes the margin between positive and negative examples. This implements the principle of structural risk minimization (SRM) [2] (Figure 1). The hyperplane H_0 and the points which are mapped on it satisfy:

$$w \cdot x + b = 0 \tag{1}$$

The vector w is the normal to the hyperplane and b is the bias of the hyperplane from the origin. Given a set of N training examples (x_i, y_i), where $x_i \in R^p$ are the points and $y_i \in \{-1, 1\}$ are the associated labels, we need to find the maximum margin subject to the constraints:

$$w \cdot x_i - b \geq 1$$

for $y_i = 1$, and

$$w \cdot x_i - b \leq -1$$

for $y_i = -1$, which can be written as:

$$y_i(x_i \cdot w + b) - 1 \geq 0, \forall i \tag{2}$$

We find that the distance between the two margins H_1 and H_2 is $\frac{2}{||w||}$. Thus, the problem can be stated as minimize $||w||$ subject to (2).

The problem can be put as a quadratic programming problem as follows:

$$L_p = \frac{1}{2}||w||^2 - \sum_{i=1}^{N} \alpha_i y_i (x_i \cdot w + b) + \sum_{i=1}^{N} \alpha_i \tag{3}$$

where the α_i are the Lagrange multipliers.

In figure 1 it can be seen that few examples can be found on the margins H_1 and H_2. These are the support vectors and their associated α_i are greater than 0.

In most cases the data examples are not perfectly separable. In other words, there exists no hyperplane that can separate all points without making any erroneous classification. This has motivated to introduce *slack* variables, ξ_i, to allow some degree of misclassification for some examples while still maximizing the distance to the nearest cleanly separated examples. The problem becomes:

Minimize:

$$\frac{1}{2}||w||^2 + C \sum_{i=1}^{N} \xi_i$$

subject to:

$$y_i(x_i \cdot w + b) \geq 1 - \xi_i, \forall i \tag{4}$$

where C is the penalty parameter of the error term.

The above theory works well as long as the data is linearly separable. In many problems, including sound classification, the data is far from being linearly separable. To deal with such problems one solution is to map the data into an extremely high dimensional feature space so that a linear separation becomes possible. However, dealing with data from a high dimensional feature space can easily lead to high computation costs [1]. This can be avoided by using Kernel functions. Typically used Kernel functions are:

$$\text{Linear:} \qquad K(x, y) = x \cdot y \tag{5}$$

$$\text{Polynomial:} \qquad K(x, y) = (\gamma x \cdot y + c)^p \tag{6}$$

$$\text{RBF:} \qquad K(x, y) = \exp(-\Gamma |x - y|^2) \tag{7}$$

The final decision function takes the form:

$$f(x) = \sum_{i=1}^{N_{sv}} \alpha_i y_i K(x, x_i) + b \tag{8}$$

and the sign of the function f represents the label of the input vector x.

III. Application to sound classification

In most cases a system has to deal with more than two classes of sounds. However SVM is a binary classification method. Although there exists a variant of SVM which can do multiclass classification, most researchers prefer splitting the problem into multiple binary problems and then using a binary classifier for each problem. There are two schemes most commonly used to do this; the one-to-all scheme and the one-to-one scheme. In the one-to-all scheme, C classifiers are created to represent C classes. Each classifier is trained by labeling examples from one class as $+1$ and examples from all the other classes as -1. An input example is thus evaluated using all the classifiers and is attributed to the class that yields the best distance. In the one-to-one scheme, a classifier is trained for each couple of classes and the final decision is achieved using a tree structure or a Directed Acyclic Graph (DAG) [6].

In most cases, a sound consists of more than one vector (i.e. frame). In [13] where SVMs are applied to speaker identification, the score of an utterance of N vectors is simply the arithmetic mean of the scores of the vectors it contains:

$$S = \frac{1}{N} \sum_{j=1}^{N} (\sum_{i} \alpha_i y_i K(x_j, x_i) + b) \tag{9}$$

Nevertheless we can also classify a sound using a majority voting on its vectors. This technique allow avoiding the influence of only some vectors missclassified.

Another way to use SVMs is to use an ensemble of classifiers. This may be very fruitful for sound classification especially when data is noisy. The idea is to obtain a set of classifiers for the same classification problem [13]. This can be achieved using bootstrapping or boosting [9].

A. Acoustical parameters

The SVM are not applied directly on the time signal but on spectral extracted vectors named acoustical parameters. The acoustical parameters can be the MFCC (Mel Frequency Cepstral Coefficients), LFCC (Linear Frequency Cepstral Coefficients), LPC (Linear Prediction Coefficients), LPCC (Linear Prediction Cepstral Coefficients), etc. In this paper we have used LFCC because are more adapted for for sound with high frequencies components. LFCCs are cepstral coefficients commonly used in speaker/speech recognition systems. They are commonly calculated as shown in figure 2 [10].

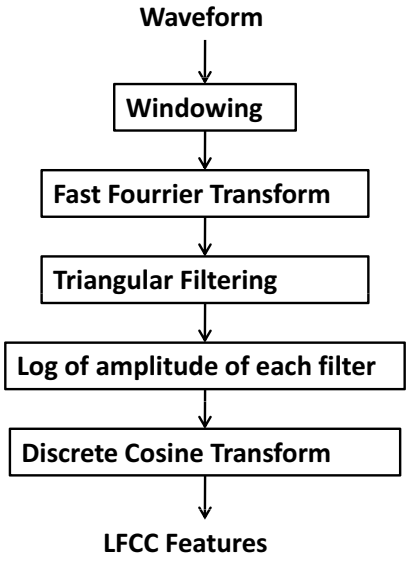


Fig. 2. Steps to derive LFCC

In the first step the signal is divided into frames, usually by using a rectangular windowing function at fixed intervals and overlap. Thus, each frames can be considered as a cepstral feature vector. The discrete Fourier Transform is then applied to each frame and triangular filter of uniformly spaced frequency bins are applied (Figure 3). The logarithm is computed on each output energy of each triangular filter. The components are finally decorrelated using the Discrete Cosine Transform. This has the advantage to reduce the final number of features in each vector.

IV. First Experiments

In order to experiment with SVMs for sound classification we have used the SVM-light library [8]. We first made a test on a part of the dataset created by the ANASON team. The dataset consists of seven categories of sound related to daily human activities. Table I shows the classes used in these experiments.

These sounds are 16kHz, 16 bits wav files. For this test, 24 order LFCCs (Linear Frequency Cepstral coefficients), energy and Zero Crossing Rate (ZCR) features were used. The frames were 16 ms of length with an overlap of 50%.

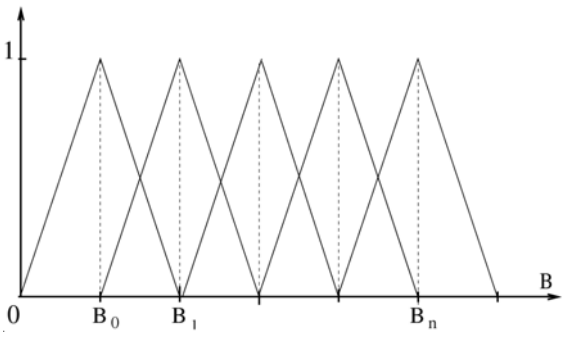


Fig. 3. Uniform frequency scale

TABLE I
Classes of sound from the dataset

Sound category	# of files
Cough	42
Door bell	14
Laugh	10
Sliding door	19
Sneeze	26
Snore	20
Yawn	21

The multiclass scheme used is the one-to-one, so a classifier is trained for each pair of classes. For each class, 50% of files are used for training and 50% for testing. To attribute a sound to a class, we first used the method consisting of calculating the sum the scores obtained by its vectors and then and then choosing the class according to the sign of the sum. This strategy yielded poor results. We then adopted a majority voting strategy which improved them moderately.

V. Results

The proposed algorithm was evaluated on the data base through the good classified rate. The accuracy of the whole database is obtained by dividing the number of correctly classified files by the total number of files. Table II shows the results obtained.

The method used is time consuming because of the non-linear kernel (RBF in our case) where almost all training examples are retained as support vectors. This results in huge models.

In order to better use SVMs and and improve the performances, many methods can be used to train a model like the use of hold-out set or cross-validation [11]. In this work we used the techniques described in [3] which consist in scaling, grid search and cross-validation.

The goal of scaling is to constraint each feature value to be in a specific range, for example $[-1, +1]$ or $[0, 1]$. This has the advantage to avoid features with greater values dominating those smaller values and to avoid numerical difficulties during calculation [3]. A grid search is used to find the couple of C and Γ which achieve the best accuracy on training data. Many combinations of these two parameters

TABLE II
The scores obtained with the first tests using two strategies of classification

	Classification strategy	
	Score sum	Majority voting
Cough	0.33	0.57
Door bell	0.57	1.00
Laugh	1.00	1.00
Sliding door	0.00	0.00
Sneeze	0.15	0.23
Snore	0.40	0.80
Yawn	0.18	0.18
Whole dataset	0.31	0.48

are thus used to train and test a classifier. One way to do this is to split the training data into two parts, train a classifier using one part and use the rest of data to determine which values of C and Γ allows for better performance.

A better way to determine the best parameters is to use cross-validation. In n-fold cross-validation the straining dataset is split into n subsets of equal size. Each subset is then used to test the classifier trained on the other subsets. In our experiments we used 5-fold cross validation.

Tables III shows the results obtained after using the procedures above. It can be seen that these results outperform the previous one. Furthermore, in table III, and contrary to table II, the performances of the two strategies are almost comparable. This is due to scaling the data before training and test. We have also noticed that the models obtained after scaling the data are fairly of smaller size than those obtained with non scaled data. This may be very interesting for real-time systems as the time required to classify one vector is closely related to the size of the model.

TABLE III
THE SCORES OBTAINED AFTER SCALING THE DATA AND USING CROSS-VALIDATION

	Classification strategy	
	Score sum	Majority voting
Cough	0.90	0.95
Door bell	1.00	1.00
Laugh	1.00	1.00
Sliding door	0.20	0.00
Sneeze	0.38	0.38
Snore	0.70	0.70
Yawn	0.18	0.18
Whole dataset	0.61	0.60

VI. CONCLUSIONS

This paper presents an application of SVMs to classify sound in a domotic environment. The sound classification is a multiclass problem but SVM are binary classifiers; two techniques was used one-against-one and one-against-all. The use of techniques like scaling and detecting the best parameters by using cross-validation allows to improve the performances. Although the first obtained results are encouraging, there are still several methods that can be used to better exploit SVMs and deal with the noise like the use of ensemble of classifiers with bootstrapping or boosting.

Next tests will aim to evaluate the noise influence on the SVM recognition performances and also the possibility to combine GMM with SVM in order to obtain a better system through score fusion.

ACKNOWLEDGMENTS

We would like to thank the ANR (French National Research Agency) and, especially VERSO program, for funding the Sweet-Home project, the framework of this research activity.

REFERENCES

[1] Joseph Picone Aravind Ganapathiraju. Hybrid svm/hmm architectures for speech recognition. 2000.

[2] Burges Christopher J. C. A tutorial on support vector machines for pattern recognition. *Data Min. Knowl. Discov.*, pages 121–167, 1998.

[3] Chih-Chung Chang Chih-Wei Hsu and Chih-Jen Lin. A practical guide to support vector classification. 2003.

[4] Richard C. Rose Douglas A. Reynolds. Robust text-independent speaker identification using gaussian mixture speaker models. *IEEE Transactions on Speech and Audio Processing*, pages 72–80, 1995.

[5] Zhou Xianzhong Luo Wen He Xin, Guo Ling. Hybrid support vector machine and general model approach for audio classification. In *ISNN '07: Proceedings of the 4th international symposium on Neural Networks*, pages 434–440, 2007.

[6] Seong-Whan Lee Hyeran Byun. A survey on pattern recognition applications of support vector machines. *International Journal of Pattern Recognition and Artificial Intelligence*, pages 459–486, 2003.

[7] D. Istrate J.E. Rougui and W. Souidene. Audio sound event detection for distress situations and context awareness. *31st Annual International Conference of the IEEE EMBS*, pages 3501–3504, 2009.

[8] Thorsten Joachims. Making large-scale support vector machine learning practical, 1998.

[9] Hyun-Chul Kim, Shaoning Pang, Hong-Mo Je, Daijin Kim, and Sung Yang Bang. Constructing support vector machine ensemble. *Pattern Recognition*, 36(12):2757 – 2767, 2003.

[10] Beth Logan. Mel frequency cepstral coefficients for music modeling.

[11] John C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pages 61–74. MIT Press, 1999.

[12] Hocine Bourouba Rafik Djemili, Mouldi Bedda. A hybrid gmm/svm system for text independent speaker identification. *International Journal of Computer Science and Engineering*, pages 22–28, 2007.

[13] Zhaohui Wu Zhenchun Lei, Yingchun Yang. Ensemble of support vector machine for text-independent speaker recognition. *International Journal of Computer Science and Network Security*, pages 163–167, 2006.

Weighted Moments Based Identification of DC Motor

Dorin Sendrescu, *Member, IEEE,* Constantin Marin, *Member, IEEE,* and Emil Petre, *Member, IEEE*

***Abstract*— In this paper one presents an algorithm for a DC motor parameters identification from sample data using the weighted power moments of the system output signal. While most of the latest methods used in identification utilize a discrete-time model, the moments method is an alternative approach to directly identify a continuous-time model from discrete-time data. The method defines a set of relationships between the power series coefficients of a stable transfer function and the power moments of the output signal of this system. Based on these relations, an algorithm for off-line parameter identification is developed. The method is applied to identify the parameters of a real experimental platform.**

I. Introduction

THE ADVANCE of digital computers and the availability of digital data furnished by the acquisition boards, most system identification algorithms usually aim at identifying the parameters of discrete-time models based on sampled input-output data. Over the last period there has been an increasing interest in continuous-time approaches for system identification from sampled data. Identification of continuous-time models is indeed a problem of considerable importance in various disciplines [3]. A simplistic way of estimating the parameters of continuous-time models by an indirect approach is to use the sampled data to first estimate a discrete-time model and then convert it into an equivalent continuous-time model. However, the second step, i.e. obtaining an equivalent continuous-time model from the estimated discrete-time model, is not always easy.

Difficulties are encountered whenever the sampling time is either too large or too small [11]. Whereas a large sampling interval may lead to loss of information, a small sampling period may create numerical problems because the poles are constrained to lie in a small area of the z-plane close to the unit circle. Some conversion methods use the matrix logarithm which may produce complex arithmetic when the matrix has negative eigenvalues. Moreover, the zeros of the discrete-time model are not as easily transformable to continuous-time equivalents as the poles are [9, 10]. In every tuning algorithm, the most difficult phase is the identification one, the whole control design depending on it. We can underline two approaches of identification algorithms: on-line identification algorithms and off-line identification algorithms. In on-line identification approach, the result is obtained in the same moment with a new observation data acquisition. The on-line identification deals with parametric methods (deterministic or stochastic), which identify the parameters of a mathematical model with a structure apriori known. The main on-line methods can be found in [2], [5], [8].

In off-line identification approach it is possible to identify both the structure of linear time invariant systems and the parameters of the mathematical model using observations over a larger time interval, including the steady state. The moments method presented in this paper is an off-line integral method.

DC motors have long been widely used in many industrial applications. A dc motor can be considered as a single input, single output (SISO) system having torque-speed characteristics compatible with most mechanical loads. This makes a dc motor controllable over a wide range of speeds by proper adjustments of its terminal voltage. Mathematical modeling is one of the most important and often the most difficult step towards understanding a physical system. In modeling a dc motor, the aim is to find the governing differential equations that relate the applied voltage with the produced speed of the rotor and to determine the parameters of the model. System identification of dc motors is a topic of great practical importance, because for almost every servo control design a mathematical model is needed. The motor parameters might be subject to some time variations and therefore, a mathematical model that is accurate at the time of the design may not be accurate at a later time.

This paper is structured as follows. Section II, describes the classical methods of moments used for identification. Section III, describes the method of weighted moments and analyzes the advantages of the developed identification algorithm. In section IV, the identification algorithms are applied to the parameter identification of a dc motor and are compared to the simulation results. Finally, conclusions of the paper are summarized in section V.

This work was supported by the National University Research Council - CNCSIS, Romania, under the research projects ID 786, 2007 (PNCDI II).

D. Sendrescu is with the Automatic Control Department, University of Craiova, A.I. Cuza, 200585, Romania (e-mail: dorins@ automation.ucv.ro).

C. Marin is with the Automatic Control Department, University of Craiova, A.I. Cuza, 200585, Romania (e-mail: cmarin@ automation.ucv.ro).

E. Petre is with the Automatic Control Department, University of Craiova, A.I. Cuza, 200585, Romania (e-mail: epetre@ automation.ucv.ro).

II. Identification Based on Moments Method

A. Definition of Moments

The moments problem is a classical one in the functional analysis [1], [4], [6]. The moments constitute the basis for a non classical representation of linear systems. The characterization of an impulse response by its moments is equivalent to the moment characterization of a probability density function. In the following one consider an original function $y(t)$ with index of convergence $\sigma_0 \le 0$ with the following Laplace transform:

$$Y(s) = L\{y(t)\} = \int_0^\infty y(t)e^{-st}dt,\ \mathrm{Re}(s) > \sigma_0 \tag{1}$$

Denoting by $\varepsilon(t)$ deviation of $y(t)$ from steady state, $\varepsilon(t) = y(\infty) - y(t)$, $t \ge 0$, one defines the j-order power moment [7] (that will be called the classical moment) for function $y(t)$:

$$m_j = \int_0^\infty \frac{(-t)^j}{j!}(y(\infty) - y(t))dt = \int_0^\infty \frac{(-t)^j}{j!}\varepsilon(t)dt \tag{2}$$

The Laplace transform of error $\varepsilon(t)$ is:

$$\begin{aligned} &E(s) = L\{\varepsilon(t)\} = L\{y(\infty) - y(t)\} = \\ &\int_0^\infty \varepsilon(t)e^{-st}dt,\ \mathrm{Re}(s) > \sigma_0 \end{aligned} \tag{3}$$

The complex function e^{-st} is holomorphic in the complex plane s and it has an ordinary point to infinity then, the Taylor series around s=0 is uniform convergent with radius ∞, so:

$$e^{-st} = \sum_{j=0}^\infty \left(\frac{(-t)^j}{j!}\right)s^j \Leftrightarrow e^{-st} = \sum_{j=0}^\infty \left(\frac{(-s)^j}{j!}\right)t^j \tag{4}$$

Because $\varepsilon(t)$ is bounded, multiplying (4) with $\varepsilon(t)$ one get an uniform convergent series in respect with t,

$$\varepsilon(t)e^{-st} = \sum_{j=0}^\infty s^j \frac{(-1)^j}{j!}t^j\varepsilon(t) \tag{5}$$

which can be integrated term by term on the interval $t \in [0, \infty]$,

$$\begin{aligned} &\int_0^\infty \varepsilon(t)e^{-st}dt = \sum_{j=0}^\infty m_j s^j = E(s) = \\ &L\{y(\infty) - y(t)\},\ \forall t \ge 0,\ \forall s \in C \end{aligned} \tag{6}$$

But

$$L\{y(\infty) - y(t)\} = \frac{y(\infty)}{s} - Y(s) \Rightarrow$$

$$Y(s) = \frac{y(\infty)}{s} - \sum_{j=0}^\infty m_j s^j \tag{7}$$

B. Identification Algorithm

Suppose first that $y(\infty)$ is known and bounded. The transfer function that we want to identify is a stable rational function with non-minimum phase with unknown order and parameters: n, m, a_k, b_k and $a_0 \ne 0$, $b_0 \ne 0$.

$$\begin{aligned} &H(s) = \frac{b_0 + b_1 s + ... + b_m s^m}{a_0 + a_1 s + ... + a_n s^n} = \frac{M(s)}{L(s)}, \\ &b_0 \ne 0;\ a_0 \ne 0;\ n \ge m \end{aligned} \tag{8}$$

If the input signal is a step function $U(s) = \frac{\Delta u}{s}$ then, if $y(\infty)$ =finite $\Rightarrow$ $a_0 \ne 0$ and, if $y(\infty) \ne 0$ $\Rightarrow$ $b_0 \ne 0$, there are no poles and zeros in the origin of the complex plane.

In these conditions one can normalize the transfer function coefficients:

$$\begin{aligned} &H(s) = K\frac{1 + b_1' s + ... + b_m' s^m}{1 + a_1' s + ... + a_n' s^n}, \\ &b_k' = \frac{b_k}{b_0};\ a_k' = \frac{a_k}{a_0}; k \ge 1; K = \frac{b_0}{a_0} \end{aligned} \tag{9}$$

Developing in power series of s the function 1/H(s) one get:

$$\frac{1}{H(s)} = \frac{L(s)}{M(s)} = \sum_{k=0}^\infty C_k s^k,\ |s| < \sigma_z = R \tag{10}$$

So $a_0 + a_1 s + ... + a_n s^n = (b_0 + b_1 s + ... + b_m s^m)(C_0 + C_1 s + ...)$ and by identification term by term and normalizing, one obtains:

$$a_k' = C_k' + b_1' C_{k-1}' + ... + b_j' C_{k-j}' + ... + b_k',\ 0 \le k \le n \tag{11}$$

$$0 = C_k' + b_1' C_{k-1}' + ... + b_j' C_{k-j}' + ... + b_m' C_k',\ k \ge n+1 \tag{12}$$

From

$Y(s)=H(s)U(s)$, $U(s)=\frac{\Delta u}{s}$ and $\frac{y(\infty)}{s} - H(s)\frac{\Delta u}{s} = \sum_{j=0}^\infty m_j s^j$

one get

$$\frac{y(\infty)}{s} - H(s)\frac{\Delta u}{s} = \sum_{j=0}^\infty m_j s^j \tag{13}$$

From (10) and (13) one obtains:

$$y(\infty)\sum_{k=0}^\infty C_k s^k - \Delta u = \left(\sum_{j=0}^\infty m_j s^{j+1}\right)\left(\sum_{k=0}^\infty C_k s^k\right) \tag{14}$$

and by identification term by term:

$$\begin{aligned} &C_0 = \frac{\Delta u}{y(\infty)} = \frac{1}{K} = \frac{a_0}{b_0} \\ &C_k' = m_0' C_{k-1}' + m_1' C_{k-2}' + ... + m_{k-1}' \end{aligned} \tag{15}$$

Now, one can construct an identification algorithm in three steps:

Step 1: From input-output sampled data one calculates the moments m_j using the relation (2)

Step 2: From relations (15) one computes the coefficients $C_k^{'}$. If there is k=n such as for $\forall k \geq n$, the system is compatible, then *n* is the denominator degree and the number of unknowns *m* is the nominator degree of the transfer function. This can be express in the following algebraic form:

$$D_k = \begin{vmatrix} C'_{k-m} & C'_{k-m+1} & .... & -C'_k \\ C'_{k-m+1} & C'_{k-m+2} & .... & -C'_{k-1} \\ ..... & ..... & & ..... \\ C'_k & C'_{k+1} & .... & -C'_{k+m} \end{vmatrix} = 0, \quad \forall k \geq n+1 \tag{16}$$

Step 3: One computes the coefficients of the transfer function with the relations (11), (12) (firstly the coefficients $b'_k, 1 \leq k \leq m$, then $a'_k, 1 \leq k \leq n$).

Example: One considers a system described by the following transfer function:

$$H_1(s) = \frac{1}{4s^2 + 8s + 1}$$

The experimental step response is presented in Fig. 1.

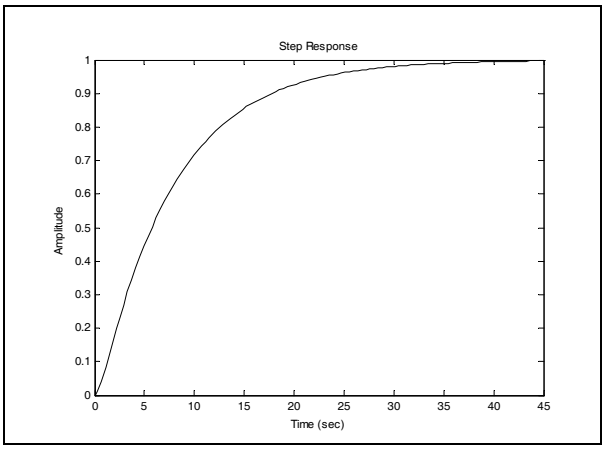


Fig. 1 Step response of H1(s)

It is easy to observe that $y(\infty) = 1$. The values of the moments m_j and of the coefficients $C_k^{'}$ are:

$m_1 = 7.9999$ $\quad C_1^{'} = 7.9999$

$m_2 = -60.0787$ $\quad C_2^{'} = 3.9899$

$m_3 = 448.5854$ $\quad C_3^{'} = -0.6795$

$m_4 = -3.3483e+003$ $\quad C_4^{'} = -0.6281$

Neglecting the small terms due to the approximation error, one obtain the following identified transfer function:

$$\hat{H}_1(s) = \frac{1}{3.98s^2 + 7.99s + 1}$$

III. Identification Based On Weighted Moments Method

As it was presented in previous sections, the *j*-power moment of an original function f(t):$[0, \infty) \rightarrow R$ is given by the following relation:

$$m_j = m_j(f) = \int_0^{\infty} \psi_j(t) f(t) dt \tag{17}$$

where the function

$$\psi_j(t) = \frac{(-t)^j}{j!} \tag{18}$$

represent a weight applied to the function f(t) in integration.

In identification, the moments m_j calculus using relation (17), when value of j is great, creates a series of problems like:

- the weighting function $\psi_j(t)$ unbound;
- the integration is effectuated on a finite time interval;
- the f(t) function represents the overall response of a system which contains both the free component due to the initial conditions and the forced component on which the moments method is based.

In Fig. 2 are presented time evolution for few functions $\psi_j(t)$. One observe that for small values of *t*, the f(t) signal in relation (17) is less amplified, but for big values of *t*, big weightings appear which can exceed the precision possibilities of the numerical calculus.

The convergence of the integrals from relation (17) is done by the condition $\sigma_0 < 0$. For slow processes the convergence radius is very small and the transient response is very long so, in the integrals evaluation components of the form $\psi_j(t) f(t)$, that is a product between a very large number $\psi_j(t)$, $j \geq 2$, and a finite value f(t).

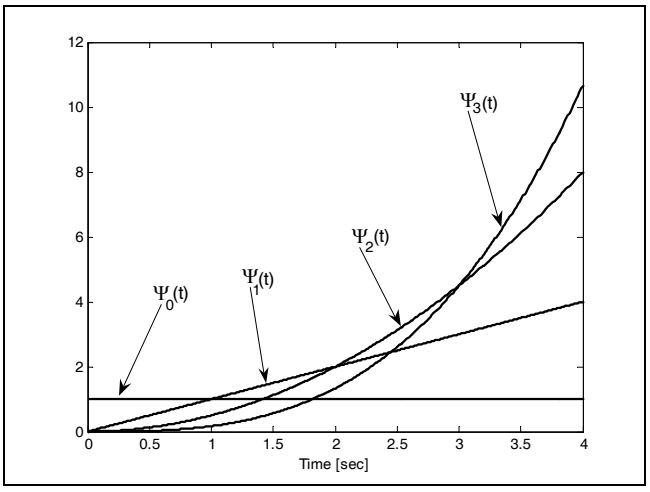


Fig. 2 Weighting functions for classical moments

On the other hand, if the integration is stopped to a finite moment t_2, the *j*-power moment of f(t) is approximated by:

$$m_j(t_0,t_2)=\int_{t_0}^{t_2}\psi_j(t)\,f(t)dt \tag{19}$$

and finite interval evaluation error:

$$\delta m_j(t_0,t_2)=m_j(0,\infty)-m_j(t_0,t_2) \tag{20}$$

is big even for a big value of t_2.

For these reasons one uses the weighted power moments that are defined by the following relation:

$$m_j^{\alpha}(f)=\int_0^{\infty}\frac{(-t)^j}{j!}w(t,\alpha)\,f(t))dt= \int_0^{\infty}\frac{(-t)^j}{j!}\,f^{\alpha}(t)dt \tag{21}$$

where $w(t,\alpha)$ is an original functions family defined par rapport with *t* in $[0,\infty)$ interval, with parameter $\alpha\geq 0$ and the convergence radius $\sigma_1=\sigma_0$.

The time function $f^{\alpha}(t):[0,\infty)\to R$, defined by:

$$f^{\alpha}(t)=f(t)w(t,\alpha) \tag{22}$$

is an original function also, with convergence radius:

$$\sigma_2=\sigma_1+\sigma_0 \tag{23}$$

The order *j* weighted moments appear as a real functional of function *f*:

$$m_j^{\alpha}(f)=\int_0^{\infty}\psi_j(t,\alpha)\,f(t))dt \tag{24}$$

with the following kernel:

$$\psi_j(t,\alpha)=\frac{(-t)^j}{j!}w(t,\alpha) \tag{25}$$

The Laplace transform $F^{\alpha}(s)$ of $f^{\alpha}(t)$ function admits a power series expansion:

$$F^{\alpha}(s)=\sum_{j=0}^{\infty}f_j^{\alpha}s^j,\quad |s|<R_2 \tag{26}$$

uniform convergence and holomorphic inside of a circle with the radius:

$$R_2=|\sigma_2|=|\sigma_0+\sigma_1(\alpha)| \tag{27}$$

If $\sigma_0\leq 0$ and $\sigma_1=\sigma_1(\alpha)\leq 0,\ \forall\alpha$, then:

$$R_2=|\sigma_0+\sigma_1|=|\sigma_0|+|\sigma_1(\alpha)|=R+|\sigma_1(\alpha)|>R \tag{28}$$

So, choosing a weighting $w(t,\alpha)$ with $\sigma_1=\sigma_1(\alpha)\leq 0$ one obtains a series with a larger convergence radius that has obvious advantages for numerical calculus.

The weighted moments contain the same information about *f* function as the classical moments but they have a better numerical robustness due to the condition imposed to convergence radius.

There are many possibilities for choosing the weighting function $w(t,\alpha)$ using efficiency and applicability criterions. In this work we consider:

$$w(t,\alpha)=e^{-\alpha t},\ \alpha\geq 0,\ \ \sigma_1=-\alpha\leq 0 \tag{29}$$

For this weighting function, one get:

$$R_2=|\sigma_0-\alpha| \tag{30}$$

$$m_j^{\alpha}(f)=\int_0^{\infty}\frac{(-t)^j}{j!}e^{-\alpha t}\,f(t))dt \tag{31}$$

$$\psi_j(t,\alpha)=\frac{(-t)^j}{j!}e^{-\alpha t} \tag{32}$$

$$F^{\alpha}(s)=F(s+\alpha),\ \mathrm{Re}(s)>\sigma_0-\alpha \tag{33}$$

Unlike of $\psi_j(t)$ presented in Fig. 2, the kernel $\Psi_j(t,\alpha)$ presented in Fig. 3 for some values of *j*, realize a weighting at the moment *t*:

$$t_j=\frac{j}{\alpha},\ j\geq 0 \tag{34}$$

with the amplitude

$$\psi_j(t_j,\alpha)=\frac{j!}{j!\alpha^j}e^{-j} \tag{35}$$

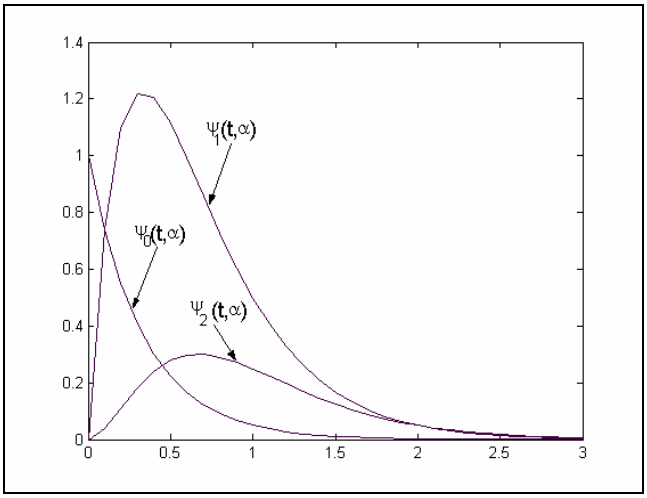


Fig. 3 The kernel $\Psi_j(t,\alpha)$

Depending on value of α parameter, the weighted moment give more importance to some time evolutions. For a big α, the attention is focus on initial evolution and for a small α, towards the final evolution. By choosing many values for α in identification procedures one get the essential aspects of time evolution of the signal *f(t)*.

The identification algorithm remains the same, the only modification is that we will process the weighting function $f^{\alpha}(t)$ instead of *f(t)*. Also, instead of the original transfer function:

$$H(s)=\frac{b_0+b_1s+\ldots+b_ms^m}{a_0+a_1s+\ldots+a_ns^n}=\frac{Y(s)}{U(s)} \tag{36}$$

characterized by n,m,b_k,a_k, we will obtain the following transfer function:

$$H^{\alpha}(s)=\frac{b_0^{\alpha}+b_1^{\alpha}s+...+b_m^{\alpha}s^m}{a_0^{\alpha}+a_1^{\alpha}s+...+a_n^{\alpha}s^n}=\frac{Y^{\alpha}(s)}{U^{\alpha}(s)} \quad (37)$$

characterized by $n, m, a_k^{\alpha}, b_k^{\alpha}$.

When $w(t,\alpha)=e^{-\alpha t}$, then

$$H^{\alpha}(s)=H(s+\alpha) \quad (38)$$

Example: One considers a system described by the following transfer function:

$$H_2(s)=\frac{s}{4s^2+2s+1}$$

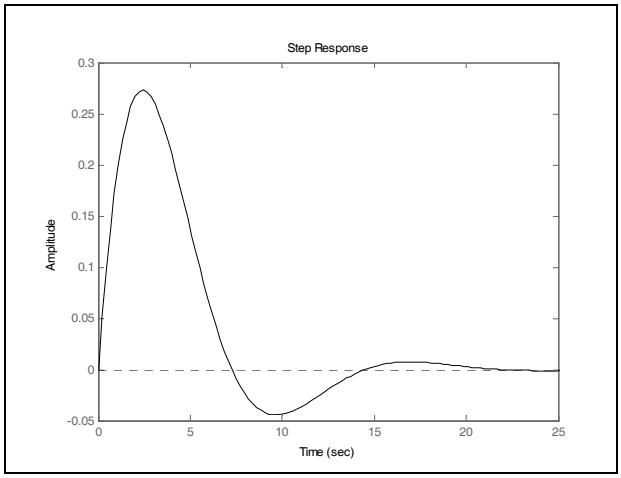


Fig. 4 Step response of $H_2(s)$

As it can be seen, $y(\infty)=0$. Using the identification algorithm one get: $K^{\alpha}=0.0483$, $a_1^{\alpha}=0.6147$, $a_2^{\alpha}=0.0992$.

One obtains the following estimated transfer function:

$$\hat{H}_2(s)=\frac{s}{4.083s^2+2.0515s+1}$$

IV. Experimental Results

To illustrate the performance of the proposed identification algorithms, one identify a real Quanser experiment using a DC servomotor with built in gearbox, is provided in this section. The "rotational series" that we have is the SRV-02ET (E-encoder, T-tachometer), and the DC servo is shown in the Figure 3. A high quality DC servo motor is mounted in a solid aluminum frame. The motor drives a built-in Swiss-made 14:1 gearbox whose output drives an external gear. The motor gear drives a gear attached to an independent output shaft that rotates in a precisely machined aluminum ball bearing block.

The output shaft is equipped with an encoder. This second gear on the output shaft drives an anti-backlash gear connected to a precision potentiometer. The potentiometer is used to measure the output angle. The external gear ratio can be changed from 1:1 to 5:1 using various gears. Two inertial loads are supplied with the system in order to examine the effect of changing inertia on closed loop performance. In the high gear ratio configuration, rotary motion modules attach to the output shaft using two 8-32 thumbscrews. The square frame allows for installations resulting in rotations about a vertical or a horizontal axis.

Fig. 5 Quanser SRV02ET experiment

The system is interfaced by means of a data acquisition card and driven by Matlab/Simulink based real time software. The model of this system can be found from physical considerations. One considers as input U of the system the voltage applied to the motor armature and as output Y of the system the angle of the output shaft. As is described in [12] the transfer function of the system has the following the form:

$$\frac{Y(s)}{U(s)}=\frac{K_p}{s(T_p s+1)} \quad (39)$$

Clearly, the open loop position response of the DC motor is unstable due to the pole at the origin. A proportional controller in closed loop is used in order to stabilize the system and to perform the identification experiments. The closed loop transfer function is:

$$\frac{Y(s)}{V(s)}=\frac{K_{cl}}{T_p s^2+s+K_{cl}} \quad (40)$$

or, equivalently:

$$\frac{Y(s)}{V(s)}=\frac{1}{a_2 s^2+a_1 s+1} \quad (41)$$

where:

$$a_1 = \frac{1}{K_{cl}},\ a_2 = \frac{T_p}{K_{cl}}, K_{cl} = K_p \cdot K_c \qquad (42)$$

K_c - controller gain.

In Fig. 6 the experimental step response of the closed-loop system with controller gain $K_c=0.2$ is presented.

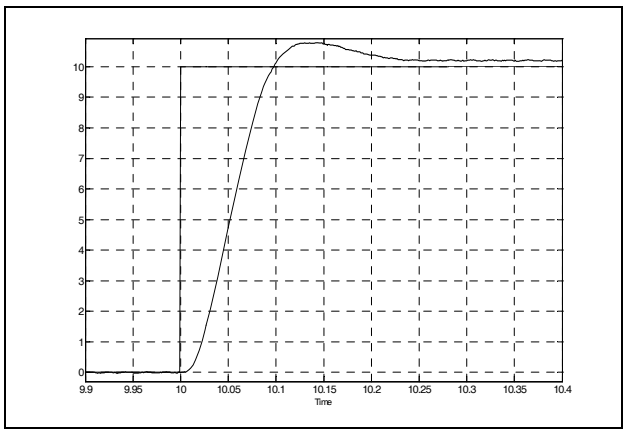


Fig. 6 Step response of the closed-loop system with controller gain K_c=0.2

Using the algorithm described the previous sections one can identify the a_1 and a_2 parameters and then one can deduce T_p and K_p. One obtains the following values:

$a_1 = 0.0412;$

$a_2 = 0.00095$

and from relations (42) one get:

$K_{cl} = 24.2584,$

$K_p = 121.2919,$

$T_p = 0.0232,$

So, the identified DC motor transfer function is:

$$\hat{H}(s) = \frac{121.2919}{s(0.0232s + 1)}$$

V. Conclusions

In this paper we presented a method for continuous-time invariant system identification using the weighted moments of the output of the system. The method has two main advantages related to most methods found in the specialty literature: first, there is no need of *a priori* information about the structure of the identified system and second, the continuous-time model is obtained direct from the input-output sampled data. The algorithm determines the order of the system from the relation between the power moments and the transfer function coefficients. The method was applied to identify the parameters of a real experimental platform consisting of a DC servomotor with built in gearbox. Because the open loop system is unstable, the closed loop identification was performed.

References

[1] N.I. Akhiezer, *The classical moment problem*, Hafner, New York, 1965.

[2] P. Eykhoff, *System Identification*, J. Wiley, London, 1974.

[3] R. Johansson, Identification of continuous-time models, *IEEE Transactions on Signal Processing*, 42(4), pp. 887-896, 1994.

[4] H.J. Landau, *Classical Background of the Moment Problem, Moments in Mathematics*, American Mathematical Society, Providence, Rhode Island, 1980.

[5] L. Ljung, *System Identification. Theory for the User*. Prentice-Hall, Englewood Cliffs, 1987.

[6] L. Kantorovitch, G. Akilov, *Analyse fonctionnelle*, Edition Mir, Moscou, 1981.

[7] C. Marin, System Identification by the method of weighted moments. *12th International Conference on Control Systems and Computer Science*, CSCS 12, Bucharest, 1999.

[8] T. Söderström, P. Stoica, *System Identification*, Prentice Hall, 1989.

[9] H. Unbehauen, G.P. Rao, Continuous-time approaches to system identification – a survey. *Automatica*, 26(1), pp. 23-35, 1990.

[10] H. Unbehauen, G.P. Rao, *Identification of continuous systems*, Amsterdam, North Holland, 1987.

[11] J. C. Willems, From time series to linear system. Part I: Finite dimensional linear time invariant systems. *Automatica*, 22, pp.561-580, 1986.

[12] *** Quanser Consulting Inc., WinCon 3.2, Real-Time Digital Signal Processing and Control under Windows 95/98 and Windows NT using Simulink and TCP/IP Technology, 1998.

Evolutionary Strategies for Sliding Mode Controller Parameters

Adriana Serbencu, Adrian Emanoil Serbencu and Daniela Cristina Cernega

***Abstract*— The performance of a Sliding Mode controller depend on some parameters. In literature, these parameters are usually experimental established. In this paper the Evolutionary Strategies (ESs) are investigated in order to obtain good values for the sliding mode control law parameters. The sliding mode control is used to solve the trajectory tracking problem for a wheeled robot, which is a nonlinear system. The influence of the obtained parameters on the control law is analyzed. The conclusions are based on the simulation results.**

I. Introduction

TO solve the trajectory tracking problem for a Wheeled Mobile Robot (WMR) it is used a nonlinear model [1]:

$$x^{(n)} = f(x,t) + b(x,t) \cdot u \tag{1}$$

where x is the state variable; $x^{(n)} = [x, \dot{x}, \ddot{x}, \ldots, x^{(n-1)}]$; $x^{(n)}$ is the n^{th}-order derivative of x; f is a nonlinear function; b is the gain and u is the control input.

The design of a variable structure control (VSC) [2] for a nonlinear system implies two steps: (1). "reaching mode" or nonsliding mode; (2). sliding mode.

For the reaching mode, the desired response usually is to reach the switching manifold s, described by:

$$s(x) = c^T \cdot x = 0 \tag{2}$$

in finite time with small overshoot with respect to the switching manifold.

The distance between the state trajectory and the switching manifold, s is stated as:

$$s(x,t) = \left(\frac{d}{dt} + \lambda\right)^{n-1} \cdot \tilde{x} = 0 \tag{3}$$

where $\tilde{x}$ is the tracking error and λ is a strictly positive constant which determines the closed-loop bandwidth. For example, if $n = 2$,

$$s = \dot{\tilde{x}} + \lambda \cdot \tilde{x} \tag{4}$$

Hence the corresponding switching manifold is:

$$s(t) = 0 \tag{5}$$

Manuscript received May 14, 2010. This work was supported by the Romanian High Education Scientific Research National Council (CNCSIS), under project PC ID-506.

All Authors are with the Control Systems and Industrial Informatics Department, Computer Science Faculty, "Dunarea de Jos" University from Galati, Romania (e-mail: Adriana.Serbencu@ugal.ro, Adrian.Serbencu @ugal.ro, Daniela.Cernega@ugal.ro)

For a system having m inputs, m switching functions are needed.

It is proved that the most important virtue of the VSC systems is robustness. Properly design of the switching functions for a VSC system ensures the asymptotic stability. A number of design criteria exist for this purpose [3],[4].

Sliding Mode is also known to possess merits such as the invariance to parametric uncertainties. Dynamic characteristics of the reaching mode are very important, and this type of control suffers from the chattering phenomenon which is due to high frequency switching over discontinuity of the control signal.

The parameters of the control laws have to be positive, and their values influence the reaching rate and the chattering. The values of these parameters are not specified in the literature. In this paper the optimal values for these parameters will be searched.

The process of optimization gained great importance in many real life engineering problems. Many optimization methods were proposed in literature. Evolution Strategy (ES) [5], [6] is a kind of an evolutionary algorithm that is known to be simple and has an excellent global search feature. The ES is presented in Section 2. Section 3 is dedicated to the Trajectory Tracking Problem for the Wheeled Mobile Robot. This problem is solved within the Sliding Mode approach and the result is the sliding-mode trajectory-tracking controller. The parameters p_i and q_i of the control law are not specified in the literature. In Section 4 Evolution strategy is used to determine the optimal values of the control law parameters in order to ensure maximum possible reaching rate of the switching manifold and minimum chattering and the results obtained are presented. Section 5 is dedicated to experimental results. Section 6 is dedicated to the conclusion and future work directions.

II. Evolutionary Strategies

In the case of evolutionary computation, there are four historical paradigms that have served as the basis for much of the activity of the field: genetic algorithms [7], genetic programming [8], evolutionary strategies [5], and evolutionary programming [9]. The basic differences between the paradigms lie in the nature of the representation schemes, the reproduction operators and selection methods. Evolution strategy (ES) was created in the early 1960s and developed further along the 1970s and later by [5], [6].

The basic version of evolution strategies uses just mutation and selection as search operators. The operators are applied in a loop. An iteration of the loop is called a

generation. The sequence of generations is continued a set number of times, or until a stop criterion is met, for example a threshold of distance (absolute or relative) between the two last positions, below which it is not necessary to go.

As far as real-valued search spaces are concerned, in the ES mutation is usually performed by adding a normally distributed random value to each solution component. The step size or mutation strength is often governed by adaptation.

The selection in evolution strategies is deterministic and based on the fitness rankings. The simplest ES operates on a population of size two: the current point (parent) and the result of its mutation. Only if the mutant's fitness is at least as good as the parent one, it becomes the parent of the next generation. Otherwise the mutant is disregarded. This is a (1 + 1)-ES. More generally, λ mutants can be generated and compete with the parent, called (1 + λ)-ES. In a (1, λ)-ES the best mutant becomes the parent of the next generation while the current parent is always disregarded. Contemporary versions of evolution strategy often use a population of μ parents and also recombination as an additional operator (called (μ/ρ+, λ)-ES) [10]. Using a population of solutions, make ESs less prone to convergence in local optima.

The evolution strategies implementation, that is used, operate on populations P of individuals a. An individual a_k with index k comprises not only the specific object parameter set (or vector) y_k and its objective function value $F_k := F(y_k)$, but also a set of endogenous and evolvable strategy parameters s_k, $a_k = (y_k , s_k, F(y_k))$.

The endogenous strategy parameters are used to control certain statistical properties of the mutation operator. Endogenous strategy parameters can evolve during the evolution process and are needed in self-adaptive ES [11].

Within one ES generation step, λ offspring individuals $\tilde{a}_l$ (note, the tilde is used to mark complete offspring) are generated from the set of μ parent individuals a_m. That is, the size λ of the offspring population P_o is usually not equal to the size μ of the parent population P_p. The strategy-specific parameters μ and λ as well as ρ (the mixing number, see below) are called "exogenous strategy parameters" which are kept constant during the evolution run.

The way the offspring population is generated is expressed by the $(\mu/\rho +, \lambda)$ notation. The ρ refers to the number of parents involved in the procreation of *one* offspring (mixing number).

Selection is the opposite of the variation operators, i.e. mutation and recombination. It gives to the evolution a direction. In the selection step, a new parental population at $(g + 1)$ is obtained by a deterministic process guaranteeing that only the μ best individuals a from the selection pool of generation (g) are transferred into $P_p(g+1)$.

There are two versions of selection technique, depending on whether or not the parental population at (g) is included in this process, i.e., *plus* selection, denoted by $(\mu + \lambda)$, and *comma* selection, denoted by (μ, λ), respectively. In the case of (μ, λ) selection, only the λ newly generated offspring individuals, i.e. the $P_o(g)$ population, define the selection pool. The plus selection takes the old parents into account. Plus selection is elitist because guarantees the survival of the best individual found so far an infinitely long time-span.

The mutation operator is the primary source of genetic variation. It is usually a basic variation operator in ES, which assure the search space exploration.

Considering the $\Re^N$ search space and given the standard deviation σ (mutation strength) as the only endogenous strategy parameter s, the mutation yields

$$\tilde{y} = y + z \tag{6}$$

with

$$z := \sigma(N_1(0, 1), \ldots, N_N(0, 1)) \tag{7}$$

where the $N_i(0,1)$ are independent random samples from the standard normal distribution.

There are two standard classes of recombination used in ES: "discrete recombination" and the "intermediate recombination".

Given a parental vector $a = (a_1, \ldots, a_D)$ (object or strategy parameter vector), the dominant ρ recombination produces a recombinant $r = (r_1, \ldots, r_D)$ by coordinate wise random selection from the ρ corresponding coordinate values of the parent family

$$(r)_k = (a_{mk})_k, \text{ with } m_k = \text{Random}\{1, . , \rho\} \tag{8}$$

The intermediate recombination simply calculates the center of mass (centroid) of the ρ parent vectors

$$(r)_k = \frac{1}{\rho}\sum_{m=1}^{\rho}(a_m)_k \tag{9}$$

The adaptation of strategy parameters control the statistical properties of the of the mutation operators. It is used the *1/5th* rule for controlling the mutation strength [12].

III. Trajectory Tracking Problem

In this paper the model used for the controlled robot is a 2-order MIMO (Multiply Input Multiply Output) nonlinear system that is "linear in control". The model used is:

$$\ddot{x} = f(x,\dot{x},t) + B(x,\dot{x},t)\cdot u \tag{10}$$

where $x = [x_1, x_2, \cdots, x_n]$, $x \in \Re^n$, f is a vector of nonlinear functions, $f \in \mathcal{L}_2^n$, B is a matrix of gains, $B \in \Re^{n\times n}$; $\det(B) \neq 0$; u is the control vector, $u \in \Re^n$. The control law is:

$$u = p(x,\dot{x},t) \tag{11}$$

For the 2nd-order MIMO nonlinear system having the model shown in (11) efficient sliding mode control can be

achieved via the following stages (see Fig. 1):

1st reaching phase motion; during this stage the trajectory is attracted towards the switching manifold (if the reaching condition is satisfied); characterized by

$$s_i \neq 0, \tilde{x}_i \neq 0, \dot{\tilde{x}}_i \neq 0 \tag{12}$$

2nd sliding mode motion; during this stage the trajectory stays on the switching manifold, i.e.

$$s_i = 0, \tilde{x}_i \neq 0, \dot{\tilde{x}}_i \neq 0 \tag{13}$$

3rd steady state; during this stage both the state variable and the state velocity will converge to the steady state value, therefore:

$$s_i = 0, \text{and} \begin{cases} \tilde{x}_i \to 0, \dot{\tilde{x}}_i \to 0 \\ \quad or \\ \tilde{x}_i = 0, \dot{\tilde{x}}_i = 0 \end{cases} \tag{14}$$

The reaching law is a differential equation which specifies the dynamics of a switching function *s(x)*. The differential equation of an asymptotically stable *s(x)*, is itself a reaching condition. In addition, by the choice of the parameters in the differential equation, the dynamic quality of the VSC system in the reaching mode can be controlled.

Gao and Hung [2] proposed a reaching law which directly specifies the dynamics of the switching surface by the differential equation

$$\dot{s} = -Q \cdot sgn(s) - P \cdot h(s) \tag{15}$$

where

$$Q = diag[q_1, q_2, \cdots, q_n],\ q_i > 0, i = 1, 2, \ldots, n$$

$$P = diag[p_1, p_2, \cdots, p_n],\ p_i > 0, i = 1, 2, \ldots, n$$

and

$$sgn(s) = [sgn(s_1), sgn(s_2), \cdots, sgn(s_n)^T]$$

$$h(s) = [h_1(s_1), h_2(s_2), \cdots, h_n(s_n)]^T$$

$$s_i \cdot h_i(s) > 0,\ \ h_i(0) = 0.$$

In this paper, a constant plus proportional rate reaching law proposed in [2] is investigated to control the mobile robot:

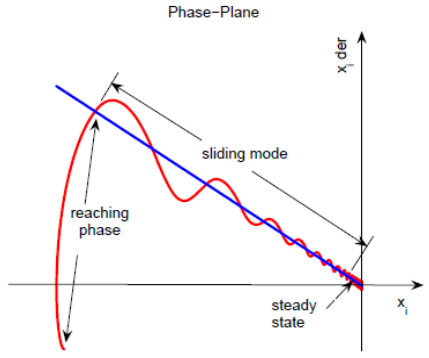


Fig. 1. Phase-Plane Diagram for the concept of 3-Stages approach.

$$\dot{s} = -Q \cdot sgn(s) - P \cdot s \tag{16}$$

Clearly, by using the proportional rate term $-P \cdot s$, the state is forced to approach the switching manifolds faster when s is large. It can be shown that the reaching time for x to move from an initial state x_0 to the switching manifold s_i is finite, and is given by:

$$T_i = \frac{1}{p_i} \cdot ln \frac{p_i \cdot |s_i| + q_i}{q_i} \tag{17}$$

The purpose of the trajectory tracking is to control the non-holonomic Wheeled Mobile Robot (WMR) to follow a desired trajectory, with a given orientation relatively to the path tangent, even when different disturbances exist. In the case of trajectory-tracking the path is to be followed under time constraints. The path has an associated velocity profile, with each point of the trajectory embedding spatiotemporal information that is to be satisfied by the WMR along the path. Trajectory tracking is formulated as having the WMR following a virtual target WMR which is assumed to move exactly along the path with specified velocity profile.

A. Kinematic model of a WMR

Fig. 2 shown a WMR with two diametrically opposed drive wheels (radius R) and free-wheeling castors (not considered in the kinematic models). *Pr* is the origin of the robot coordinates system. $2L$ is the length of the axis between the drive wheels. ω_R and ω_L are the angular velocities of the right and left wheels. Let the pose of the mobile robot be defined by the vector, $q_r = [x_r\ y_r\ \theta_r]^T$ where $[x_r\ y_r]^T$ denotes the robot position on the plane and θ_r the heading angle with respect to the x-axis. In addition, v_r denotes the linear velocity of the robot, and ω_r the angular velocity around the vertical axis. For a unicycle WMR rolling on a horizontal plane without slipping, the kinematic model can be expressed by:

$$\begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\theta}_r \end{bmatrix} = \begin{bmatrix} cos\theta_r & 0 \\ sin\theta_r & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_r \\ \omega_r \end{bmatrix} \tag{18}$$

which represents a nonlinear system.

Controllability of the system (18) is easily checked using the Lie algebra rank condition for nonlinear systems. However, the Taylor linearization of the system about the origin is not controllable, thus excluding the application of classical linear design approaches.

B. Trajectory-tracking

Without loss of generality, it can be assumed that the desired trajectory $q_d(t) = [x_d(t)\ \ y_d(t)\ \ \theta_d(t)]^T$ is generated by a virtual unicycle mobile robot (see Fig. 3). The kinematic relationship between the virtual configuration $q_d(t)$ and the corresponding desired velocity inputs

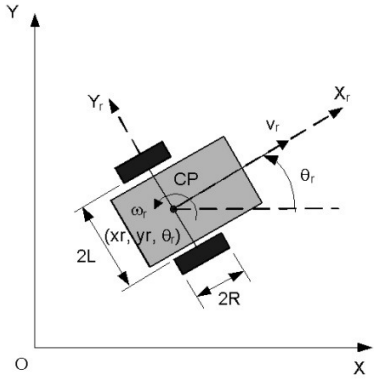


Fig. 2. WMR model and symbols.

$[v_d(t)\ \ \omega_d(t)]^T$ is analogue with (18):

$$\begin{bmatrix}\dot{x}_d\\ \dot{y}_d\\ \dot{\theta}_d\end{bmatrix}=\begin{bmatrix}cos\theta_d & 0\\ sin\theta_d & 0\\ 0 & 1\end{bmatrix}\cdot\begin{bmatrix}v_d\\ \omega_d\end{bmatrix} \tag{19}$$

When a real robot is controlled to move on a desired path it exhibits some tracking error. This tracking error, expressed in terms of the robot coordinate system, as shown in Fig. 3, is given by

$$\begin{bmatrix}x_e\\ y_e\\ \theta_e\end{bmatrix}=\begin{bmatrix}cos\theta_d & sin\theta_d & 0\\ -sin\theta_d & cos\theta_d & 0\\ 0 & 0 & 1\end{bmatrix}\cdot\begin{bmatrix}x_r\text{-}x_d\\ y_r\text{-}y_d\\ \theta_r\text{-}\theta_d\end{bmatrix} \tag{20}$$

Consequently one gets the error dynamics for trajectory tracking as

$$\begin{cases}\dot{x}_e=-v_d+v_r\cdot cos\theta_e+\omega_d\cdot y_e\\ \dot{y}_e=v_r\cdot sin\theta_e-\omega_d\cdot x_e\\ \dot{\theta}_e=\omega_r-\omega_d\end{cases} \tag{21}$$

C. Sliding-mode trajectory-tracking control

Uncertainties which exist in real mobile robot applications degrade the control performance significantly, and accordingly, need to be compensated. In this section, is proposed a sliding-mode trajectory-tracking (SM-TT) controller, in Cartesian space, where trajectory-tracking is achieved even in the presence of large initial pose errors and disturbances.

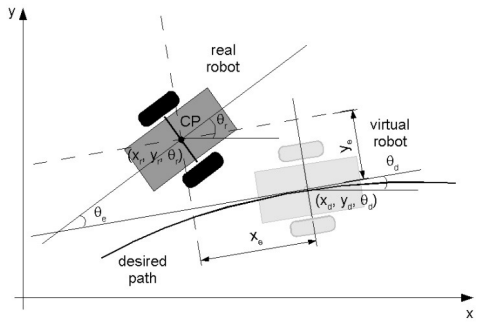


Fig. 3. Lateral, longitudinal and orientation errors (trajectory-tracking).

Let us define the sliding surface $s=[s_1\ \ s_2]^T$ as

$$\begin{aligned}s_1&=\dot{x}_e+k_1\cdot x_e\\ s_2&=\dot{y}_e+k_2\cdot y_e+k_0\cdot sgn(y_e)\cdot\theta_e\end{aligned} \tag{22}$$

where k_0, k_1, k_2 are positive constant parameters, x_e, y_e and θ_e are the trajectory-tracking errors defined in (20).

If s_1 converges to zero, trivially x_e converges to zero.

If s_2 converges to zero, in steady-state it becomes $\dot{y}_e=-k_e\cdot y_e-k_0\cdot sign(y_e)\cdot\theta_e$.

For $y_e<0\Rightarrow\dot{y}_e>0$ if only if $k_0<k_2\cdot|y_e|/|\theta_e|$.

For $y_e>0\Rightarrow\dot{y}_e<0$ if only if $k_0<k_2\cdot|y_e|/|\theta_e|$.

Finally, it can be known from s_2 that convergence of y_e and $\dot{y}_e$ leads to convergence of θ_e to zero. From the time derivative of (22) and using the reaching laws defined in (16), yields:

$$\begin{aligned}\dot{s}_1&=\ddot{x}_e+k_1\cdot\dot{x}_e=-q_1\cdot sgn(s_1)-p_1\cdot s_1\\ \dot{s}_2&=\ddot{y}_e+k_2\cdot\dot{y}_e+k_0\cdot sgn(y_e)\cdot\dot{\theta}_e=-q_2\cdot sgn(s_2)-p_2\cdot s_2\end{aligned} \tag{23}$$

From (20), (21) and (23), and after some mathematical manipulation, the output commands of the sliding-mode trajectory-tracking controller result:

$$\begin{aligned}\dot{v}_c&=\frac{-p_1\cdot s_1-q_1\cdot sgn(s_1)-k_1\cdot\dot{x}_e-y_e\cdot\dot{\omega}_d-\dot{y}_e\cdot\omega_d+v_r\cdot\dot{\theta}_e\cdot sin(\theta_e)+\dot{v}_d}{cos(\theta_e)}\\ \omega_c&=\frac{-p_2\cdot s_2-q_2\cdot sgn(s_2)-k_2\cdot\dot{y}_e+x_e\cdot\dot{\omega}_d+\dot{x}_e\cdot\omega_d-\dot{v}_r\cdot sin(\theta_e)}{v_r\cdot cos(\theta_e)+k_0\cdot sgn(y_e)}+\omega_d\end{aligned} \tag{24}$$

Let us define $V=s^T\cdot s/2$ as a Lyapunov function candidate, therefore its time derivative is

$$\begin{aligned}\dot{V}&=s_1\cdot\dot{s}_1+s_2\cdot\dot{s}_2=s_1\cdot(-q_1\cdot s_1-p_1\cdot sgn(s_1))+\\ &+s_2\cdot(-q_2\cdot s_2-p_2\cdot sgn(s_2))=s^T\cdot Q\cdot s-p_1\cdot|s_1|-p_2\cdot|s2|\end{aligned} \tag{25}$$

For $\dot{V}$ to be negative semi-definite, it is sufficient to choose q_i and p_i such that $q_i, p_i \geq 0$. But the optima values for $q_i, p_i \geq 0$ will be determined in the next section.

The *signum* functions in the control laws were replaced by *saturation* functions, to reduce the chattering [1],

$$sat\left(\frac{s}{\phi}\right)=\begin{cases}\dfrac{s}{\phi} & \text{IF}\left|\dfrac{s}{\phi}\right|\leq 1\\ sgn\left(\dfrac{s}{\phi}\right) & \text{IF}\left|\dfrac{s}{\phi}\right|>1\end{cases} \tag{26}$$

where constant factor ϕ defines the thickness of the boundary layer.

IV. Sliding Mode Controller Parameters Evaluated With ES

Solving the Trajectory Tracking Problem with a SMC, leads to the reaching laws (24). In literature, the parameters *q1, q2*, *p1* and *p2* are usual determined through experiments [13] and have great impact on the performance of the controller. *q1, q2* influence the rate at which the switching

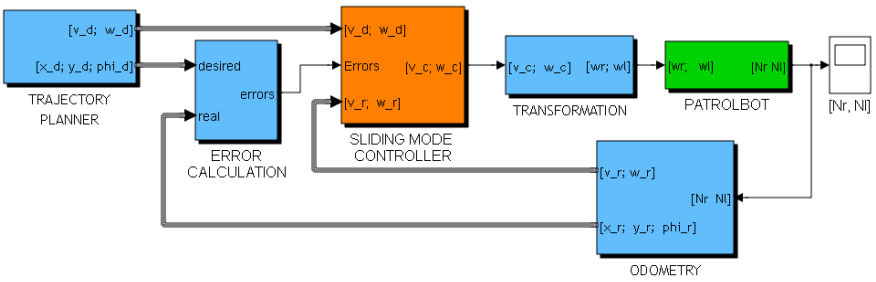


Fig. 4. Simulink schema of Sliding-Mode Trajectory-Tracking control for PatrolBot.

variable *s(x)* reach the switching manifold *S*. Parameters *p1, p2* force the state *x* to approach the switching manifolds faster when *s* is large. Choosing parameters through experiments only depends on experience or repeated debugging.

In this paper the parameters of the Sliding Mode Controller are selected using the Evolutionary Strategies (ES). The advantages of ES are: simplicity and an excellent global search feature, proven in other parameters training problems [14], [15], [16]. The optimization algorithm is working off-line. The results of the algorithm are the *P* and *Q* parameters of the reaching law implemented in the Sliding Mode Controller. The parameters found by ES can be used in real-time implementation of SM-TT controller on PatrolBot Robot.

The objective function used in ES takes into account booth the speed of reaching manifolds and the amplitude of the chattering. This is accomplished using the sum of root mean square of the two errors x_e and y_e (20). The evaluation of every set of parameters is achieved after running a numerical simulation of the SM-TT control structure implemented in a Matlab Simulink schema that contains the model of the robot, as shown in Fig. 4.

The horizon of simulation and initial conditions are chosen to allow a correct comparison between sets of parameters. The step of simulation is selected according to the one used to control the PatrolBot.

V. Experimental Results

Based on the above analysis, mathematical simulation software MATLAB was used to accomplish the experiment simulation study.

The mobile robot PatrolBot used in simulation is assumed to have the same structure as in Fig. 2. Parameter values of the PatrolBot are: mass of the robot body 46 [Kg], radius of the drive wheel 0.095 [m], and distance between wheels 0.48 [m]. The parameters of sliding modes were held constant during the experiments: $k_1 = 0.75$, $k_2 = 3.75$, and $k_0 = 2.5$; and the desired trajectory is given by $v_d = 0.5$ [m/s], $\omega_d = 0$ [rad/s].

The experiments were done on the robot with the initial error ($x_e = -0.5$ [m], $y_e = -0.5$ [m], $\theta_e = 0$ [deg]) and used the reaching law (16).

Settings, used in Matlab implementation of ES algorithm are: the size of the parent population P_p is $\mu = 20$; the size λ of the offspring population $\lambda=1$; the number of parents involved in the procreation of one offspring $\rho=4$; maximal number of generation 30. The criterion function used for solution evaluation are the sum of root mean square (RMS) of the two errors longitudinal - x_e and lateral - y_e. Root mean square error is an old, proven measure of control and quality. RMS can be expressed as

$$RMS = \sqrt{\left(\sum x^2(i)\right)/N}$$

Taking into account that parameters must be positive and value too large can causes chattering, the search interval for each SM parameters was selected to be [0.01 5].

In Fig. 5 the evolution of criterion function best value found by ES is presented. Note that a number of 30 generation are sufficient to find a good set of values of parameters.

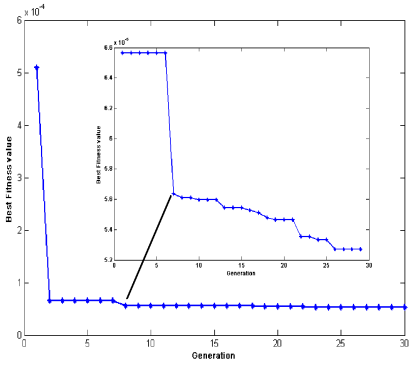


Fig. 5 The evolution of best value found by ES

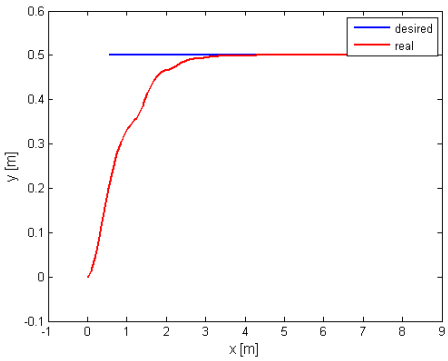


Fig. 6. Simulated trajectory with the parameters value found by ES. Experimental SM-TT control starting from an initial error state ($x_e(0) = -0.5$, $y_e(0) = -0.5$, $\theta_e(0) = 0$).

The parameters values for the considered PatrolBot, found by ES are q1=0.3312, q2=0.0340, p1=1.0599 and p2= 4.9295.

In Fig. 6 and 7, the simulation results for the case of optimised parameters are presented.

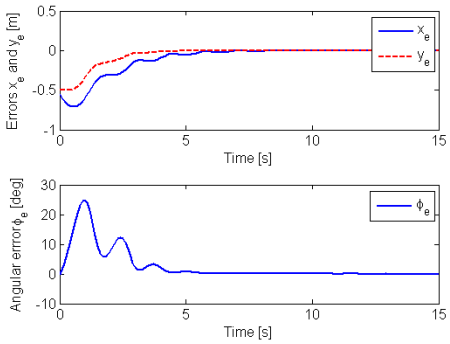


Fig. 7. Longitudinal, lateral and orientation errors for experimental SM-TT control.

In Fig. 8 the two sliding manifolds are represented. In Fig. 8 one can also see the value of the reaching time.

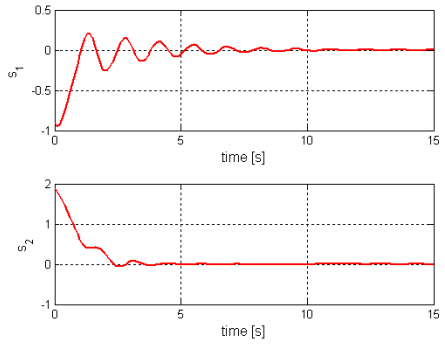


Fig. 8. Sliding surface for SM-TT controller.

Fig. 9 presents the response of the robot corresponding to the situation of a poor choice of the control law parameters without any optimization. It is easy to see the difference between the performances between Fig. 6 and 9.

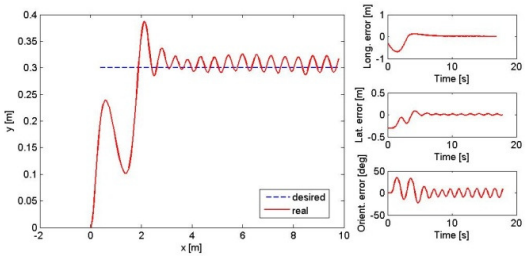


Fig. 9. An unfavourable case of experimental SM-TT control.

VI. Conclusion

The paper proposed an efficient method to determine the optimum set of parameters for the sliding mode controller. The Evolution Strategy proved to be adequate for this problem because it eliminates the need for repeated simulations in order to find a satisfactory set of parameters.

The tests have proven that this optimization technique is efficient for the problem to be solved. A very good solution without chattering was found in a quite acceptable time interval and number of iterations.

The search of the optimum values for the sliding mode trajectory tracking control laws parameters was done in order to use, in the future, such optimum parameters into a supervised control structure having the ability to switch between different controllers.

References

[1] J. Slotine, W. Li, *Applied Nonliner Control, Prentice Hall*, New Jersey 1991.

[2] W. Gao, J. Hung, *Variable structure control of nonlinear systems: A new approach.* IEEE Transactions on Industrial Electronics, vol.40, 1993, pp.45–55.

[3] V. Utkin, K. Young, *Methods for constructing discontinuity planes in multidimensional variable structure systems.* Automation and Remote Control 39, 1978, pp.1466–1470.

[4] C. Dorling, A. Zinober, *Two approaches to hyperplane design in multivariable variable structure control systems*. Int Journal Control, vol.44, 1986, pp.65–82.

[5] I. Rechenberg, *Evolution sstrategie: Optimierung Technischer Systeme nach Prinzipien der Biologischen Evolution.* Frommann-Holzboog, Stuttgart 1973.

[6] H. P. Schwefel, *Numerische Optimierung von Computer-Modellen* (PhD thesis) 1974. Reprinted by Birkhäuser 1977

[7] J. H. Holland, *Adaptation in natural and artificial systems*, Ann Arbor MI: The University of Michigan Press 1975.

[8] J. R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection.* Cambridge, MA:MIT Press 1992.

[9] L. J. Fogel, A. J. Owens, M. J. Walsh, *Artificial Intelligence through Simulated Evolution*, John Wiley, NY 1966.

[10] H. G. Beyer, H.-P. Schwefel, *Evolution strategies - A comprehensive introduction*. Natural Computing 1(1): 3-52 (2002)

[11] H. G. Beyer, K. Deb, *On self-adaptive features in real-parameter evolutionary algorithms.* IEEE Transactions on Evolutionary Computation 5(3), 2001, pp. 250–270.

[12] H. P. Schwefel, *Evolution and Optimum Seeking*. Wiley, New York 1995.

[13] R. Solea, D. C. Cernega, *Sliding Mode Control for Trajectory Tracking Problem* - Performance Evaluation. Artificial Neural Networks – ICANN 2009, Lecture Notes in Computer Science, vol. 5769, pp. 865-874.

[14] Ch. L. Lin, H. Y. Jan, *Application of evolution strategy in mixed H∞/H2 control for a linear brushless DC motor*. Advanced Intelligent Mechatronics, vol 1, 2003, pp. 1-6.

[15] M. W. Iruthayarajan, S. Baskar, *Evolutionary algorithms based design of multivariable PID controller*. Expert Syst. Appl. 36, 2009, pp.9159-9167.

[16] R. Richter, W. Hofmann, *Evolution Strategies Applied to Controls on a Two Axis Robot Source*. Lecture Notes In Computer Science, vol. 1226, pp: 434 – 443, Publisher Springer-Verlag London, UK, 1997.

Automatic cell nuclei detection in tissue sections from colorectal cancer

C. Smochina, V. Manta, G. Bises, and R. Rogojanu

***Abstract*—An automatic segmentation technique is proposed for detecting cell nuclei in images taken from tissues with colon carcinoma. The segmentation problems encountered in these images and solved by the proposed technique are related to the out-of-focus nuclei, non-uniform illumination on the background, the physical structure of cells in the tissue section and the clustered or touching cell nuclei. First, the region growing method is used to obtain accurate background detection. A new method is proposed to detect the separation areas between clustered/touching cell nuclei. Then, the watershed segmentation algorithm is applied twice on the image using as seeds the detected background and the separation lines. Since the watershed usually produces over-segmented results, an adequate merging region criterion is used to correctly identify the nuclei.**

I. Introduction

TISSUE development and disease-related processes such as tumorigenesis are determined in large part by communication between neighboring cells. Therefore, it is necessary to analyze and to monitor the evolution of each cell in its natural tissue environment [1]. Image processing has various application potentials in cytometry and histometry. Several segmentation methods have been developed due to the great diversity of biological samples where different artifacts and feature must be manually or automatically recognized [2], [3].

A major drawback of such segmentation techniques is the lack of an efficient method to segment whole cells or cell nuclei from microscopy images. There are two main types of nuclei according to their position: isolated and clustered. Segmentation of nuclei in grouped structures introduces additional problems compared with the isolated ones because these are in a close packing arrangement, often in contact with their neighbors [1].

Manuscript received May 15, 2010. This work was supported in part by the "BRAIN - An Investment in Intelligence" doctoral Scholarship Program, within the Technical University of Iasi, Romania and the FFG project N° 818094 "Automatisierte Zellerkennungs-Technologie für Forschung und Diagnose" at Medical University of Vienna, Austria.

Cristian Smochina is with the Department of Computer Science and Engineering,"Gheorghe Asachi" Technical University of Iasi, Romania (e-mail: smochina.cristian@yahoo.com).

Vasile Manta is with the Department of Computer Science and Engineering,"Gheorghe Asachi" Technical University of Iasi, Romania (e-mail: vmanta@cs.tuiasi.ro).

Giovanna Bises is with the Institute of Pathophysiology and Allergy Research, Medical University of Vienna, Austria (e-mail: giovanna.bises@meduniwien.ac.at).

Radu Rogojanu is with the Institute of Pathophysiology and Allergy Research, Medical University of Vienna, Austria (e-mail: radu@rogojanu.com).

II. Context of the Study

A. State of the Art

Automatic segmentation on optical microscopy images has been researched for many applications [4].

Watershed method is widely used to segment cell nuclei [5], [6]. In [5] the seeds are fully automatic obtained based on the original image and the gradient magnitude of the image. After applying watershed using the detected seeds a merging algorithm and a separation of the clusters of nuclei based on shape information is performed. The merging region criterion is based on the gradient magnitude along the boundary separating neighboring objects.

An important category of automatic segmentation uses the model-based approach [7], [8], [9]. In [8], the image primitives of convex objects such as step edges, roof edges and concave corners are used in dynamic programming and the minimization of a cost function to find out the cell nuclei.

The thresholding techniques, especially those based on the local automatic thresholding, succeed to proper delimit the non-uniform background. However, they fail in choosing a suitable threshold level for clustered objects or objects with un-sharp edges (out-of-focus cells/nuclei) [10]. Many methods also combine grey value thresholding with the morphological operators [8], [11].

Artificial neural networks (ANN) and machine learning (ML) are also used in microscopy image processing [12] but mainly in the field of single cell image classification [13].

B. Aim of the study

Due to the difficulty of this task many techniques deliberately included a small amount of proactive user interaction to guide the segmentation procedure. For instance, to guarantee correct segmentation of every cell, the algorithm presented in [1] required the user to mark two points per cell, one approximately in the center and the other on the border.

In our paper we focus on automatic cell nuclei detection in fluorescence images from colorectal cancer tissue sections (Fig. 1). These images contain cell nuclei labeled with DAPI, a fluorescent stain that binds strongly to DNA

[14] and were acquired using TissueFAXS slide scanner (TissueGnostics GmbH, Austria).

By analyzing the content of these images we can highlight the common problems that appear, problems pointed out also in other papers which present research in this field [15]:

--Non-uniform illumination which determines different grey-values for the background on different regions of the image, especially near the nuclei.

--Low contrast and weak boundaries on out-of-focus nuclei.

--The physical structure of the cells and the manner of sectioning determine a non-uniform distribution of material inside the nucleus, denser near the border that leads to lower intensities within the nuclei [16].

--Considerable variation of objects features like shape and/or size and/or orientation.

--Besides isolated nuclei, clustered or touching cell nuclei have very weak boundaries and often are not convex.

Considering these fundamental problems, none of the traditional segmentation methods used alone will produce a satisfactory result on these fluorescent images. Consequently the task of automatic segmentation on microscopy images is generally ranked as a demanding one [15]. The publications related to the image processing applied on microscopy images are wide-spread in literature, i.e. through the fields of microscopy, biomedical engineering, biomedical imaging, bioinformatics and pattern recognition.

III. A NEW AUTOMATIC SEGMENTATION TECHNIQUE

A. Background Extraction

Although many existing methods use a selected threshold value to separate the objects of interest and the background [7, 17], our method is based on similarity between neighboring pixels in the original intensity image.

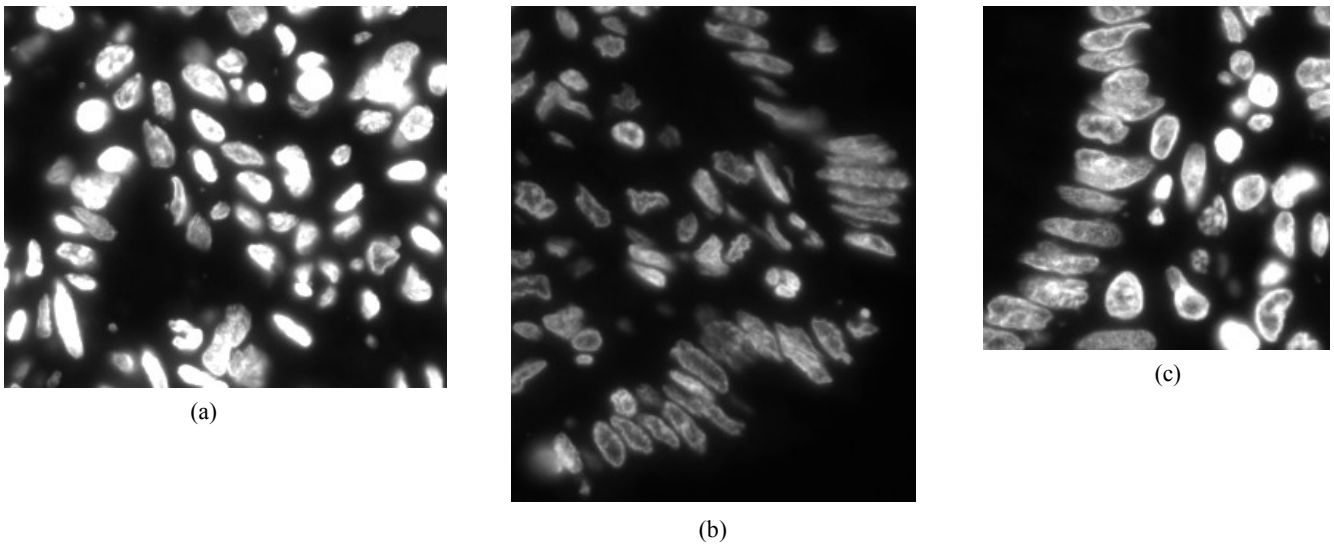
Non-uniform illumination determines different gray values for the background in different regions of the image, especially near the nuclei. Also the out-of-focus nuclei increase the amount of noise which makes a correct delimitation of the background more difficult. Because of non-uniform illumination, defining the border between cell nucleus and background by a single threshold value in the image intensity is very unlikely to provide acceptable results.

Many techniques use the region growing algorithm for objects delimitation but we will use this method to detect the background with a higher accuracy. The central idea of this method is to let the regions grow from predefined small regions, known as seeds. The regions grow by connecting non-processed neighboring pixels which satisfy an established criterion. Using this approach in cell nuclei segmentation causes many difficulties in constructing a seeding method that puts exactly one seed in each nucleus. A failure of this method will determine unsatisfactory results in case of clustered and/or touching nuclei or those who have high internal intensity variations.

The seeding method for region growing applied for background detection can use more seeds. For images with isolated nuclei a single seed is enough for background extraction. Beside isolated nuclei there are nuclei arranged in ring formations. For those situations a seed is also needed to delimit the interior of the ring (region $R51$ in Fig. 5(a)).

In our images the seeds are small regions since we use a global threshold on the intensity image. The value of this threshold, $thr_{\min}$, is a lower one so that no cell nuclei areas could be selected as seed (e.g. $thr_{\min} = 10$ for 8 bit images).

Starting from these areas, the neighboring pixels of the selected areas are considered background only if their grey-values are similar with values of those selected in a previous iteration. We compare the grey-value only with the mean value of the marked pixels as background from a close neighborhood of the candidate pixel. In our case, a neighborhood includes the pixel within a circle with the center given by the candidate pixel and radius *r*. The value thr_{grow} represents the maximum allowed difference between the candidate's grey value and the mean value of its neighborhood. The results of the background extraction are shown in Fig. 3(a), Fig. 4(a) and Fig. 5(a).



(a) (b) (c)

Fig. 1. Cell nuclei in tissue sections with colorectal cancer. Image b) is slightly under-exposed than a) and c).

B. Inter-nuclei separation line

The critical problem in some images is the touching nuclei. There are often situations in which the separation between two nuclei has a width of only one pixel. A specific method to detect these critical regions must be applied. Usually the separation between touching nuclei has a width of 3-4 pixels, but the neighboring pixels have higher values since they belong to the nucleus boundary. Based on this feature and considering the various orientations of the separation lines, a sequence of filters is applied on the image in order to highlight the background lines between clustered cell nuclei.

The pixels on these separation lines have the property that their grey-values are smaller than the grey-values of the neighboring pixels both from the right and the left side. We can imagine that these pixels are in a valley (the separation line) with low intensities and the neighboring pixels with higher values form two hills one on the left and another on the right side. The profile of the section has a V shape, thus the name of the proposed method.

For each candidate pixel, we consider all pixels in a 2D kernel/rectangle K with the center given by the candidate pixel. We mark a pixel as belonging to a separation line if the majority of the neighboring pixels from a specific orientation can be considered as belonging to a separation line.

Each pixel on the center column of the kernel is compared to the results of the convolutions between the left and the right line and a 1D Gaussian kernel. If the pixel's grey-value plus a value $nMinOffset$ is smaller than both convolutions results then it is marked. After checking each pixel from the center column, the following rule is applied: if the number of marked pixels from the center column is bigger than a value $nMinVpixels$, the center of the kernel is marked as belonging to a separation line since it is surrounded by possible pixels belonging to separation lines.

As we do not know the orientation of the possible separation line which contains the candidate pixel, the process described above will be repeated by rotating the kernel K with 45, 90 and 135 degrees. In this way the pixels from most separation lines will be found.

For a better understanding let's consider a possible kernel K with the following sizes $nKernelWidth = 9$, $nKernelHeight = 3$ and a 1D Gaussian kernel G with the standard deviation 1, G=[0.1345 0.3655 0.3655 0.1345]. The number of values from G is $(nKernelWidth - 1)/2$, where $nKernelWidth$ is an odd number. The kernels used to process each pixel from the center column are shown in Fig. 2. The symbol $\otimes$ is used to represent the convolution operation between an image and a kernel.

The first intermediate results **lineV0** (Fig. 2 (d)) contains for each pixel a value. This value represents the number of possible pixels belonging to a separation line with orientation 0 from the center column of the kernel K; the center of the kernel K is given by the current pixel. A pixel is marked as belonging to a separation line if this value is higher than $nMinVpixels$. This process is repeated for all rotations of the kernels from Fig. 2 with 45, 90 and 135 degrees. The obtained results **lineV45, lineV90** and **lineV135** will contain the possible pixels situated on the separation lines with orientation of 45, 90 and 135 degrees.

Each intermediate result is validated if the number of pixels from the center column is bigger than the value $nMinVpixels$. The final result of the method is given by an 'OR' operation between all intermediate results:

$$\text{lineV} = (lineV0 > nMinVpixels) \mid (lineV45 > nMinVpixels) \mid (lineV90 > nMinVpixels) \mid (lineV135 > nMinVpixels).$$

K_line1_L

0.1345	**0.3655**	**0.3655**	**0.1345**	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

K_line1_C

0	0	0	0	**1**	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

K_line1_R

0	0	0	0	0	**0.1345**	**0.3655**	**0.3655**	**0.1345**
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

(a)

K_line2_L

0	0	0	0	0	0	0	0	0
0.1345	**0.3655**	**0.3655**	**0.1345**	0	0	0	0	0
0	0	0	0	0	0	0	0	0

K_line2_C

0	0	0	0	0	0	0	0	0
0	0	0	0	**1**	0	0	0	0
0	0	0	0	0	0	0	0	0

K_line2_R

0	0	0	0	0	0	0	0	0
0	0	0	0	0	**0.1345**	**0.3655**	**0.3655**	**0.1345**
0	0	0	0	0	0	0	0	0

(b)

K_line3_L

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0.1345	**0.3655**	**0.3655**	**0.1345**	0	0	0	0	0

K_line3_C

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	**1**	0	0	0	0

K_line3_R

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	**0.1345**	**0.3655**	**0.3655**	**0.1345**

(c)

Fig. 2. a, b, c) Kernels used to detect the points in separation regions for each line of the kernel K; d) the operations applied on the intensity image **I** to determine the possible pixels on a separation line with orientation 0.

lineV0 = $[(\mathbf{I} \otimes \mathbf{K\_line1\_C} + nMinOffset) < (\mathbf{I} \otimes \mathbf{K\_line1\_L})$ AND $(\mathbf{I} \otimes \mathbf{K\_line1\_C} + nMinOffset) < (\mathbf{I} \otimes \mathbf{K\_line1\_R})] +$
$[(\mathbf{I} \otimes \mathbf{K\_line2\_C} + nMinOffset) < (\mathbf{I} \otimes \mathbf{K\_line2\_L})$ AND $(\mathbf{I} \otimes \mathbf{K\_line2\_C} + nMinOffset) < (\mathbf{I} \otimes \mathbf{K\_line2\_R})] +$
$[(\mathbf{I} \otimes \mathbf{K\_line3\_C} + nMinOffset) < (\mathbf{I} \otimes \mathbf{K\_line3\_L})$ AND $(\mathbf{I} \otimes \mathbf{K\_line3\_C} + nMinOffset) < (\mathbf{I} \otimes \mathbf{K\_line3\_R})]$.

(d)

This approach will produce false-positive (FP) [18] results in case of the nuclei with low intensity within the nucleus. These regions with lower intensities will be considered by the 'V-detection' method as separation lines since they are expected to appear between nuclei.

The FP results will be ignored by considering their position. The true positive results founded between nuclei have the property that they 'touch' the background (regions drawn with green in Fig. 3(b), 4(b) and 5(b)). The wrong separation lines founded within nuclei will be ignored since they cannot 'touch' the background because of the nuclei boundaries (regions drawn with red in Fig. 3(b), 4(b) and 5(b)).

C. *Region growing applied on selected separation lines*

In case of clustered nuclei, the 'V-detection' will highlight the separation lines. There are situations in which a ring or a group of touched nuclei surrounds a small region of background not founded by the background extraction approach.

We apply the region growing algorithm again using as seeds the selected separation lines but with a smaller value for parameter thr_{grow}. After this growing the selected separation lines became more obviously and the small areas between grouped nuclei are filled (marked as separation line). The growth of the founded separation lines are depicted with blue in Fig. 3(b), 4(b) and 5(b).

D. *Watershed with seeds*

A popular region growing method widely used in many areas of image segmentation and analysis is the watershed algorithm [19, 20].

Compared with region growing method the watershed approach works per intensity layer instead of per neighbor layer. If watershed segmentation is applied directly to the image, it will most likely result in over-segmentation [5].

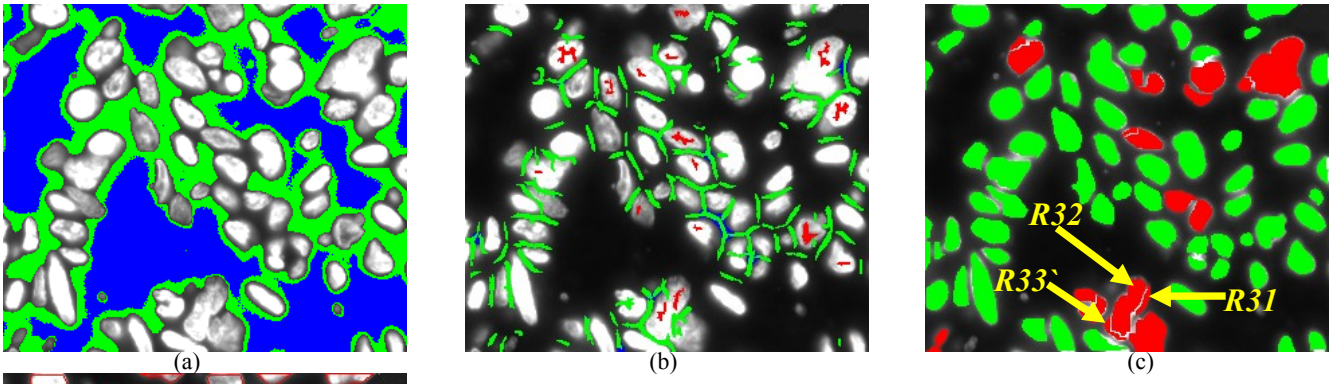
To detect the nuclei boundaries, the watershed algorithm is applied in two phases. The structure of the nuclei with high intensity variations within the nuclei causes difficulties in boundaries detection. Because of the internal DNA distribution, some nuclei have the ring-shape pattern, with lower intensities in the center of the nucleus The first application of the watershed algorithm tries to detect these regions.

The extended minima transform [21] is applied in order to simplify the intensity image by connecting regions of pixels with the intensity smaller than a value *hmin*, whose external boundary pixels all have a value greater than *hmin*.

The watershed algorithm is applied considering as seeds the detected background, the reached separation lines and the results offered by the extended minima transform. In this case, the watershed basins will grow in regions within the nuclei and they will fill each area with low intensities founded in the interior of the nuclei. For the nuclei without low intensities in interior there will be no catchment basins created. Imposing as seeds the local minima makes the watershed lines to appear within the nucleus (for the nuclei with low intensities in interior) but do not coincide with nucleus boundaries.

Another category of nuclei are those with higher intensity values in the interior of the nucleus than on the boundaries, cone-shaped nuclei. To detect these cell nuclei the watershed algorithm should be applied on the reversed intensity image. If the images contain only nuclei with higher intensity values in the interior, by reversing the intensity image these nuclei will form the proper basins in which the water can accumulate.

Since the images contain both types of nuclei (with higher and lower intensities within nucleus), the result of the first watershed run is used on the reversed intensity image. By reversing the nuclei with lower intensities in interior, the nucleus boundaries will surround regions with high intensities. The watershed lines founded on the first watershed run delimit these regions with high intensities



(a) (b) (c) (d)

Fig. 3. Intermediate results of applying the segmentation technique on the image Fig. 1(a).

a) the result of the initial global thresholding with the value $thr_{\min}$ is depicted in blue; the growth of these regions using region growing is drawn with green.
b) The regions with red represent the results of the "V-detection" method (FP results) ignored because they are within the nuclei. The selected separation lines are depicted with green and the growth of these lines with blue.
c) The result of the watershed algorithm: the isolated regions are depicted with green and the regions to be verified for merging with red.
d) The boundaries of the final detected regions after merging.

(from the reversed intensity image). If we impose as seeds the regions founded by the first watershed run in the reversed image we will eliminate the problems related to the nuclei with lower intensities within nucleus. Also the background and the reached separation lines are used as seeds. The second watershed run detects the nuclei in the reversed intensity image. The obtained regions are shown in Fig. 3(c), 4(c) and 5(c).

E. Merging region

Even if the seeded watershed is used, the over-segmented results are likely to occur. We compute a region adjacencies graph of the labeled image after applying the watershed. The relationship between regions is given by the watershed lines. Two regions are considered as neighbor if they are separated by a watershed line. In Fig. 3(c) the isolated nuclei are depicted with green and the regions to be verified for merging with red.

We used a simple shape descriptor which has been widely used in pattern recognition tasks: shape compactness. The shape compactness of a region is given by $C = 4\pi A / P^2$ where P is the region perimeter and A is the region area; $C = 1$ for a circle and decreases according to the shape changes [22], [23].

Two adjacent regions are merged if the compactness of the unified region is bigger than the minimum compactness of these two regions. The condition $C(R_{12}) > \min(C(R_1), C(R_2))$ is evaluated. R_1 and R_2 are the neighboring regions and R_{12} is the union of these two. For instance, the regions $R31$, $R32$ and $R33$ (Fig. 3(c)) or regions $R52$ and $R53$ (Fig. 5(c)) have been merged.

IV. Results

We tested the proposed segmentation technique on different datasets of images of cell nuclei labeled with DAPI [14] and compared against ground-truth segmentations. In all the datasets the algorithm gave comparable results even if there were some significant differences between the datasets. In some images the nuclei were distributed sparsely throughout the image while in some others they were grouped together. The results confirmed that the proposed method could efficiently segment isolated and touching cell nuclei. Fig. 1 shows some of the representative images and figures 3(d), 4(d) and 5(d) present the segmentation results as a red border around the nuclei.

The selected threshold for background extraction $thr_{\min}$ and the value thr_{grow} used in region growing algorithm in case of Fig. 1(a) could be higher because the image is slightly over-exposed. Bigger values for these parameters determine quick and accurate background detection. In case of Fig. 1(b) these highly values will produce FP results by marking as background also parts of cell nuclei with low intensities. The applied region growing method cannot mark as background the separation region between cell nuclei; it can only delimit the clustered or touching nuclei (for instance region $R41$ from Fig. 4(b)). The proposed method succeeds to detect the separation regions in these situations and can also be applied on images from other application areas.

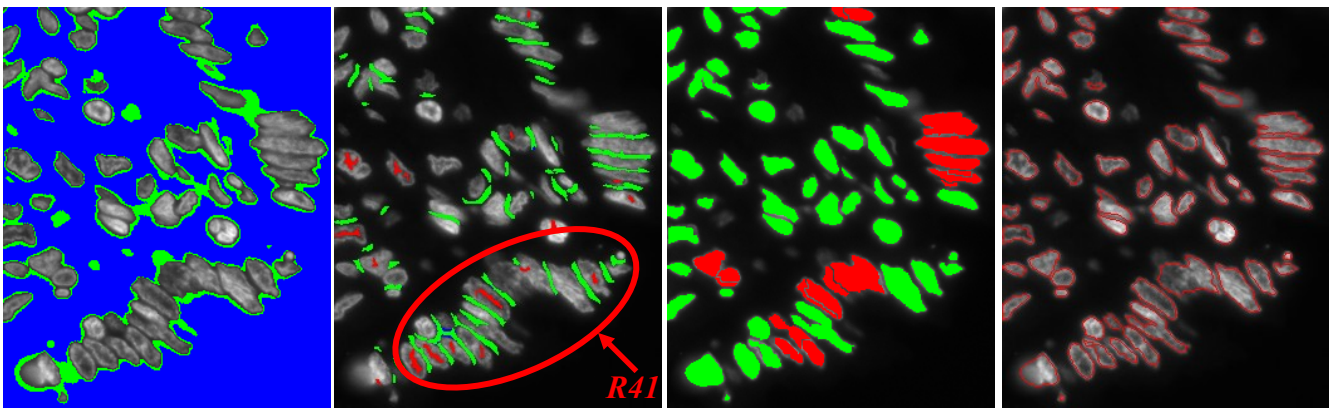
V. Conclusions

In this paper we approach the problem of segmentation of cell nuclei in isolated and clustered configurations. The automatic proposed technique is based on methods which detect first the background and the separation lines between nuclei and then nuclei boundaries.

We propose a method to detect the critical regions between clustered and touching nuclei based on gray-level criterion. Usually this method produces many FP results but they are filtered judging by their link with the detected background.

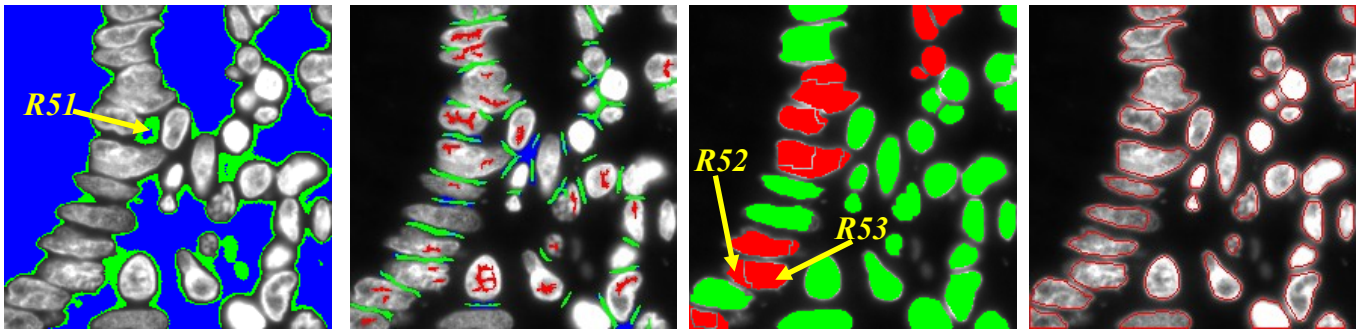
The watershed algorithm is used in two phases; first it is used to 'fill' the regions with low intensities within the nuclei and second it detects the nuclei boundaries applied on the reverse image. In our case the compactness descriptor used as merging criterion offers good results but many other more complex criteria can be used.

Future work will be devoted to the extraction of feature shape descriptors which will be used in creating model templates. These templates will facilitate the detection of overlapping nuclei.



(a) (b) (c) (d)

Fig. 4. The intermediate and the final results of applying the segmentation technique on the image from Fig. 1(b).



(a) (b) (c) (d)

Fig. 5. The intermediate and the final results of applying the segmentation technique on the image from Fig. 1(c).

REFERENCES

[1] D. Baggett, M. Nakaya, M. McAuliffe, T.P. Yamaguchi, and S. Lockett, "Whole cell segmentation in solid tissue sections," *International Society for Advancement of Cytometry*, vol. 67(2)-A, pp. 137–143, 2005.

[2] L. V. Guimarães, A. A. Suzim, and J. Maeda, "A New Automatic Circular Decomposition Algorithm Applied to Blood Cell Images," *Proceedings of IEEE Int. Conf. on Bio-Informatics & Biomedical* (*BIBE'00*), Washington, pp. 277-280, 2000.

[3] T. Figueiro, N. Schuch, L.V. Guimarães, and A. Susin, "A Method for Automatic Detection of Blood Cells on Images using Image Correlation and Connected Components," *Proc. WICCGPI*, São Carlos, Brazil, October 12-15, 2003.

[4] T. Figueiro, N. Schuch, A. Soares, L. Guimaraes, and A. Susin, "Automatic detection of blood cells on color images using image matching and flood map," *Brazilian Symposium on Computer Graphics and Image Processing*, SIBGRAPI, Natal, Brasil, 2005.

[5] C. Wahlby, I.M. Sintorn, F. Erlandsson, G. Borgefors, and E. Bengtsson, "Combining intensity, edge and shape information for 2D and 3D segmentation of cell nuclei in tissue sections," *Journal of Microscopy,* vol. 215(Pt 1), pp. 67–76, July, 2004.

[6] N. Malpica, C.O. de Solórzano, J.J. Vaquero, A. Santos, I. Vallcorba, J.M. García-Sagredo, and F. del Pozo, "Applying watershed algorithms to the segmentation of clustered nuclei," *Cytometry,* vol. 28(4), pp. 289-297, Aug, 1997.

[7] G. Cong and B. Parvin, "Model based segmentation of nuclei," *Pattern Recognition*, 2000, vol. 33(8), pp. 1383-1393.

[8] H. Talbot and I. Villalobos, "Binary image segmentation using weighted skeletons," *Proc. SPIE 1992,* vol. 1769, pp. 393-403.

[9] F. Cloppet and A. Boucher, "Segmentation of complex nucleus configurations in biological images," *Pattern Recognition Letters,* 2010, vol. 3(8), pp. 755–761.

[10] C.O. de Solorzano, G. Rodriguez, A. Jones, D. Pinkel, J.W. Gray, D. Sudar, and S. Lockett, "Segmentation of confocal microscope images of cell nuclei in thick tissue sections," *Journal of Microscopy,* 1999, vol. 193(Pt 3), pp 212–226.

[11] A. Nedzved, S. Ablameyko, and I. Pitas, "Morphological segmentation of histology cell images," *The 15th International Conference on Pattern Recognition (ICPR'00),* 2000, Barcelona, Spain.

[12] I. Kononenko, "Machine Learning for Medical Diagnosis: History, State of the Art and Perspective," *Artificial Intelligence in Medicine,* vol. 23(1), pp 89-109, August, 2001.

[13] M.V. Boland and R.F. Murphy, "A neural network classifier capable of recognizing the patterns of all major subcellular structures in fluorescence microscope images of HeLa cells," *Bioinformatics,* 2001, vol. 17(12), pp 1213-23.

[14] K. Morikawa and J. Yanagida, "Visualization of individual DNA molecules in solution by light microscopy: DAPI staining method," *Japanese Biochemical Society,* 1981, vol. 89(2), pp. 693-696, Tokyo.

[15] T.W. Nattkemper, "Automatic segmentation of digital micrographs: A survey," *Journal Studies in health technology and informatics*, 2005, vol. 107(2), pp 847-852.

[16] A. Todman and E. Claridge, "IntroCell segmentation in histological images of striated muscle tissueduction – a perceptual grouping approach," *Proceedings of Medical Image Understanding and Analysis* , Oxford, 7-8 July, 1997.

[17] V. Laurain, H. Ramoser, C. Nowak, G. Steiner, and R. Ecker, "Fast automatic segmentation of nuclei in microscopy images of tissue sections," *Conf Proc IEEE Eng Med Biol Soc,* 2005, vol(4), pp. 3367-3370.

[18] C. Smochina, V. Manta and R. Rogojanu, "New discrepancy measure for segmentation algorithm evaluation," *Proceeding (679) Computer Graphics and Imaging - 2010*, February 17 – 19, 2010, Innsbruck, Austria.

[19] E. Monsen, T. Randen, L. Sønneland, and J.E. Odegard, "Geological Model Building: A Hierarchical Segmentation Approach," *Mathematics in industry*, vol. 7, pp. 213-246, 2005.

[20] S. Beucher and C. Lantuejoul, "On the change of space in image analysis", *Proceeding of the ICSS*, Salzbourg, 1979.

[21] P. Soille, "Morphological Image Analysis Principles & Applications," *Springer*, 2nd ed. 2001, XVI, 391 p. 260 illus.

[22] R.S. Montero, "State of the Art of Compactness and Circularity Measures," *International Mathematical Forum*, vol. 4, no. 25-28, pp. 1305 – 1335, 2009.

[23] M.S. Nixon and A.S. Aguado, "Feature Extraction & Image Processing," *Elsevier Ltd.*, 2002.

On low complexity model predictive control of DC/DC converters

V. Spinu, M. Lazar, *Member, IEEE,* and P.P.J. van den Bosch, *Member, IEEE*

***Abstract*— The application field of DC/DC converters is very challenging for high performance controller design. The nonlinear system behavior, very fast dynamics and hard constraints on inputs and states are the main concerns for controller synthesis. Recently, an increased attention was given to implementation of model predictive control techniques for this type of applications. As part of this trend, a new methodology for low complexity predictive controller synthesis is proposed in the current paper. The proposed controller has a low computational load and decouples stability requirements from the performance objective. Flexible control Lyapunov functions are employed as a non-conservative tool to attain stability. The low complexity of the algorithm is achieved by employing the infinity norm to define both the Lyapunov and the cost function. This makes it possible to implement the predictive control algorithm as a single linear program. The control algorithm is tested in simulation of a two-inputs buck-boost DC/DC converter and a detailed analysis of the results is provided.**

***Index Terms*— Model predictive control, Lyapunov methods, DC/DC converters**

I. Introduction

Power conversion technology plays a major role in real-world applications. Increased efficiency, high performance and low weight are the key factors in new product design. DC/DC converters may range from small single chip, low cost solutions for consumer products to high performance and high power converters for the automotive, aerospace and energy industry. A comprehensive description of pulse-width modulation (PWM) control of power converters can be found in [1].

Buck-boost DC/DC converters are switching devices which have strong nonlinear dynamics and they are subject to hard constraints on inputs and states. A very fast switching frequency and small sampling time (ranging from μs to ns) pose a serious challenge to control algorithm design and implementation. That is why complex high-performance control strategies such as *model predictive control* (MPC) are rarely used in power electronics. Nevertheless, increased performance of the digital signal controllers (DSCs) [2] and recent advances in control theory allows the MPC methodology to be seriously considered for this type of application, see, for example, [3]–[11].

Within MPC algorithms that have been applied in the power electronics field, there are two approaches that appear more frequently in literature: *explicit* MPC (e-MPC), see, e.g., [7], [8] and *real-time optimization* MPC (rto-MPC), see, e.g, [5], [6], [10], [11]. With some abuse of terms, rto-MPC is meant to include all MPC solutions that require solving an optimization problem online, at each sampling instant, without necessarily having a real-time guarantee of feasibility. In contrast, e-MPC relies on the off-line calculation of an explicit, piecewise affine (PWA) state-feedback control law via multi-parametric programming, see, e.g., [12]. This explicit solution is defined over a partition of a bounded subset of the state-space and its implementation amounts to an online search through a look-up table. The most important aspect of e-MPC is that the explicit solution was shown to be equivalent to the corresponding rto-MPC solution [12] for linear and PWA models. An issue of e-MPC for DC/DC converters is that the memory requirements and the search time of such controllers grow exponentially with the length of the prediction and control horizon and the size of the set of admissible exogenous inputs, such as the load current and the supply voltage. The applicability of e-MPC schemes is restricted to linear and PWA models and a relatively small (control) horizon length. For e-MPC, stability of the closed-loop system can be checked a posteriori, by searching for a candidate Lyapunov function for the explicit PWA closed-loop system. If this search fails, there is no systematic procedure for modifying the original MPC scheme such that the resulting e-MPC closed-loop system is stable.

On the other hand, the rto-MPC algorithms have lower memory requirements and classical MPC stabilization theory offers several solutions for ensuring stability [13]. Nevertheless, the required calculation time for solving the optimization problem is significantly higher compared to e-MPC solutions, especially when a nonlinear model is involved. Thus, rto-MPC with prediction horizon larger than one is hardly implementable in the power electronics field. Moreover, most classical stabilization methods that rely on a monotone decrease of a candidate control Lyapunov function (CLF) are overly conservative for horizon-1 rto-MPC. In particular, the standard CLF approach to stabilization is over-conservative for power converters, as the startup evolution of the inductor current clearly violates the Lyapunov sufficient conditions [14]. At first, the inductor current should rise to the maximum allowed value and only when the output voltage is near the reference it should drop gradually to the stationary value, which can be several orders of magnitude smaller than the maximally allowed value.

This paper investigates the applicability to the control of DC/DC converters of a recently developed non-conservative stabilization method for horizon-1 rto-MPC [15]. The non-conservative stabilization mechanism makes use of *flexible* control Lyapunov functions [15]. In contrast to classical CLFs, a flexible CLF allows a non-monotone evolution of

V. Spinu, M. Lazar and P.P.J van den Bosch are with the Department of Electrical Engineering, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands, E-mail: v.spinu@tue.nl, m.lazar@tue.nl, p.p.j.v.d.bosch@tue.nl.

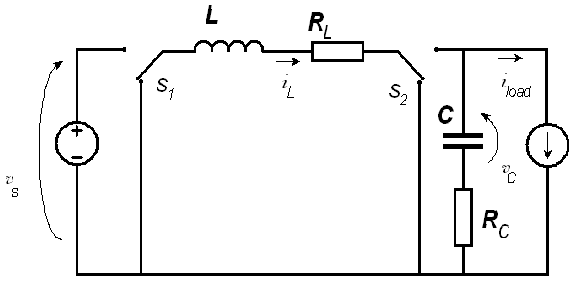


Fig. 1. A schematic representation of the non-inverting buck-boost converter.

the candidate CLF along the closed-loop trajectory. In [15] it was shown that this relaxation can yield an extension of the feasible set of control inputs and an improved overall performance. Another attractive feature of this approach is given by its low computational complexity, i.e., for infinity norms as candidate CLFs, the corresponding horizon-1 rto-MPC scheme requires solving a single linear program (LP) online, while employing a classical averaged nonlinear model [16]. The developed rto-MPC scheme is tested in simulation by controlling a two-inputs buck-boost DC/DC converter.

II. Non-inverting buck-boost converter

The non-inverting buck-boost converter is essentially one buck and one boost converter connected in series. This type of converters can produce lower as well as higher output voltages than the supplied one. The converter topology employed in this paper has a separate control input for each stage. The control signal is a PWM waveform with a constant frequency and controlled duty-cycle. The schematic representation of such a converter is shown in Fig. 1.

A. Schematic representation

The basic operation of the buck-boost converter may be explained as a two stage process: firstly, the energy from the power supply is pumped into the inductor by means of the switch s_1 and after that the transferred energy is discharged into the output capacitor and the load through the switch s_2. The synchronous implementation of this topology allows better efficiency by replacing a constant voltage drop across the diode by a very small resistive load across the power transistor. To further minimize the losses in high current applications this topology may be extended by coupling several transistors in parallel. Throughout the paper a synchronous implementation of the converter is assumed. However, with an appropriate change in the model, the same technique can be successfully applied for control of a non-synchronous converter.

B. Mathematical model

In this paper half-bridges from the buck and the boost stages of the converter are modeled as ideal switches, i.e., s_1 and s_2, respectively. The resistive losses from power transistors and inductor L are combined in R_L. The continuous dynamics of the lumped parameter system from Fig. 1 is described by the next differential equation:

$$\begin{aligned}\dot{x}(t) &= (A_1 + s_2(t)A_2)x(t) + s_1(t)B_1 v_s(t) + \\ &\quad + (W + s_2(t)B_2)i_{load}(t), \\ &\quad t \in \mathbb{R}_+,\end{aligned} \tag{1}$$

where $x := \begin{bmatrix} v_C & i_L \end{bmatrix}^\top \in \mathbb{R}^2$ is the system state (i.e., the capacitor voltage and the inductor current), v_s is the supply voltage, i_{load} is the load current and $s_1(t)$, $s_2(t)$ are the switch functions. For duty-cycle less than 1 the switch functions are defined as follows:

$$\begin{aligned}s_i(t) &= \begin{cases} 1, & \frac{t}{T_s} - k \in [0, d_i(k)), \\ 0, & \frac{t}{T_s} - k \in [d_i(k), 1), \end{cases} \\ &\quad k \in \mathbb{Z}_+,\end{aligned} \tag{2}$$

and $s_i(t) = 1$ for $d_i(k) = 1$. Above, $\mathbb{R}_+$ and $\mathbb{Z}_+$ denote the set of non-negative real and integer numbers, respectively. In (2) T_s is the PWM period and $d_i(k) \in [d_i^{min}, d_i^{max}] \subseteq [0, 1], i \in \{1, 2\}$ are the duty-cycles for the buck and the boost stage of the converter. The matrix coefficients from (1) are specific to the circuitry implementation and they are described in terms of system parameters such as inductance, capacitance and resistance, i.e.,

$$\begin{aligned}A_1 &= \begin{bmatrix} 0 & 0 \\ 0 & -\frac{R_L}{L} \end{bmatrix}, \; A_2 = \begin{bmatrix} 0 & \frac{1}{C} \\ -\frac{1}{C} & -\frac{R_C}{L} \end{bmatrix}, \\ B_1 &= \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix}, \; W = \begin{bmatrix} -\frac{1}{C} \\ 0 \end{bmatrix}, \; B_2 = \begin{bmatrix} 0 \\ \frac{R_C}{L} \end{bmatrix}.\end{aligned} \tag{3}$$

The average voltage drop across R_C is zero for the constant average voltage at the output capacitors. Therefore, the voltage across the capacitor can be considered as the output of the system without affecting the generality of the proposed methodology.

Usually, maintaining a constant value of the output voltage is required from the controller. The set of duty-cycle pairs which provides the given output voltage in steady state operation is not finite. If the losses are neglected then the relation between the duty-cycles d_1 and d_2 which give the same output voltage is

$$d_1 = d_2 \frac{V_{ref}}{v_s}, \tag{4}$$

where V_{ref} is the reference voltage for the output. At each sampling time k, the controller must decide on a best suitable pair of d_1 and d_2 from the set of control inputs which drives the system to the reference output voltage. The easiest way to do this is to impose a direct relation between the control inputs. This transforms the converter into a single-input system. This simple approach yields lower performance compared to what is achievable by exploiting the new degree of freedom given by the second control input. As it will be shown in Section IV, adding a separate reference for the inductor current may increase the overall performance of the converter and/or minimize power losses.

For the purpose of this paper, the continuous-time dynamic model (1) is averaged and a discrete-time model is derived

from it using the forward Euler approximation method. The interested reader is referred to [3], [5] for the details. The model considered for control algorithm synthesis acquires the following form:

$$\begin{aligned} x(k+1) &= \phi(x(k), d_1(k), d_2(k)) \\ &:= (I_2 + A_1 T_s + d_2(k) A_2 T_s) x(k) \\ &\quad + (W + d_2(k) B_2) T_s i_{load}(k) \\ &\quad + d_1(k) B_1 T_s v_s(k), \end{aligned} \tag{5}$$

where I_2 is the identity matrix of dimensions 2×2.

Remark II.1 In equation (5), it can be observed that ϕ is a function of the supply voltage v_s and the load current i_{load} as well. These two signals are disturbances that can deviate from a pre-specified nominal value. In practice, they are either measured or estimated. For simplicity, in what follows, it is assumed that $v_s(k)$ and $i_{load}(k)$ are known for all $k \in \mathbb{Z}_+$. Further work deals with the inclusion of a suitable estimator in the proposed MPC scheme. □

III. Problem formulation and controller design

A. Problem formulation

The primary goal of the controller is to maintain the output voltage of the converter at the prescribed level while maintaining the inductor current and control inputs within specified limits. More formally, let $\mathbb{U} = \{(d_1, d_2) | d_i \in [d_i^{min}, d_i^{max}], i \in \{1, 2\}\}$ be the set of admissible control inputs and let $\overline{\mathbb{X}}$ be the set of system states which do not violate the restrictions for the inductor current. Let $\mathbb{X} \subseteq \overline{\mathbb{X}}$ denote a robust constrained controlled invariant set for the system (5), i.e., for each $x \in \mathbb{X}$ there exists a pair of control inputs $(d_1, d_2) \in \mathbb{U}$ such that $\phi(x, d_1, d_2) \in \mathbb{X}$ for all admissible values of v_s and i_{load}.

Another major requirement regarding the closed-loop system is stability. Considering the results from [15], stability of the closed-loop system can be attained if there exist control inputs $d_1(k)$, $d_2(k)$ and a relaxation variable $\lambda(k)$ that satisfy the following inequalities for all $k \in \mathbb{Z}_+$ and $x(k) \in \mathbb{X}$, i.e.

$$V(\phi(x(k), d_1(k), d_2(k)) - r) \leq \rho V(x(k) - r) + \lambda(k), \tag{6a}$$

$$\lim_{k \to \infty} \lambda(k) = 0, \tag{6b}$$

$$\phi(x(k), d_1(k), d_2(k)) \in \mathbb{X},\ \lambda(k) \geq 0,\ \rho \in (0, 1), \tag{6c}$$

$$\text{where } r := \begin{bmatrix} V_{ref} & I_L^{ref} \end{bmatrix}^\top, \tag{6d}$$

and I_L^{ref} is the reference for the inductor current. One can select $V(\cdot)$ as a local CLF for the linearized model of the system (5) around the nominal operating point, which can be calculated systematically, see [15]. To assure (6b), one can parameterize the relaxation variable as $\lambda(k) := \Lambda \mu^k$ for some $\Lambda > 0$ and some $\mu \in (0, 1)$, for all $k \in \mathbb{Z}_+$. The local CLF used throughout this paper is of the form

$$V(x) = \|Px\|_\infty, \tag{7}$$

where $P \in \mathbb{R}^{p \times 2}$, $p \in \mathbb{Z}_+$, $p \geq 2$ and $\|\cdot\|_\infty$ is the infinity norm. One of the available methods for numerical computation of the matrix P is presented in [5], see also [17] for more details.

As it was mentioned in Section II-B the set of values for control inputs for which the desired voltage is achieved in steady state is not finite. To decide on the best pair of duty-cycles, a specific reference for the inductor current was imposed. To achieve a high efficiency, the inductor current should be minimized. The expression of the lowest sustainable current through the inductor is given in the following equation:

$$I_L^{ref} = \frac{1}{d_2^{st}} i_{load}, \tag{8a}$$

$$d_2^{st} = \min\{d_2^{max}, d_1^{max} \frac{v_s}{V_{ref}}\}. \tag{8b}$$

The losses in the circuitry are neglected in (8).

Considering the lowest sustainable value for the inductor current has several drawbacks. First of all, without taking into account the losses the expressed value in (8a) is lower than the real current needed for stationary operation of the converter under supply voltage v_s, load current i_{load} and $v_C = V_{ref}$. Thus, this value can cause feasibility issues with respect to the conditions imposed by (6). Also, as it will be shown in Section IV, having the lowest sustainable energy conserved in the inductor can deteriorate the performance of the converter. To formulate the tradeoff between efficiency and performance two additional coefficients are introduced in (8), i.e.,

$$\begin{gathered} I_L^{ref} = \alpha \frac{1}{d_2^{st}} i_{load} + \theta, \\ \alpha \geq 1 \text{ and } \theta \geq 0. \end{gathered} \tag{9}$$

Considering all of the above mentioned, the optimization problem to be solved by the developed control scheme at each sampling instant can be summarized as follows:

Problem III.1 At time $k \in \mathbb{Z}_+$ let $x(k) \in \mathbb{R}^2$ be given. Calculate a pair of control inputs $(d_1(k), d_2(k)) \in \mathbb{U}$ such that the cost function

$$\begin{aligned} &J(x(k), d_1(k), d_2(k), r) = \\ &\|Q(\phi(x(k), d_1(k), d_2(k)) - r)\|_\infty, \end{aligned} \tag{10}$$

is minimized and the following restrictions are satisfied

$$\begin{aligned} &\|P(\phi(x(k), d_1(k), d_2(k)) - r)\|_\infty \leq \\ &\qquad \rho \|P(x(k) - r)\|_\infty + \lambda(k), \end{aligned} \tag{11a}$$

$$\lambda(k) = \mu \lambda(k-1) \text{ for } k \geq 1 \tag{11b}$$

$$\lambda(0) = \Lambda,$$

$$\phi(x(k), d_1(k), d_2(k)) \in \mathbb{X}.$$

The matrix $Q \in \mathbb{R}^{2 \times 2}$, the real scalars $\rho, \mu \in (0, 1)$ and $\Lambda \in \mathbb{R}_+$ are the tuning parameters of the controller. More details on how to select these parameters are given in the remainder of the paper.

Remark III.2 The selection of the parameter Λ is mainly dictated by $\sup(\|P(\phi(x,d1,d2)-r)\|_\infty - \rho\|P(x-r)\|_\infty)$ over $x \in \mathbb{X}$ and $(d_1,d_2) \in \mathbb{U}$.. The parameter μ should be selected large enough to allow the system to converge to the desired reference and to maintain Problem III.1 feasible for a wide operating range. Notice that large disturbances in conjunction with an a priori fixed value of the parameter μ can still lead to infeasibility of Problem III.1. To address this issue, an auxiliary optimization variable $s(k) \in \mathbb{R}_+$ can be introduced along with a highly penalized cost function as follows. The cost function (10) can be replaced by

$$J(x(k),d_1(k),d_2(k),s(k),r) = \|Q(\phi(x(k),d_1(k),d_2(k))-r)\|_\infty + qs(k) \quad (12)$$

and restriction (11a), (11b) with

$$\|P(\phi(x(k),d_1(k),d_2(k))-r)\|_\infty \leq \rho\|P(x(k)-r)\|_\infty + \lambda(k) + s(k), \quad (13a)$$

$$\lambda(k) = \mu(\lambda(k-1)+s(k-1)) \text{ for } k \geq 1 \quad (13b)$$

where $q \in \mathbb{R}_+$. Extensive experiments indicate that the modified version of Problem III.1 given above remains recursively feasible for a wider range of operating conditions and disturbance scenarios. This encourages us to further develop a formal analysis of the properties of the closed-loop system. However, it should be noted that in this case performance and stability conditions are no longer decoupled. □

B. Controller design

The first step in the controller design procedure is the construction of a local CLF for the linearized dynamics. The system dynamics is linearized around nominal reference r^{nom} calculated for nominal values of load current i_{load}^{nom}, supply voltage v_s^{nom} and V_{ref}. The linearized model of the system dynamics (5) is

$$x'(k+1) = A_l x'(k) + B_l u'(k), \quad (14a)$$

$$A_l = I_2 + A_1T_s + A_2T_s\begin{bmatrix}0 & 1\end{bmatrix}u^{nom}, \quad (14b)$$

$$B_l = \begin{bmatrix}B_1T_s & B_2T_si_{load}^{nom} + A_2T_sr^{nom}\end{bmatrix}, \quad (14c)$$

$$x'(k) = x(k) - r^{nom},\ u'(k) = u(k) - u^{nom}, \quad (14d)$$

$$u^{nom} = \begin{bmatrix}B_1T_sv_s^{nom} & B_2T_si_{load}^{nom} + A_2T_sr^{nom}\end{bmatrix}^{-1} \times T_s(Wi_{load}^{nom} + A_1r^{nom}), \quad (14e)$$

where $u(k) := \begin{bmatrix}d_1(k) & d_2(k)\end{bmatrix}^\top$.

Next, the control input is parameterized as a linear state-feedback, i.e., $u(k) = Kx(k)$. Then, the matrices P and K are obtained as a solution of the following inequality:

$$\|P(A_l + B_lK)(P^\top P)^{-1}P^\top\|_\infty - \rho \leq 0. \quad (15)$$

In [5] it was shown that this provides a local CLF of the form (7) for the considered system. A solution to inequality (15) can be obtained using a nonlinear optimization solver, such as the *fmincon* solver of Matlab. Notice that this optimization problem only needs to be solved once, off-line, and it does not affect the computational load of the real-time algorithm. Also, the local control gain K is never used by the rto-MPC algorithm.

To formulate Problem III.1 as a single linear program, the minimization of (10) at each $k \in \mathbb{Z}_+$ for a given $x(k)$ is casted as follows:

$$\min_{d_1(k),d_2(k)} z(k) \quad (16a)$$

$$\|Q(\phi(x(k),d_1(k),d_2(k))-r)\|_\infty < z(k). \quad (16b)$$

As for any vector $x \in \mathbb{R}^n$, $\|x\|_\infty \leq c$ is equivalent to $\pm[x]_i \leq c$ for all $i \in \{1,\dots,n\}$, where $[x]_i$ is the i-th element of x, and ϕ is an affine function of d_1 and d_2 for a given x, Problem III.1 can be casted as an equivalent linear program with 4 optimization parameters and 17 inequalities.

As it is typically the case with horizon-1 rto-MPC algorithms for the considered power converter dynamics, in startup and heavy boost operation the algorithm may yield an undesirable control action. For example, the fastest way to rise the capacitor voltage is to maintain $d_2(k) = 1$, but the problem is that in boost operation this is not sustainable. For the correct operation the boost stage duty-cycle should be less than $\frac{v_s}{v_C}d_1^{max}$. To assure that capacitor voltage can rise at any given moment the upper bound of $d_2(k)$ will be adjusted dynamically, at every sampling time $k \in \mathbb{Z}_+$, i.e.,

$$d_2^{max}(k) := \beta d_1^{max}\frac{v_s(k)}{v_C(k)}, \quad (17)$$

where $\beta \in (0,1)$ is a coefficient that guarantees a rise of inductor current when $d_1(k) = d_1^{max}$. Choosing β close to 1 will slow down the startup response in heavy boost situations, but a small value for β will affect negatively the overall converter performance, i.e., larger ripple and higher losses during boost operation.

IV. Simulation results

To evaluate the performance of the described algorithm, which corresponds to Problem III.1, several simulations were performed using the model (1) with the following circuit parameters: $R_L = 0.2\Omega$, $R_C = 0.05\Omega$, $C = 22\mu F$, $L = 220\mu H$. The sampling frequency of control algorithm was considered to be the same as the frequency of the PWM and equal to 100kHz. The maximum allowed current through the inductor was set equal to 3A.

Two simulation scenarios were considered. In both scenarios the desired output voltage was taken equal to 20V. The first scenario contains a mix of buck and boost modes with a step in the supply voltage from 10V to 30V and then from 30V to 15V, and a step change in the load current from 0.2A to 0.5A and then from 0.5A to 0.1A. The supply voltage and load current profiles for this scenario are presented in Fig. 3 and Fig. 2 respectively. The second scenario considers startup operation in heavy boost mode and is carried out for a constant load current of 0.2A and supply voltage of 5V.

For this particular case study, the values $Q = \begin{bmatrix}4 & 0.1\\ 0.1 & 1\end{bmatrix}$, $\Lambda = 10$, $\mu = 0.95$, $\beta = 0.9$, $\theta = 0.3$ and $\alpha = 1$ offered a good tradeoff between disturbance rejection, minimization of

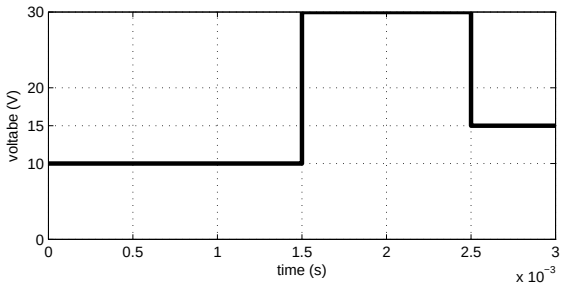


Fig. 2. The supply voltage profile for the first scenario.

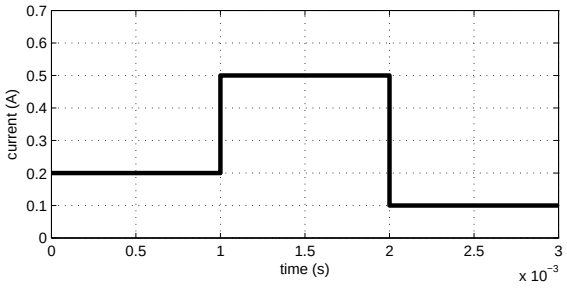


Fig. 3. The load current profile for the first scenario.

converter losses and a fast startup. These values were used in all of the following simulations, unless other values are specified. The solution for calculating the local Lyapunov function indicate in Section III-B returned the matrix $P = \begin{bmatrix} 2 & 0.1 \\ 0.2 & 4 \end{bmatrix}$ for $\rho = 0.9$. The waveforms of the voltage from output capacitor and inductor current for the first simulation scenario are plotted in Fig. 4 and Fig. 5, respectively.

Next, the analysis of the closed loop system behavior for different inductor current references is presented. Two simulation results for different I_L^{ref} modified by $\theta = 0.1$ and $\theta = 0.6$, with $\alpha = 1$ in both cases, are presented in Fig. 6, which shows the magnified waveform of the output voltage and Fig. 7, which shows the inductor current. Increasing θ by 0.5 gives in some situations up to several times times better disturbance rejection, but will add 0.5A to the inductor current, which translates into higher losses and thus, a slightly reduced efficiency. Another concern is that the increase of the inductor current translates into an increase in the output voltage ripple. Changing the α parameter will

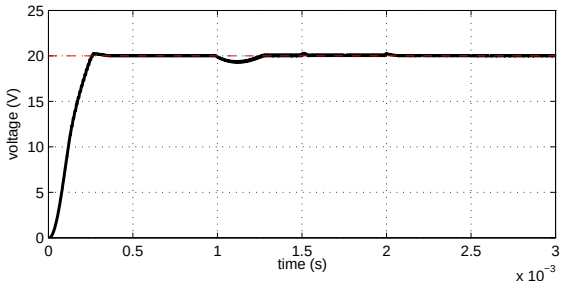


Fig. 4. Output voltage, $\beta = 0.9$, $\theta = 0.3$, $\alpha = 1$ (first scenario).

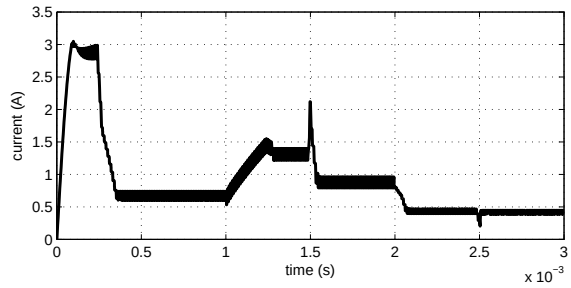


Fig. 5. Inductor current, $\beta = 0.9$, $\theta = 0.3$, $\alpha = 1$ (first scenario).

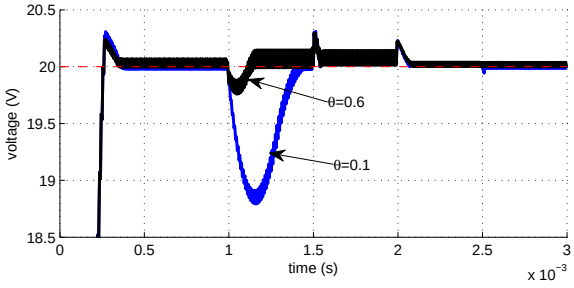


Fig. 6. Output voltage for $\theta = 0.1$ and $\theta = 0.6$ (first scenario).

give similar results, but the extra current through inductor will change according to the load current. With $\alpha > 1$ better disturbance rejection as well as higher losses for higher load current are expected. This behavior might be beneficial for applications where high efficiency in standby mode and good reference tracking in full load mode are required.

The second scenario shows the system behavior in heavy boost situation. Two simulations with $\beta = 0.75$ and $\beta = 0.95$ were performed in this scenario. It can be observed in Fig. 8 that the startup in heavy boost mode is significantly better for $\beta = 0.75$. Nevertheless, a further decrease of β is not reasonable. This will limit the set of feasible control inputs and the overall converter performance in other mods than heavy boost will decrease. The current trajectories for this operation mode are shown in Fig. 9.

The worst-case control algorithm calculation time was below $0.1ms$[1]. Shorter calculation times are achievable by

[1]Calculations were performed on one core of the Intel T9600 processor, in Matlab 7.9.0.529, using the CDD Criss-Cross LP solver.

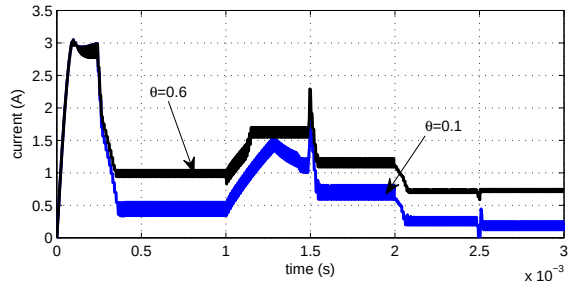


Fig. 7. Inductor current for $\theta = 0.1$ and $\theta = 0.6$ (first scenario).

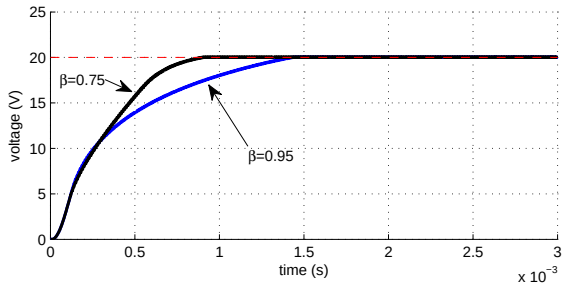


Fig. 8. Output voltage for $\beta = 0.75$ and $\beta = 0.95$ (second scenario).

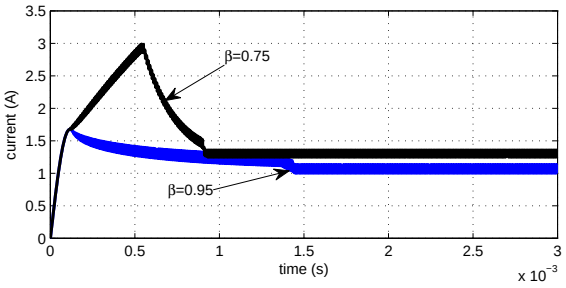


Fig. 9. Inductor current for $\beta = 0.75$ and $\beta = 0.95$ (second scenario).

deploying the algorithm in a real-time environment. Preliminary experiments show that a real-time implementation of the algorithm can cope with a sampling time of $10\mu s$, which corresponds to a typical operating frequency for DC/DC power converters.

V. Conclusions

This paper investigated the applicability of a novel horizon-1 rto-MPC scheme to the control of DC/DC converters. The novel aspect of the proposed MPC scheme consists in using flexible control Lyapunov functions as a non-conservative stabilization method. The design of the control scheme was illustrated for a two-inputs buck-boost DC/DC converter topology. The promising results in terms of performance and computational load requirements obtained in simulation encourage a further theoretical analysis of the closed-loop properties and real-time experiments.

VI. Acknowledgements

The authors are grateful to Dr. Sébastien Mariéthoz for many useful discussions and sharing his two-inputs buck-boost DC/DC Matlab simulator. The research presented in this paper is supported by the Veni grant "Flexible Lyapunov Functions for Real-time Control", grant number 10230, awarded by STW and NWO as well as by the IOP EMVT-II project "Ultra-High Precision Power Amplifier", number 08201.

References

[1] M. K. Kazimierczuk, *Pulse-width modulated DC-DC power converters*. Wiley, 2008.

[2] H. Qiu, G. zhong Cao, J. Pan, and L. ming Lin, "The development of magnetic levitation ball control system based on TMS320F2812," in *3rd International Conference on Power Electronics Systems and Applications*, 2009.

[3] M. Lazar and R. De Keyser, "Nonlinear predictive control of a DC-to-DC converter," in *Symposium on Power Electronics, Electrical Drives, Automation & Motion - SPEEDAM*, Capri, Italy, 2004.

[4] T. Geyer, G. Papafotiou, and M. Morari, "Model Predictive Control in Power Electronics: A Hybrid Systems Approach," in *IEEE Conference on Decision and Control*, Seville, Spain, 2005.

[5] M. Lazar, B. J. P. Roset, W. P. M. H. Heemels, H. Nijmeijer, and P. P. J. v. d. Bosch, "Input-to-state stabilizing sub-optimal nonlinear MPC algorithms with an application to DC-DC converters," *International Journal of Robust and Nonlinear Control, Special Issue on Nonlinear MPC of Fast Systems*, vol. 18, pp. 890–904, 2008.

[6] Y. Xie, R. Ghaemi, J. Sun, and J. Freudenberg, "Implicit model predictive control of a full bridge DC-DC converter," *IEEE Transactions on Power Electronics*, vol. 24, no. 12, pp. 2704–2713, 2009.

[7] A. Beccuti, S. Mariéthoz, S. Cliquennois, S. Wang, and M. Morari, "Explicit model predictive Control of DC-DC switched-mode power supplies with extended Kalman filtering," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 6, pp. 1864–1874, june 2009.

[8] S. Mariéthoz, S. Almer, and M. Morari, "Optimal control of a two control input buck-boost converter," in *48th IEEE Conference on Decision and Control*, 15-18 2009.

[9] S. Mariéthoz and M. Morari, "Explicit model-predictive control of a PWM inverter with an *LCL* filter," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 2, pp. 389–399, 2009.

[10] D. Quevedo, R. Aguilera, M. Pérez, and P. Cortés, "Finite control set MPC of an AFE rectifier with dynamic references," in *International Conference on Industrial Technology*, Viña del Mar - Valparaíso, Chile, March 2010.

[11] S. Richter, S. Mariéthoz, and M. Morari, "High-speed online MPC based on a fast gradient method applied to power converter control," in *American Control Conference (ACC)*, june 2010, pp. 4737 –4743.

[12] A. Bemporad, F. Borrelli, and M. Morari, "Model predictive control based on linear programming - The explicit solution," *IEEE Transactions on Automatic Control*, vol. 47, no. 12, pp. 1974–1985, 2002.

[13] J. B. Rawlings and D. Q. Mayne, *Model predictive control theory and desing*. Nob Hill Pub, 2009.

[14] R. E. Kalman and J. E. Bertram, "Control system analysis and design via the second method of Lyapunov, II: Discrete-time systems," *Transactions of the ASME, Journal of Basic Engineering*, vol. 82, pp. 394–400, 1960.

[15] M. Lazar, "Flexible control Lyapunov functions," in *American Control Conference*, St. Louis, Missouri, USA, 2009, pp. 102–107.

[16] N. Mohan, T. M. Undeland, , and W. P. Robbins, *Power Electronics: Converters, Applications and Design*. Wiley, 1989.

[17] M. Lazar, "Model predictive control of hybrid systems: Stability and robustness," Ph.D. dissertation, Eindhoven University of Technology, The Netherlands, 2006.

Architectural Support for Subroutine Execution Time Monitoring in Embedded Microprocessors

A. Stan, L. Panduru, Fl. Ungureanu

Abstract—Many embedded systems are used to implement safety critical applications that have to satisfy real time requirements. The real time requirements are derived from the functional characteristics of the physical system controlled by the application. In these systems, the correct operation as a whole depends also on the timeliness of the results. Results that are available earlier or later than a specified time, although they may be logically correct, they may cause unpredictable system behavior because of the timeliness violation. This paper presents the design of a digital module (watchdog) that may be included in embedded microprocessors in order to provide a mean to detect and signal the violation of timing characteristics of the executing code. The timing behavior monitoring is performed by measuring the execution time of subroutines and comparing the results with reference values. The proposed module does not require any modification of the monitored embedded microprocessor architecture. The module may be also used in non real time systems to implement security checking mechanisms by detecting various abnormal operating conditions that alter the execution time of subroutines (e.g. insertion of malicious/virus code).

I. Introduction

MOST embedded systems have stringent requirements for the safety and security characteristics due to their integration into safety critical systems. The failure of a safety critical system leads to unacceptable consequences with a negative impact on assets or even human lives. A failure is the manifestation or the effect of a fault in the system.

The handling of the faults is performed at design time or at run time [1]. At design time, there are used certain methods in order to avoid the occurrence of faults in the system: repeated testing (practical method) or formal verification by using modeling and simulation (theoretical method). At run time, there are used some measures in order to tolerate the faults that were not detected and removed during design time.

The safety and security characteristics of an embedded system may be dependent of the timing behavior of the running application. For example, in automotive industry the physical environment imposes the execution speed of the applications that are running on ECUs (Electronic Control Units). If the running application does not keep the pace with the environmental parameters rate of change then catastrophic consequences may occur.

The timing behavior of a running application may influence other characteristics of the system, namely the quality of delivered service. For example, missing samples from a multimedia playback causes a low quality of the service. This is an example of a non safety critical application where the violation of timing characteristics decreases the performance of the application and may get to the point where the application generates an unusable output.

The timing behavior of an embedded system is influenced by some factors [2]. Embedded systems are designed as reactive systems: they make intensive use of interrupts in order to quickly react to external triggered events. The interrupts may be seen as a source of unknown external interference. They are asynchronous events for the currently executing code and they usually have a higher priority, so the execution time for the service routines adds to the total execution time. The unpredictable nature of external interrupts represents a threat to the timing behavior of an application.

Another factor that influences the timeliness of the operation of an embedded system is the communication process with other systems. Nowadays, more and more systems make use of distributed processing that includes multiple processing nodes and interconnection networks. The communication process may be easily disturbed by unknown interference that leads to packet corruption or loss.

The limited knowledge and understanding of the external interference are the cause for the timeliness violation of embedded applications.

In modern embedded microprocessors there are implemented many techniques aimed at improving the average performance: speculation, branch prediction, cache memories hierarchies and other. These methods enhance the average performance but they make the system very hard to analyze and to predict its behavior. It is true that for real time applications such architectural features are not the main requirement. The choice of a microprocessor architecture for this kind of applications is driven by other criteria like predictability.

The limited analyzability of a microprocessor architecture is another important cause for the timeliness violation for embedded applications.

Because of the before mentioned reasons there is a need

Manuscript received May 14, 2010.

A. Stan is with the Technical University of Iasi, Romania (phone: +40-232-278680 / int. 1349; e-mail: andreis@cs.tuiasi.ro).

L. Panduru is with the Technical University of Iasi, Romania (phone: +40-232-278680 / int. 1349; e-mail: pandurluk@cs.tuiasi.ro).

Fl. Ungureanu is with the Technical University of Iasi, Romania (phone: +40-232-278680 / int. 1309; e-mail: fungurea@cs.tuiasi.ro).

to design and implement mechanisms that detect and signal timing violations in embedded systems in order to avoid unwanted consequences.

II. Related work

There are numerous articles that present methods for checking the correctness of the code execution on microprocessors by using architectural enhancements.

In paper [3] the authors present the concept of watchdog processors and highlight some implementations that were available at that time. A watchdog processor is a coprocessor that performs concurrent system-level error detection by monitoring the behavior of main processor. The information that watchdog processor uses to detect errors can be about: memory access behavior, control flow, control signals or the reasonableness of the results. The operation of a watchdog processor is a two phase process. First, the watchdog is provided with some information about the processor to be checked and second, it collects the relevant information concurrently as the main processor is running. Error detection is done by comparing the runtime acquired information with reference information provided at setup. For our monitoring device we use the same philosophy, but the information about the correctness of the operation is the execution time of the subroutines.

The authors of paper [4] propose a methodology for embedding security monitoring technique within the microinstructions forming self-monitoring instructions. There are created machine instructions that monitor themselves for improper operations via embedding extra microinstructions. The monitoring mechanism is applied to monitor: the return address for subroutines, the control flow instructions, data path of the microprocessor and the memory access.

In paper [5] the authors present a hardware-based scheme to detect anomalies by checking program execution paths dynamically. The anomalous path checking mechanism is built into a secure processor that stores a record of whole program path (WPP) acquired in training phase. At runtime, the anomalies are detected by checking if the current executing path exists in the paths collected during training.

The authors of paper [6] propose a dedicated hardware monitor that enforces permissible behavior as program executes. They present techniques to monitor the program behavior at different levels of granularity: the inter-procedural control flow, the intra-procedural control flow and the integrity of instruction stream within each basic block (a block free of branches). The first two levels of the monitor are implemented using a Finite State Machine (FSM) that models the function call graph of the program or the control flow graph within a function. An invalid execution path is represented by a transition to invalid state or an invalid transition. The integrity of instruction stream is validated by using a cryptographic hash function of the instructions in a basic block. Hash values are computed beforehand, loaded into the hardware monitor and checked during program execution.

In paper [7] the authors propose an approach that extends the embedded microprocessors with hardware that significantly accelerates the execution of the additional computations involved in memory-safe execution by designing custom instructions that perform various memory safety checks.

In order to augment the original architecture with error detection mechanisms, all the previous presented solutions include a customized design flow or methodology for the applications to take advantage of the new added features.

III. Monitoring architecture

The monitoring module (watchdog – WDT) is connected to the microprocessor address bus and to the instruction bus, as presented in Figure 1. It signals abnormal operation by issuing control signals like reset, interrupt or NMI or issuing appropriate opcodes on the instruction bus.

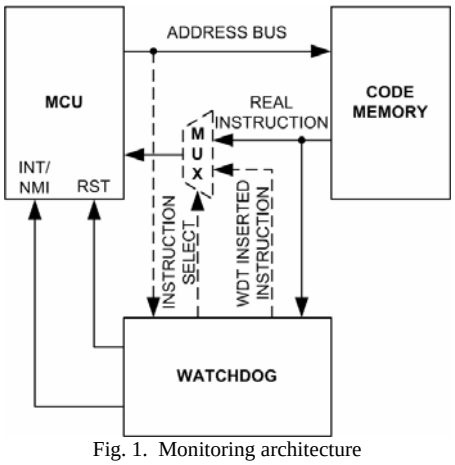


Fig. 1. Monitoring architecture

The monitoring module is built using the following data structures and algorithms.

The watchdog stack (WS) is a last in first out data structure that stores the called subroutines starting addresses. The top of the WS contains the starting address of the last called subroutine. This data structure is used to easily retrieve the last called subroutine starting address when a return instruction is detected on the instruction bus.

Execution Time Counters – this component is made by a counter array; each counter may be individually selected in order to reset its internal value or to enable or disable its operation (counting). Each monitored subroutine has a corresponding execution time counter. The number of counters limits the number of simultaneously monitored subroutines, but it may be appropriately chosen to fit the application needs.

The watchdog uses a table – named Monitoring Info Table (MIT) – with information about the expected timing behavior of all or only critical subroutines of the application. The table contains the starting address of the monitored subroutines, the index of the timer from the counter array

used to monitor the timing behavior and the values of four parameters that describe the timing behavior of the subroutine, as presented in Figure 2:

- Lower timing bound (LTB) – the minimum acceptable execution time
- Best case execution time (BCET) – the minimum execution time computed by simulation or by testing
- Worst case execution time (WCET) – the maximum execution time computed by simulation or by testing
- Upper timing bound (UTB) – the maximum acceptable execution time

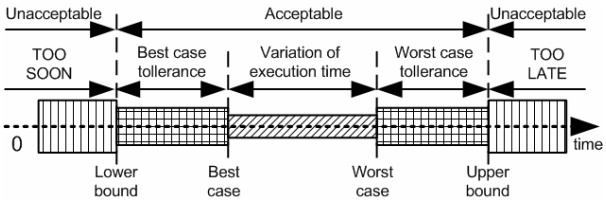


Fig. 2. Timing characteristics of a subroutine.

These values define five possible domains for the execution time of a subroutine as presented in the figure 2. The acceptable domain for execution time is defined by the difference between UTB and LTB. All other values for execution time that exceed UTB or are lower than LTB are unacceptable values for systems with real time requirements or may indicate abnormal timing behavior. This situation may be called a timing exception for a real time system. This event may be of equal value or importance as it is overflow and division by 0 for arithmetic operations and ultimately for the correct/safe/secure operation of the system. If arithmetic exceptions, like those mentioned above, signal a possible corrupted or wrong numeric value, the timing exception, in the context of real time systems, may signal an obsolete data being used because additional unwanted delay added between the moment the data was produced and the moment the data it is used. This may have a catastrophic impact on system operation.

The watchdog module monitors the address bus for code memory and the instruction extracted from code memory. The watchdog decodes the instruction in order to detect subroutine calls and the matching return from subroutine instructions.

When a timing violation occurs this triggers a handling mechanism. There are three possible handling mechanisms:

- Reset the MCU – this is the usual method that a classic watchdog uses
- Issues a NMI or INT to the MCU – this method may be used if a handling mechanism is implemented in software as a handling routine for NMI interrupt or INT;
- Insert an instruction or sequence of instructions to the MCU – the inserted instructions may be a call or a jump to a handling routine; this mechanism is similar with the NMI interrupt, but it is more flexible by allowing a specific handling routine for timing violation for each subroutine.

The watchdog uses a finite state machine to sequence the operations need by its operation. The inputs of the state machine are condition signals generated by logical operations between the values of WS, MIT, address bus, instruction bus, as explained below. The outputs of the finite state machine are control signals.

The operation of the watchdog is presented in Figure 3.

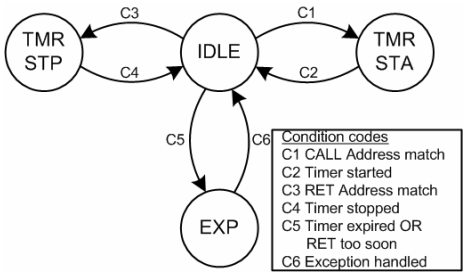


Fig. 3. State chart for the FSM of the monitoring module.

The initial state of the watchdog is IDLE state. In this state, the watchdog monitors the instruction bus and also checks the state of execution time counters for monitored subroutines. There are three possible transitions from this state. The transitions are discussed next in the decreasing order of their priorities.

If one execution time counter reaches a value greater than corresponding WCET then a transition to the exception handling state is performed.

If the instruction currently fetched is a CALL instruction, then a transition to the setup monitoring state for the called subroutine.

If the instruction currently fetched is a RET instruction, then the value of the counter is compared with the BCET of the returning subroutine. If this value is lower than BCET then a transition to the exception handling state is performed, otherwise a transition to the end monitoring state for the currently monitored subroutine is performed.

The setup monitoring state (TMR STA) checks if the called subroutine needs to be monitored. This is done by searching subroutine starting address (the address field in the call instruction) in the MIT. If the address is found, then the corresponding execution time counter is started and the starting address is stored on the top of the WS. The only transition from this state is to the IDLE state and it takes place when the setup is done.

The end monitoring state (TMR STP) extracts the value from the watchdog stack and stops the execution time counter associated with the returning subroutine. A transition to IDLE state is performed. The operation explained before assumes that the return instruction detected on the instruction bus correspond to the last called subroutine (a normal sequence of operation).

The development of an application that takes advantage of

this monitoring architecture must follow a specific flow. The software is designed and implemented first. In order to generate the monitoring module, the critical subroutines must be determined and after that their execution time characteristics must be computed. This may be accomplished by using static analysis [8] or by repeated testing. With the information provided in this stage, the monitoring module may be generated and together with the binary of the application the final architecture is implemented. The design flow is presented in figure 4.

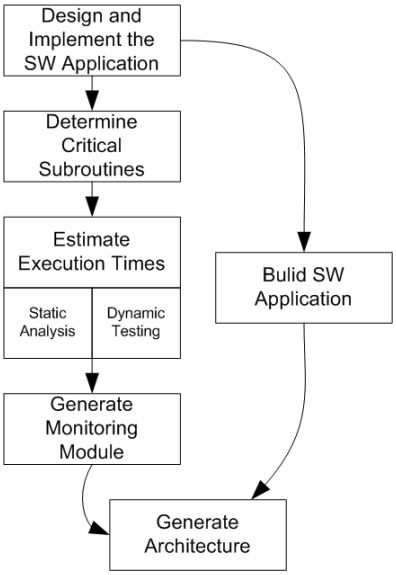


Fig. 4. The design flow.

IV. Conclusions

The implemented monitoring module offers a mechanism that allows the dynamic checking of the time behavior of subroutines.

In order to evaluate the monitoring module, we used the Spartan3E FPGA from Xilinx and a very simple soft-core processor: PicoBlaze. This is a constant coded programmable state machine which is designed and optimized by Xilinx for its FPGAs: this is a machine based on constants which are allowed to be specified within any instruction word. We designed two scenarios: in the first we synthesized only the PicoBlaze and in the second we augmented the first design with the watchdog module. The cost for implementing the two scenarios is presented in Table I. The table presents the quantity of FPGA digital resources required by the two implementations.

TABLE I
LOGIC UTILIZATION AND FREQUENCY

Resources	Picoblaze Alone	Picoblaze and WDT
Number of Slice Flip Flops	76	129
Number of 4 input LUTs	176	411
Number of occupied slices	98	236
Operating frequency (MHz)	116.136	93.092

We notice that the actually implemented monitoring module uses significantly more resources compared to the processor alone. This can be explained by the simplicity of the soft-core processor and by the fact that this processor is highly optimized for FPGA implementation. The monitoring module (WDT) uses a lot of memory resources – see Table I – in order to implement the stack and also to store the relevant timing information for each monitored subroutine. The monitoring module uses logic resources to implement large and numerous comparators for address matching logic.

The degradation of the maximum operating frequency also may be observed. This fact is due the existence of a critical path in the implementation of the stack inside the monitoring module. This may be eliminated by redesigning the stack or by specifying additional constraints for the synthesis process (or even change the synthesis tools).

For more complex processors we expect that the resource overhead required by the implementation of our monitoring module to be less significant.

V. Further work

Further work is aimed at implementing the monitoring module into more complex microprocessors architectures like MicroBlaze, OpenRISC or SPARK T1 or T2. Also, a mechanism for handling indirect subroutine calls should be designed.

VI. Acknowledgement

This work is supported by Romanian Minister of Education, Research and Innovation in the framework of National Program for Research, Development and Innovation (PNCDI-2) under partnership project no. 11-070 SIMPA.

References

[1] F. Salewski and A. Tylor, "Fault Handling in FPGAs and Microcontrollers in Safety-Critical Embedded Applications – A Comparative Survey", *10th Euromicro Conference on Digital System Design*, Lubeck, 2007, pp. 124 – 131

[2] L. Thiele and R. Wilhelm, "Design for Timing Predictability", *Journal of Real Time Systems*, Kluwer Academic Publishers, 2004

[3] A. Mahmood and E. J. McCluskey, "Concurrent Error Detection Using Watchdog Processors – A Survey", *IEEE Transactions on Computers*, vol. 37, no. 2, 1988

[4] R. G. Ragel, S. Parameswaran and S. Mohammad Kia, "Micro Embedded Monitoring for Security in Application Specific Instruction-set Precessors", *Proc. of Intl. Conf. on Compilers, Architectures and Synthesis for Embedded Systems*, San Francisco, 2005, pp. 304 - 314

[5] T. Zhang, X. Zhuang and S. Pande, "Anomalous Path Detection with Hardware Support", *Proc. of Intl. Conf. on Compilers, Architectures and Synthesis for Embedded Systems*, San Francisco, 2005, pp. 43 - 54

[6] D. Arora, A. Raghunathan, N. K. Jha, Architectural Support for Safe Software Execution on Embedded Processors, *Proc. of 4th Intl.Conf. on Hardware/Software Codesign and System Synthesis*, Seoul, 2006, pp. 106 - 111

[7] D. Arora, S. Ravi and A. A. Raghunathan, N. Jha, "Secure Embedded Processing through Hardware-assisted Run-time Monitoring", *Proc. of the Conf. on Design, Automation and Test in Europe - Volume 1*, Munich, 2005, pp. 178 – 183

[8] R. Wilhelm, "The Worst-Case Execution Time Problem – Overview of Methods and Survey of Tools", *ACM Transactions on Embedded Computing Systems*, vol. 7, no. 3, 2008

Control the Packets Transmission Using Quality of Service Protocol

Roxana Stănică and Emil Petre, *Member, IEEE*

***Abstract*—Quality of Service (QoS) refers to the mechanisms used for controlling and reserving network resources in order to provide different priority to specific applications/data flows and to guarantee their level of performance. This preferential treatment might be at the expense of other traffic flows. Without QoS, a switch offers best-effort service to each packet and transmits packets without any assurance of reliability, delay bounds (latency), or throughput (bandwidth). In this paper, an alternative QoS architecture based on a new Hybrid scheduling algorithm is proposed. This algorithm combines Strict Priority (SP) and Weighted Round Robin (WRR) scheduling. To apply and configure the new Hybrid scheduling algorithm, new commands were implemented.**

I. Introduction

QUALITY of Service (QoS) controls congestion by determining the order in which the packets are transmitted based on priorities assigned to them. QoS queuing policies can protect bandwidth for important categories of applications, or specifically limit the bandwidth associated with less critical traffic. QoS policies have little effect during periods of light traffic since packets are transmitted as soon as they arrive. They are effective for congestion, when a port cannot transmit all packets simultaneously and there is a need for defining the order in which the queued packets are transmitted.

QoS is defined in [3] as the set of service requirements to be met by a network while it carries a stream.

The QoS technology [7] prioritize the delay sensitive data traffic, which means that the voice and multimedia applications have priority over web-browsing. This provides a better quality voice and enables applications such as video and multimedia to run smoothly [1]. QoS Monitoring involves tracing levels (parameters) for an application and compares them with those required.

II. Basic QoS Architecture

In Fig. 1 it is shown how QoS affects traffic flow during the routing process [8].

On ingress pipe, the traffic is remarked according to the rate limit (detailed in Rate Limit section).

On egress pipe, traffic is:

• Distributed into eight priority queues according to internal priority and drop precedence (color). An alternative queuing mechanism, Tail-Drop, can be used for configuring the queuing process. Up to eight tail-drop profiles can be configured.

• Transmitted according to a queuing algorithm: Strict Priority (SP), Weighted Round Robin (WRR), or the new developed Hybrid scheduling algorithm.

QoS processing is divided in ingress and egress pipe units (see Fig. 1):

• (Egress) QoS Enforcement: utilizes eight egress queue-priorities per port. Congestion avoidance and congestion resolution techniques are used to provide the required service.

• (Egress) QoS Initial Marking: associates every packet with a set of QoS attributes that determine QoS processing by subsequent stages. Potentially, all types of packets - data, control, and mirrored to analyzer port - are subject to egress QoS initial marking.

• (Egress) Setting the Packet Header QoS Fields: the packet header 802.1p User Priority and/or IP-DSCP (Differentiated Services Code Point) is defined or modified. DSCP replaces the IP precedence - a 3-bit field in the Type of Service byte (see Fig. 3) of the IP header originally used to classify and prioritize types of traffic. (For more information, also refer to QoS Standards section.)

• (Ingress) Traffic Policing and QoS Remarking: if enabled on a policy-based traffic flow, and if the packet is classified as data, the given flow is measured according to a configurable rate limit that classifies packets as either in-profile or out-of-profile. Out-of-profile packets may be discarded or have their QoS attributes remarked.

• (Ingress) QoS Initial Marking: associates every packet classified as data with a set of QoS attributes that determine the QoS processing by subsequent stages. The sequence of the markers is shown in Fig. 1.

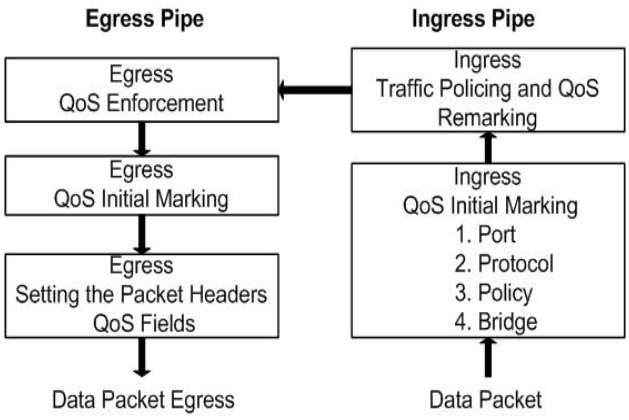


Fig.1. Basic QoS architecture

R. Stănică is with the S. C. BATM Systems S.R.L., Craiova, Romania (corresponding author; e-mail: rstanica@batmsystems.ro).

E. Petre is with the Department of Automatic Control, University of Craiova, Romania (e-mail: epetre@automation.ucv.ro).

A. Traffic Analysis

To effectively configure QoS, first it must to analyze the types of traffic [5] used by the port and determine their relative bandwidth demands. Also it must to evaluate the supported applications' sensitivity to:

• *Delay/latency*: the time a packet takes before it reaches its destination.

• *Jitter*: the variation of delay/latency that can affect the quality of streaming audio and/or video.

• *Packet loss*: the routers may fail to deliver some packets if they arrive, when their buffers are already full. Some, none, or all of the packets may be dropped, depending on the state of the network. The receiving application might ask for this information to be retransmitted, causing severe delays in the transmission.

The traffic types are [8]:

• Voice: demands small amounts of bandwidth. However, the bandwidth must be constant and predictable because voice applications are sensitive to latency (inter-packet delay) and jitter.

• Video: similar to voice application but requires larger bandwidth, depending on the encoding. Some applications can transmit large amounts of data for multiple streams in one spike or burst, causing the switch to buffer significant amounts of sent video-stream data.

• Database (DB): does not demand significant bandwidth and is tolerant to delay. Therefore, it requires minimum bandwidth and can be set to use lower priority.

• Web-browsing: cannot be generalized into a single category. Most browser-based applications have an asymmetric dataflow (small dataflow from the client's browser and large dataflow from the server to the client). An exception to this pattern might be created by some Java-based applications. Web-based applications are generally tolerant of latency, jitter, and some packet loss.

• File server: has the greatest demand on bandwidth, although it is tolerant to latency, jitter, and some packet loss, depending on the network operating system and the use of TCP (Transmission Control Protocol) or UDP (User Datagram Protocol).

B. Implementation

The typical QoS model (see Fig. 1) is based on the following:

• At the network edge (ingress), the packet is assigned to a QoS service. The service is assigned based on the packet header information (if the packet is trusted) or on the ingress port configuration (if the packet is untrusted).

• The QoS service defines the packet internal QoS handling (Class of Service - CoS and drop precedence - Color) and optionally the packet external QoS marking, through either the 802.1p User Priority and/or the IP header DSCP field (detailed in Packet's QoS Attributes section).

• A switch may modify the assigned CoS if a packet stream exceeds the configured profile. In this case, the packet may be dropped or reassigned to a lower CoS.

C. Tail-Drop Configuration

Congestion avoidance techniques ensure the monitoring traffic tasks in a network, also ensure the congestion avoidance by initiating the type of packet dropping (sending packets are dropped). One of the techniques is the *tail-drop* mechanism. When the queues reach their maximum length, this mechanism drops the recently received packets until the congestion is eliminated and the queue is not full. Tail-drop treats all traffic flows equally and does not differentiate between classes of service.

D. Packet's QoS Attributes

Every packet classified as data has assigned a set of QoS attributes that can be modified by each ingress pipeline.

The ingress pipeline contains several initial QoS markers that assign the packet initial QoS attribute. The ingress pipeline also contains a QoS remarker that can modify the initial QoS attributes. The packet's QoS attributes are:

• *QoS Precedence*: a switch can incorporate multiple QoS markers operating in a sequence. As a result, a later marker overrides an earlier QoS attribute assignment.

• *QoS Profile Index*: is used as a direct index, from 0 to 127, into the global QoS Profile table. Each entry in the QoS Profile table contains the set of attributes:

o TC/FC: Traffic Class/Forwarding Class assigned to the packet. One of the following traffic classes can be used: be (Best-Effort), 12 (Low-2), af (Assured), 11 (Low-1), h2 (High-2), ef (Expedited), h1 (High-1), nc (Network Control).

o DP: Drop Precedence (color) assigned to the packet.

o UP: If the packet's QoS attribute <Modify UP> is set and the packet is received untagged, this field is the value used in the packet 802.1p User Priority field. If the switch receives a tagged packet, the existing User Priority is modified with this value.

o DSCP: if setting the packet's QoS attribute to <Modify DSCP>, and the packet is IPv4 or IPv6, this field is the value used to modify the packet IP-DSCP field.

o QoS profiles 0-15 are used for all types of services.

• *Modify DSCP*: enables Packet DSCP field when the packet egresses the switch.

o 0 = Packet DSCP field is not modified when the packet egresses the switch

o 1 = Packet DSCP field is modified to the <DSCP> value of the QoS profile entry for the packet QoS Profile index.

• *Modify User Priority*: enables packet 802.1p-User Priority field modification.

o 0 = Packet User Priority field is preserved when the packet egresses the switch

o 1 = Packet User Priority field is modified to the <UP> value of the QoS Profile entry for the packet QoS Profile index, when the packet egresses the switch.

• *Default User Priority*: is assigned by the ingress port configuration, only when the <Modify UP> is cleared and the packets are received untagged.

III. QoS Standards

QoS implementation complies with the IEEE 802.1p and IETF-DiffServ standards.

The *IEEE 802.1p standard* [9], also known as Class of Service (CoS), is a 3-bit field within an Ethernet frame header when using tagged packets on an 802.1 network. This standard specifies a priority value between 0 and 7

TABLE I
UNITS FOR MAGNETIC PROPERTIES

User priority	Traffic Type
0 (lowest)	Best Effort
1	Background
2	Standard (Spare)
3	Excellent Load (Business Critical)
4	Controlled Load (Streaming Multimedia)
5	Voice and Video (Interactive Media and Voice) [Less than 100 ms latency and jitter]
6	Layer 3 Network Control Reserved Traffic [Less than 10 ms latency and jitter]l
7 (highest)	Layer 2 Network Control Reserved Traffic [Lowest latency and jitter]

that can be used by Quality of Service (QoS) protocol to differentiate traffic. In Table I are shown the IEEE QoS priority levels.

Differentiated Services (DiffServ) [10] is a multiple service model that can satisfy different QoS requirements. However, an application using DiffServ does not explicitly signal the switch before sending data.

For differentiated services, the network tries to deliver a particular kind of service based on the QoS specified by each packet. This specification can occur in different ways; for example, using the IP Precedence bit or the 6-bit Differentiated Services Code Point (DSCP) setting in IP packets, or source and destination addresses.

Fig. 2 shows the location of the ToS octet within the IPv4 packet header.

Fig. 3 shows the IP ToS octet fields. The 3 precedence bits have a value from 0 to 7 and are used to indicate the importance of a datagram. Bits 3, 4, and 5 represent the following:

- D: requests low delay
- T: requests high throughput
- R: requests high reliability

IV. QOS DECISION FLOW

In Fig. 4 it is shown the order of decision once the packets enter the pipeline [8]. In case of conflicting configurations, the first box shows which configuration takes precedence. Priority remarking and mapping are available only for 802.1p packets.

V. TRAFFIC SCHEDULING

Traffic Scheduling allows the control of packets transmission, based on priorities assigned to those packets. Congestion management determines the creation of queues, the assignment of packets to the queues that are based on the packet's classification, and scheduling of the packets in a queue for transmission.

The packets are scheduled for transmission according to their assigned priority and their queuing algorithm. The switch determines the order of packets transmission by controlling which packets are placed in which queue and the order in which the queues are serviced.

The QoS traffic behavior can be controlled by selecting the queuing algorithm to be applied to the outbound queues (8 queues are used).

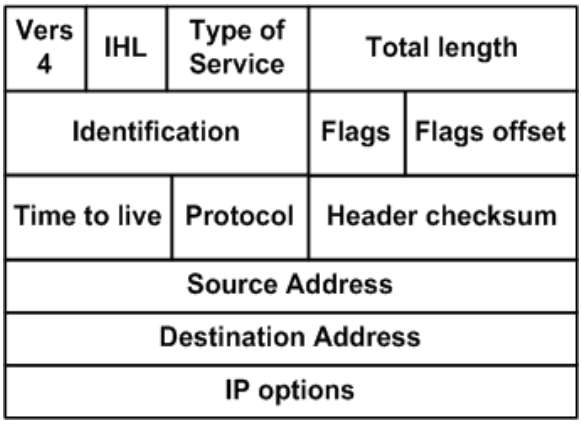


Fig. 2. IPv4 Header structure

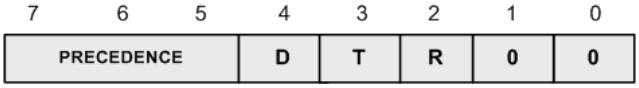


Fig. 3. ToS octet fields

Three queuing algorithms can be used:

- Strict Priority (SP)
- Weighted Round-Robin (WRR)
- Hybrid scheduling. A new Hybrid queuing algorithm that combines SP and WRR is proposed.

A. Strict Priority (SP)

SP [8] provides preferential treatment to traffic with high priority (lowest value), making sure that mission-critical traffic gets priority treatment.

SP Algorithm:

- The highest ranking queue, txq8, is serviced first until it is empty,

then

- The lower queue, txq7, is serviced and so on, down to txq1.

SP provides a faster response time for high priority traffic than other methods of queuing. The SP algorithm guarantees a fixed portion of available bandwidth to an application (e.g., interactive multimedia applications), possibly at the expense of less critical traffic.

When selecting SP, the lower priority traffic is often denied in favor of higher priority traffic. In the worst case, lower priority traffic is never transmitted. However, these scenarios can be avoided by using the Rate Limit (detailed in Rate Limit section) to control higher-priority traffic rate. Rate limit controls congestion on service provider networks, and ensures proper use of bandwidth resources.

Fig. 5 illustrates the SP process in four-queue architecture. The incoming packets are classified with a high transmission priority (e.g., 1), and are transmitted through a queue with high priority (e.g., txq8).

B. Weighted Round Robin (WRR)

WRR [8] is a scheduling algorithm that cycles through the queues. A weighting factor determines how many bytes of data the system delivers from each queue before moving to the next queue.

WRR Algorithm:

- The packets in the queue are sent until the number of bytes to be transmitted exceeds the bandwidth or until the

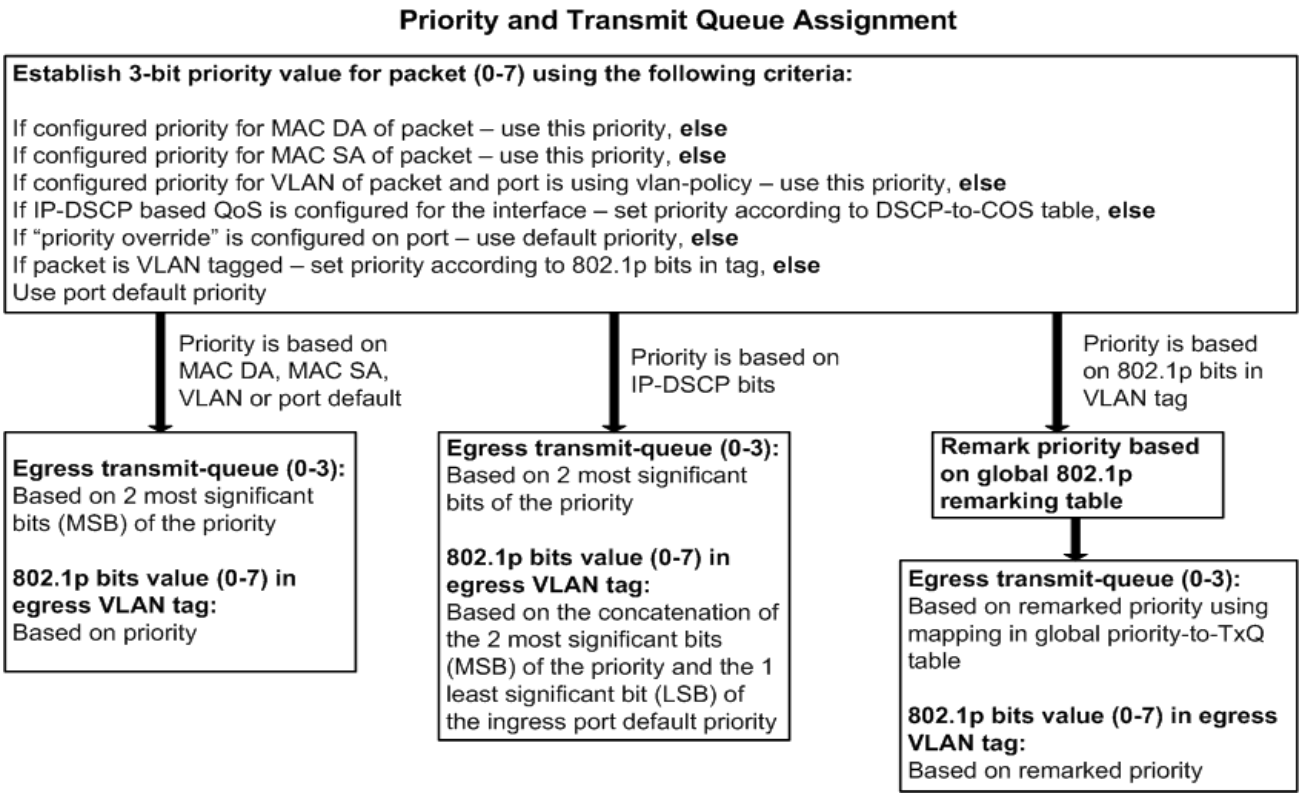


Fig. 4. Priority and transmit queue assignment

queue is empty (only then WRR moves to the next queue).

• If a queue is empty, the switch sends packets from the next queue that has packets to send.

• If a packet's length exceeds the queue's allowed bandwidth, the packet is still transmitted during its time slot, but its quota is overdrawn so next time it receives a smaller allocation.

This algorithm guarantees a minimum bandwidth for each queue, but allows the minimum to be exceeded if one or more of the port's other queues are idle.

Fig. 6 illustrates the WRR queuing in four-queue architecture. The incoming packets are classified with an average transmission priority (e.g., 3), and are transmitted through a queue with medium priority (e.g., txq4).

C. Hybrid Scheduling

In this section a new hybrid scheduling algorithm that combines SP and WRR algorithms is developed.

Hybrid Algorithm:

• The queues with higher priority are serviced with SP;

• The remaining queues are serviced in accordance with WRR, after the higher priority queues are empty.

Hybrid queuing guarantees immediate delivery of packets from high-ranking queues while avoiding lowest-ranking queues.

VI. Rate Limit

Rate limiting is performed by policing (discarding excess packets), queuing (delaying packets in transit) or

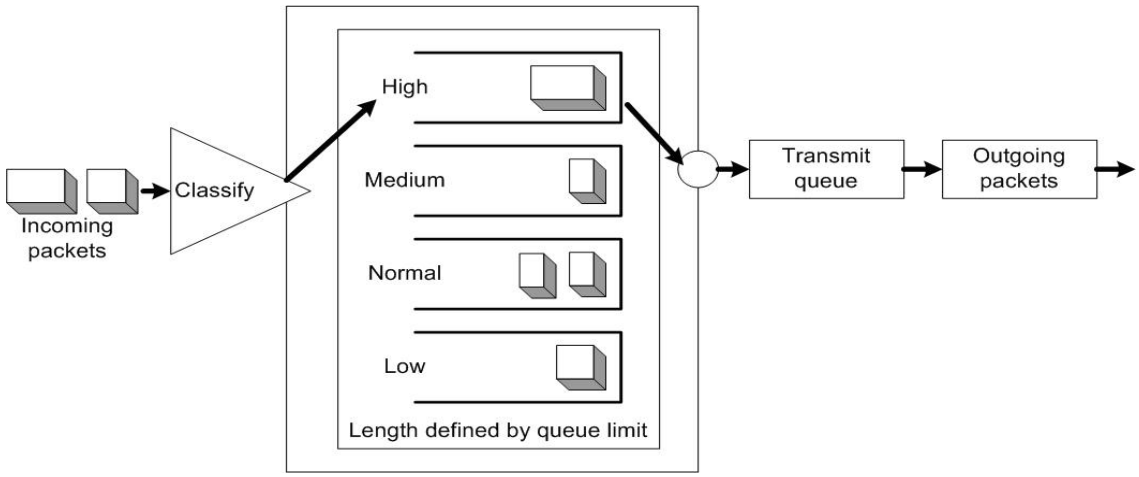


Fig. 5. Strict Priority queuing

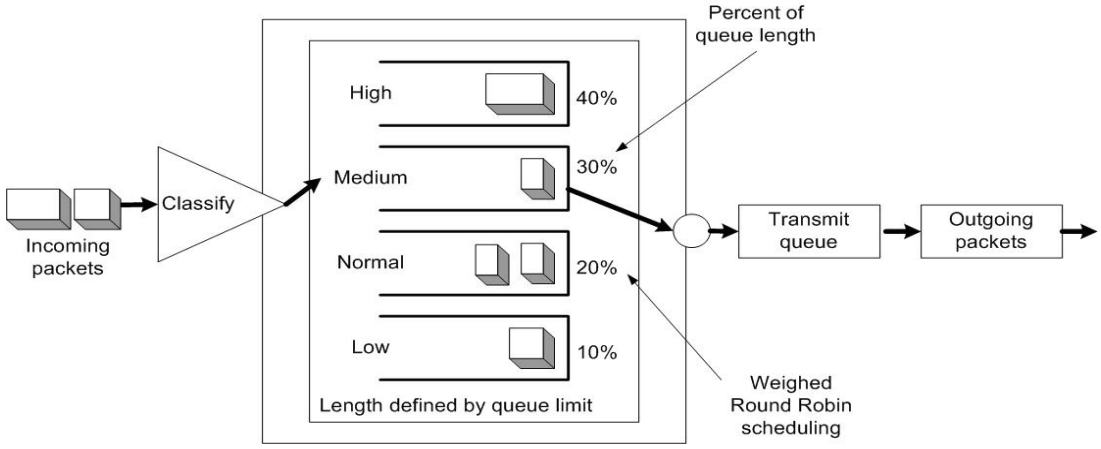


Fig. 6. Weighted Round Robin queuing

congestion control (manipulating protocol's congestion mechanism). Traffic congestion, caused by heavy network traffic, can cause incoming packets to be dropped.

A traffic rate limiter monitors the incoming traffic by:

• forwarding conforming traffic (within the predefined rate);

• dropping non-conforming traffic or marking this traffic.

A. *Single Rate Three Color Marker (RFC 2697)*

The Single Rate Three Color Marker (srTCM) [4] measures the traffic stream and marks it according to two parameters:

• The Committed Information Rate (CIR) determines the long-term average transmission rate;

• The Committed Burst Size (CBS) determines how large traffic bursts can be before some of the traffic exceeds the rate limit.

The traffic is then marked as follows:

• Traffic within CIR always conforms and is marked green;

• Traffic that exceeds CBS is dropped or marked yellow.

B. *Exceed Action*

Once the packet is classified as exceeding a particular rate limit, the switch:

• either drops the packet;

• marks the packet with yellow color and continue.

C. *Color-Blind and Color-Aware*

Rate limiting operates in one of the two modes:

• Color-Blind mode: assumes that the packet stream is uncolored;

• Color-Aware mode: assumes that some preceding entity has pre-colored the incoming packet stream, so each packet can be colored green or yellow.

VII. New Scheduling Commands

Fig. 7 details the steps required to configure the QoS parameters [8]. To apply and configure the new Hybrid scheduling algorithm, the following commands were implemented:

• scheduling-profile sp <*profile_number*> command uses the Strict Priority (SP) scheduling.

o *profile_number*: the scheduling profile ID. The range is <1–8>.

• scheduling-profile wrr <*profile_number*> <*txq1*> <*txq2*> <*txq3*> <*txq4*> <*txq5*> <*txq6*> <*txq7*> <*txq8*> command applies and configures Weighted Round-Robin (WRR) scheduling.

o <*txq1*>…<*txq8*>: the number of bytes assigned to each of the eight queues. The range is <1–255> bytes.

o In WRR scheduling, bandwidth is allocated proportionally for each queue. Network resources are shared among all of the applications the user services, each having the specific bandwidth requirements that can be identified.

• scheduling-profile hybrid-1 <*profile_number*> <*txq1*> <*txq2*> <*txq3*> <*txq4*> <*txq5*> <*txq6*> <*txq7*> command applies and configures the first Hybrid QoS algorithm. In the first Hybrid algorithm, *txq8* behaves according to SP scheduling, and the rest of the queues behave according to WRR scheduling.

• scheduling-profile hybrid-2 <*profile_number*> <*txq1*> <*txq2*> <*txq3*> <*txq4*> <*txq5*> <*txq6*> command applies and configures the second Hybrid QoS algorithm. In the second Hybrid algorithm, *txq7* and *txq8* behave according to SP scheduling, and the rest of the queues behave according to WRR scheduling.

• rate-limit <*cir*> <*cbs*> [color-aware | exceed-action mark-yellow] command applies and configures a rate limit for all ports of the switch.

o *cir*: the Committed Information Rate in K, M, or G (in bits/second - bps). The range is <64 K - 1 G> bps.

o *cbs*: the Committed Burst Size in K, M or G (in bytes). The range is <4 K–16384 K> bytes.

o color-aware: (optional) the rate limit is in color-aware mode. If this option is not specified, the default is assumed: the color marking of the packet is ignored (color-blind).

o exceed-action: (optional) the action performed once the packet is classified as exceeding the CIR. If this is not specified, the out-of-profile traffic is dropped.

o mark-yellow: marks in yellow the packet classified as exceeding the CIR. If this is not specified, the out-of-profile traffic is dropped.

• show scheduling-profile [<*profile_number*>] command displays the scheduling profile configuration for

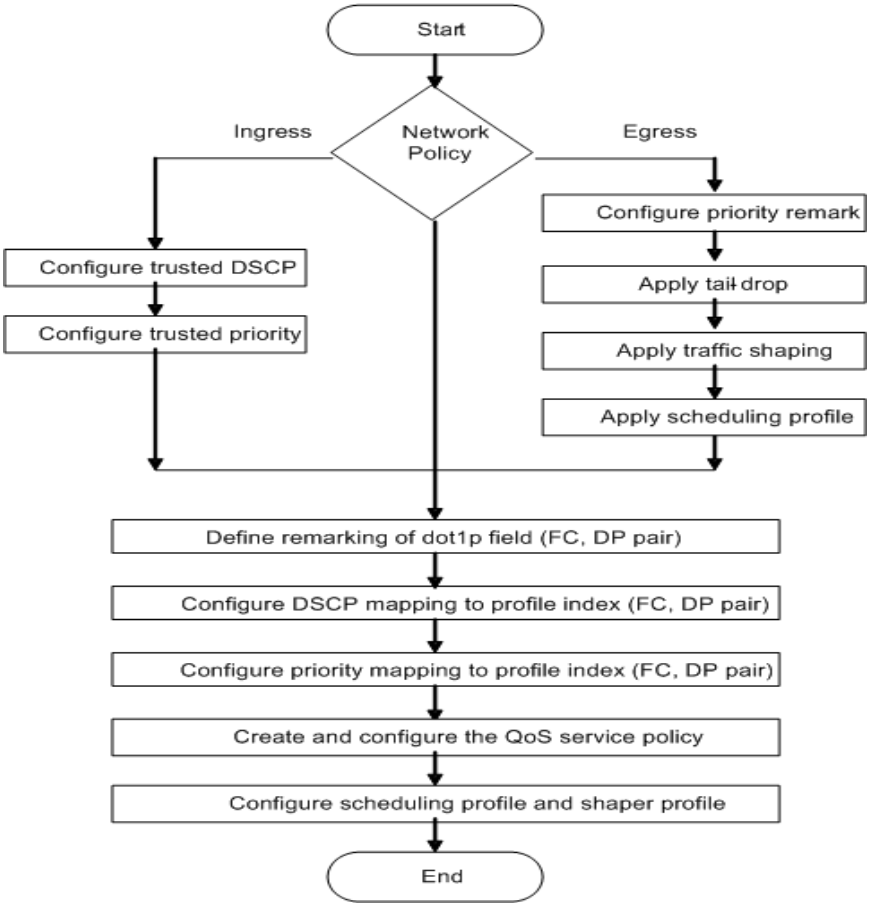


Fig. 7. QoS flow

all profiles or for the specified profile ID.

o *profile_number*: (optional) the scheduling profile ID, in the range of <1–8>. If the profile ID is not specified, all scheduling profiles are displayed.

VII. Concluding Remarks

Quality of Service provides different priorities to different applications, users, or data flows, or guarantees a certain level of performance to a data flow. For example, a required bit rate, delay, jitter, packet dropping probability, and/or bit error rate may be guaranteed.

Requirements for different types of packet traffic are specified through the QoS constraints [3]:

- constraints that accounts (additive);
- constraints that are breeding (multiplicative);
- constraints that are selected according to the smallest or bigger value (concave or convex).

QoS routing finds a path from a source to a destination and satisfies certain specified constraints. QoS routing consists of two phases:

- collecting and updating/maintaining the information required for QoS routing process;
- searching for possible ways, based on information collected in the first phase.

Traffic Scheduling allows controlling the packets transmission - based on priorities assigned to packets - and selecting a queuing algorithm. To improve the QoS scheduling, a queuing algorithm can be applied. A new Hybrid algorithm that combines SP and WRR was developed, and new scheduling commands were implemented.

References

[1] A. Campbell and G. Coulson, "A QoS adaptive multimedia transport system: design, implementation and experiences," *J. Distributed Systems Engineering*, vol. 4, no. 1, pp. 48-58, 1997.

[2] C.-N. Chuah, L. Subramanian, R. H. Katz, and A. D. Joseph, "QoS provisioning using a clearing house architecture," *IEEE/IFIP International Workshop on Quality of Service (IWQoS)*, Pittsburgh, Pennsylvania, pp. 115-124, June 2000.

[3] E. S. Crawley, R. Nair, B. Rajagopalan, and H. Sandick, "A Framework for QoS-based Routing in the Internet," RFC Editor, USA, Aug. 1998, pp. 4-14.

[4] J. Heinanen, T. Finland, and R. Guerin, "A Single Rate Three Color Marker", Category: Informational, University of Pennsylvania, pp. 2-5, Sept. 1999.

[5] M. J. Karam and F. A. Tobagi, "Analysis of delay and delay jitter of voice traffic in the internet," *The International Journal of Computer and Telecommunications Networking*, vol. 40, no. 6, pp. 711-726, Dec. 2002.

[6] F. A. Kuipers, T. Korkmaz, M. Krunz, and P. Van Mieghem, "An Overview of Constraint-Based Path Selection Algorithms for QoS Routing," *IEEE Communications Magazine*, *Special Issue on IP-Oriented Quality of Service*, vol. 40, no. 12, pp. 50–55, Dec. 2002.

[7] T. Radulescu and H. G. Coanda, *QoS in IP multimedia networks*, Editura Albastra, Cluj-Napoca, 2007.

[8] http://support.batm.com/products.php

[9] IEEE 802.1p standard, http://www.tml.tkk.fi/Opinnot/Tik-110.551/1999/papers/08IEEE802.1QosInMAC/qos.html

[10] IETF DiffServ Working Group, http://www.linktionary.com/d/diffserv.html

Hybrid Control Scheme Implemented on a Programmable Logical Controller

Florin Stinga, Dan Mitoiu, Ionut Nisipeanu, and Andreea Soimu

Abstract— **In this paper is presented a hybrid control scheme for a two-tank plant. The control scheme combine a classical controller (proportional-integrator controller) and a controller based on logical rules, taking in consideration the physical restrictions imposed on the control input and plant outputs. The hybrid control scheme was implemented on a PLC (programmable logical controller), using ladder diagrams. Some experimental results are presented in order to illustrate the performance of the proposed control method.**

I. Introduction

HYBRID systems are systems that involve interaction between discrete and continuous dynamics. Such systems have been studied with growing interest and activity in recent years. For hybrid systems it is needed to consider dynamical elements and their mutual relations altogether for exact analyze and optimize a process [1]. A hybrid control system is a control system with analog and digital parts. Such a system generates a mixture of continuous and discrete signals. An example of a hybrid control system is a switching system where the dynamic behavior of interest can be adequately described by a finite number of dynamical models that are typically sets of differential or difference equations, together with a set of rules for switching among these models.

In this paper, the problem of modeling, control and implementation of control scheme for two-tanks plant it is approached.

Using the classic control for MIMO systems when the number of outputs is larger than number of inputs, it is hard to control each of the measured outputs independently. Instead, a hybrid control scheme offers a good performance tracking for each of outputs.

Hybrid control of a double tank system has been treated over time in several papers [2, 3 and 4].

The paper is organized as follows: in Section II the two-tanks plant is described and a nonlinear mathematical model is obtained. The structure of the hybrid control system is proposed in Section III. The mathematical model of the extended hybrid system is formulated including the transitions conditions. In section IV the proposed control scheme is implemented on a PLC, using ladder diagrams. Finally, some experimental results are presented in order to illustrate the performance of the proposed control scheme.

This work was supported in part by the grant no. POSDRU/6/1.5/S/14.

Florin Stinga is a PhD student in Department of Automatic Control, University of Craiova, 13 A.I. Cuza str., 200585 Craiova, Romania. (e-mail: florin@ automation.ucv.ro).

Dan Mitoiu is Master Student at Faculty of Automation, Computers and Electronics, University of Craiova, 13 A.I. Cuza str., 200585 Craiova, Romania. (e-mail: mitoiudan@yahoo.com).

Ionut Nisipeanu is Master Student at Faculty of Automation, Computers and Electronics, University of Craiova, 13 A.I. Cuza str., 200585 Craiova, Romania. (e-mail: ionut_nisipeanu@yahoo.com).

Andreea Soimu is a PhD student in Department of Automatic Control, University of Craiova, 13 A.I. Cuza str., 200585 Craiova, Romania. (e-mail: andreea@ automation.ucv.ro).

II. Mathematical Model

The two tank plant is presented in Fig. 1, where: D_{t1}, D_{t2} are the inside diameters of the tanks; L_1, L_2 are the levels of the liquid in the two tanks; D_{o1}, D_{o2}, D_{o3} are the diameters of the output sections; E_1, E_2 are ON/OFF electro-valves; V_p is feed voltage of the pump P.

We consider the following nonlinear mathematical model for two tank system [5]:

$$\begin{aligned} dL_1/dt &= (1/A_{t1})\left[F_{i1}^q - F_{o1}(L_1) - F_{w1}(L_1)\right] \\ dL_2/dt &= (1/A_{t2})\left[F_{i2}^q + F_{o1}(L_1) - F_{o2}(L_2)\right] \end{aligned} \quad (1)$$

where: A_{t1} and A_{t2} are the inside areas of the tanks, F_{i1}^q and F_{i2}^q are the input flows, F_{o1} and F_{o2} are the output flows, F_{w1} is the perturbation flow and q is the discrete mode.

$$A_{ti} = \frac{\pi D_{ti}^2}{4}, \ i = 1, 2$$

The four discrete modes can be expressed by:

Mode 1 $(q=1)$: $E_1 = ON, E_2 = OFF$

Mode 2 $(q=2)$: $E_1 = OFF, E_2 = ON$

Mode 3 $(q=3)$: $E_1 = OFF, E_2 = OFF$

Mode 4 $(q=4)$: $E_1 = ON, E_2 = ON$

According with the four modes associated with the system, input flows are expressed by:

$$F_{i1}^q = \begin{cases} K_p V_p, & if \quad q=1 \\ 0, & if \quad q=2 \vee q=3 \\ K_p V_p/2, & if \quad q=4 \end{cases}, \ F_{i2}^q = \begin{cases} K_p V_p, & if \quad q=2 \\ 0, & if \quad q=1 \vee q=3 \\ K_p V_p/2, & if \quad q=4 \end{cases}$$

where: K_p is pump flow constant;

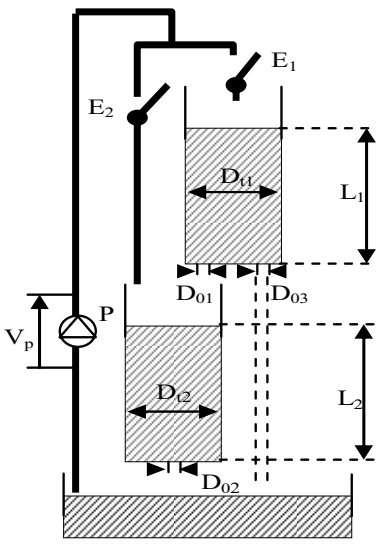


Fig. 1. Two-tank system

The output and perturbation flows are described by:

$F_{o1}(L_1) = A_{01}\sqrt{2gL_1}$, $F_{o2}(L_2) = A_{02}\sqrt{2gL_2}$,

$F_{w1}(L_1) = \alpha \cdot A_{03}\sqrt{2gL_1}$

where: A_{o1}, A_{o2}, A_{o3} are the cross-section areas of the outputs.

$$A_{oi} = \frac{\pi D_{oi}^2}{4},\ i = 1, 2, 3$$

and $\alpha \in [0, 1]$ is perturbation factor (we supposed that the perturbation flow is variable).

The steady state analysis shows that the maximum levels which can be obtained are expressed by:

$$L_1(V_{p\max}) = \begin{cases} K_p^2 V_{p\max}^2 / A_o^2\left(2g(1+\alpha)^2\right) & if \quad q = 1 \\ 0 \quad if \quad q = 2 \vee q = 3 \end{cases} \quad (2)$$

$$L_2(V_{p\max}) = \begin{cases} K_p^2 V_{p\max}^2 / A_o^2\left(2g(1+\alpha)^2\right) & if \quad q = 1 \\ K_p^2 V_{p\max}^2 / A_o^2 2g \quad if \quad q = 2 \\ 0 \quad if \quad q = 3 \end{cases} \quad (3)$$

Remark 1: We considered $A_{o1} = A_{o2} = A_{o3} \overset{\Delta}{=} A_o$. For the hybrid control system, only the mode 1, 2, and 3 are used.

From (2) and (3) we remark that $L_2(V_{p\max}) \geq L_1(V_{p\max})$, $\forall q \in \{1, 2, 3\}$.

For the nonlinear system (1), linearization around nominal point (L_{1N}, L_{2N}) leads to the following form:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B^q u^q(t) + f^q \\ y(t) &= Cx(t) \end{aligned} \quad (4)$$

where:

$x = [L_1\ L_2]^T$

$$u^q = \begin{cases} V_p & if \quad q = 1 \wedge q = 2 \\ 0 & if \quad q = 3 \end{cases},\ A = \frac{1}{A_t}\begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix},$$

$$a_1 = -0.5\left(A_o\sqrt{2g}L_{1N}^{(-1/2)}\right)(1+\alpha),\quad a_2 = 0,$$

$$a_3 = 0.5\left(A_o\sqrt{2g}L_{1N}^{(-1/2)}\right),\ a_4 = -0.5\left(A_o\sqrt{2g}L_{2N}^{(-1/2)}\right)$$

$$B^q = \frac{1}{A_t}\begin{cases} [K_p \quad 0]^T, & if \quad q = 1 \\ [0 \quad K_p]^T, & if \quad q = 2 \\ [0 \quad 0]^T, & if \quad q = 3 \end{cases},\ C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

$$f^q = \frac{1}{A_t}\begin{cases} \begin{bmatrix} K_p u_N^1 - A_o(1+\alpha)\sqrt{2gL_{1N}} \\ A_o\sqrt{2gL_{1N}} - A_o\sqrt{2gL_{2N}} \end{bmatrix} if\ q = 1 \\ \begin{bmatrix} -A_o(1+\alpha)\sqrt{2gL_{1N}} \\ K_p u_N^2 + A_o\sqrt{2gL_{1N}} - A_o\sqrt{2gL_{2N}} \end{bmatrix} \quad if\ q = 2 \\ \begin{bmatrix} -A_o(1+\alpha)\sqrt{2gL_{1N}} \\ A_o\sqrt{2gL_{1N}} - A_o\sqrt{2gL_{2N}} \end{bmatrix} if\ q = 3 \end{cases}$$

where: u_N^1 and u_N^2 are the nominal values for the mode $q = 1$, respectively $q = 2$ and:

$$u_N^1 = \frac{A_o(1+\alpha)\sqrt{2gL_{1N}}}{K_p},\ u_N^2 = \frac{A_o\sqrt{2gL_{2N}}}{K_p}$$

Remark 2: We considered $A_t = A_{t1} = A_{t2}$.

The system (4) can be rewrite in a more common form:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B^q v^q(t) \\ y(t) &= Cx(t) \end{aligned} \quad (5)$$

where: $v^q(t) = u^q(t) + (B^q)^+ f^q$ and, $(B^q)^+$ is the Moore-Penrose pseudoinverse of the B^q.

III. Hybrid Control System

We consider the graph representation of the hybrid system as in Fig. 2, where c_1 and c_2 are small constants that characterize the tolerances of the controlled levels.

Remark 3: From the transition conditions we remark that some priority is given to the level control in the tank 1.

In order to generate the command v^q, we use a discrete PI controller [6].

So, the command law can be expressed by:

$$v(k) = v(k-1) + (k_R + k_I)e(k) - k_R e(k-1) \quad (6)$$

where: k_R and k_I are given by:

$$k_R = \frac{p_1 - \beta - a_1 + 1}{b_1},\ k_I = \frac{-\beta p_2 + k_p b_0 + a_0}{b_0}$$

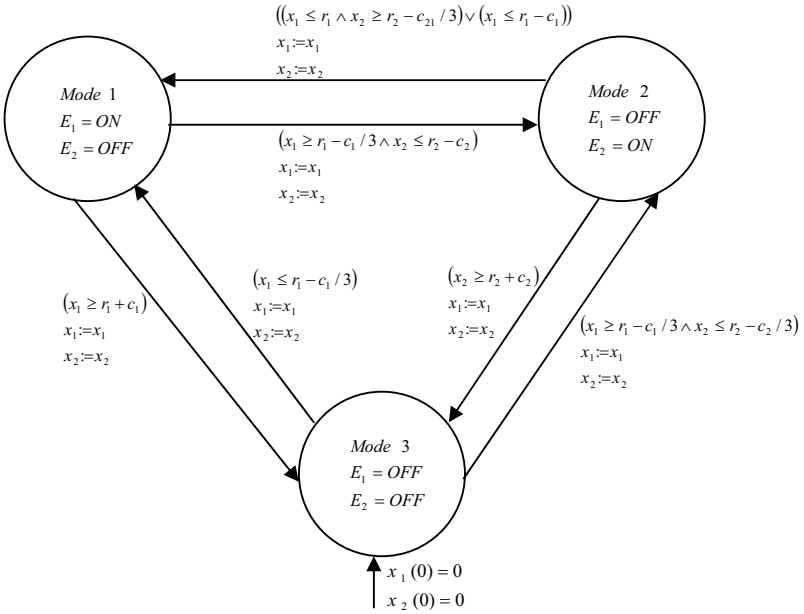


Fig. 2. Hybrid system

$p_1 = -2\exp^{-\xi\omega T}\cos\left(\omega T\sqrt{1-\xi^2}\right)$, $\beta = \exp^{-\delta\omega T}$, $\delta = 5 \div 10$

$p_2 = \exp^{-2\xi\omega T}$

where:

- b_0, b_1, a_0, a_1 are the coefficients of the equivalent transfer function of zero holder discretization of the system (5):

$$H(z) = \frac{b_1 z + b_0}{z^2 + a_1 z + a_0}$$

- T is sampling time, ξ and ω have desired values for closed loop performance of the system., e is the error of the system.

Remark 4: The PI command is computed only for mode 1 and 2. For mode 3 $v(k) = 0, \forall k$.

IV. Implementation of Control Scheme

The hybrid control scheme was implemented on an Allen Bradley controller, more exactly a 1768-L43 CompactLogix controller.

CompactLogix controller is designed to provide solutions for medium applications. The controller has 2 Mb of memory and offers one built-in serial port.

CompactLogix offers common programming environment, modules (analog, digital and dedicated) that cover a wide range of applications and advanced system connectivity to EtherNet/IP. It is ideal for systems that require standalone and system-connected control over EtherNet/IP.

The controller supports a maximum of 16 local 1769 I/O modules [7].

The software used for program development is RSLogix 5000, a software package for discrete, process, batch, motion, safety and drive-based applications. It offers an easy-to-use interface, symbolic programming with structures and arrays and a comprehensive instruction set that serves many types of applications.

It provides ladder logic, structured text, function block diagram and sequential function chart editors for creating control applications [8].

The process – PC interface (Fig. 3) is a LAN network where the computer is connected with the controller and a touch screen monitor through a switch.

The touch screen monitor is a PanelView Plus terminal from Rockwell and it gives operators a clear view into monitoring and controlling applications.

The CompactLogix L43 Controller commands the water pump trough the analog output and reads the voltage given by the pressure sensors, used to measure the level of water in the tanks, via an Analog I/O Module situated on the 1769 Bus. The control of the solenoid valves is done using a Digital Output Module situated on the same bus.

The control program is situated in the memory of the controller.

All the information about the process and command is passed in the local network to the PanelView Plus and the Computer using TCP-IP protocol.

The HMI (humane machine interface) application can run parallel on the Computer and PanelView Plus.

Levels can be viewed in real time on both the Computer and PanelView Plus.

The control program can be synthesized using a logical representation (Fig. 4), as follows:

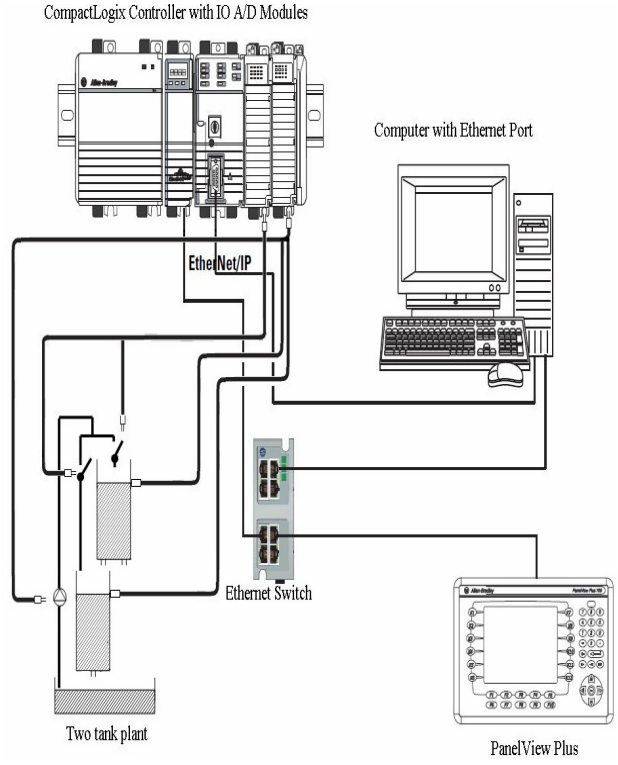


Fig. 3. Process – PC interface

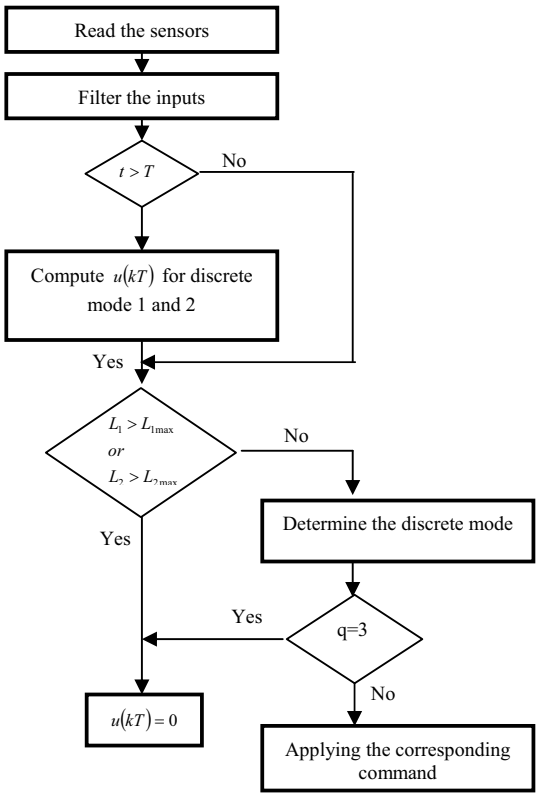


Fig. 4. Logical representation of control program

The first step is data acquisition and filtering, then we compute de commands with the fixed sampling time, followed by discrete mode determination and applying the corresponding command.

A necessary test is performed in order to determine if the levels were reached their maximum values $(L_{1\max,}\ L_{2\max})$.

Our control scheme was implemented using ladder diagram and some routines were written in structured text.

The program is structured on three major parts:

- Initialization – the sampling time is set using a timer and the inputs and output are scaled;

- States implementation – the conditions for each state are specified;

- Alarms – is set the maximum level for the two tanks and in case of malfunction the plant will have a forced stop preventing any other damages.

The main program includes also three routines written in structured text, one for filtering the sensors outputs and one used to calculate the pump command in each state.

V. Experimental Results

The following parameters were used [9]:

$$A_{o1} = A_{o2} = 0.17\,cm^2,\quad A_{t1} = A_{t2} = 15.51\,cm^2,$$
$$g = 981\,cm/s^2,\ K_p = 3.3\,cm^3/s/V,\quad \alpha = 0.5$$

The linearized system (5) around nominal point $(L_{1N} = 5, L_{2N} = 10)$ is:

$$\begin{cases} \begin{bmatrix} \dot{L}_1 \\ \dot{L}_2 \end{bmatrix} = \begin{bmatrix} -0.17 & 0 \\ 0.11 & -0.08 \end{bmatrix} \begin{bmatrix} L_1 \\ L_2 \end{bmatrix} + B^q v^q \\ y = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} L_1 \\ L_2 \end{bmatrix} \end{cases}$$

where:

$$B^q = \begin{cases} [0.21 \quad 0]^T, & if \quad q = 1 \\ [0 \quad 0.21]^T, & if \quad q = 2 \\ [0 \quad 0]^T, & if \quad q = 3 \end{cases},\ f^q = \begin{cases} \begin{bmatrix} 0 \\ -0.47 \end{bmatrix} & if \quad q = 1 \\ \begin{bmatrix} -1.13 \\ 1.13 \end{bmatrix} & if \quad q = 2 \\ \begin{bmatrix} -1.13 \\ -0.47 \end{bmatrix} & if \quad q = 3 \end{cases}$$

The command applied to the plant is given by:

$$u^q = \begin{cases} v^1 & if \quad q = 1 \\ v^2 - 5.41, & if \quad q = 2 \\ 0, & if \quad q = 3 \end{cases}$$

where v^1 and v^2 are mode 1 and 2 commands, generated by the discrete PI controller (6), with the following input parameters:

$$T = 0.5\text{sec},\ \xi = 0.7,\ \omega = 1 rad/\text{sec},\ \delta = 5$$

The parameters values of the PI controller, computed as described in Section III are:

$$\begin{cases} k_R = 14.48, \quad k_I = 5.89, \quad if \quad q = 1 \\ k_R = 14.17, \quad k_I = 5.37 \quad if \quad q = 2 \end{cases}$$

For hybrid system we consider: $c_1 = 1cm, \quad c_2 = 0.5cm$.

Additionally, based on plant constraints, we define the limits for the levels and command:

$$L^q_{\min} = 0\,cm,\ L^q_{\max} = 25\,cm, \forall q = 1,2,3$$
$$u^q_{\min} = 0\,V,\ u^q_{\max} = 10.5\,V, \forall q = 1,2$$

The filters used in filtering process are low-pass filters.

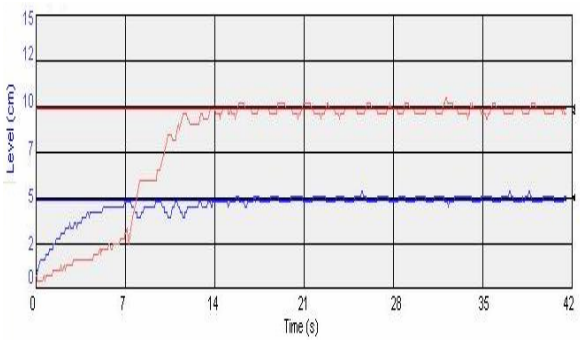


Fig. 5. Two tanks plant levels

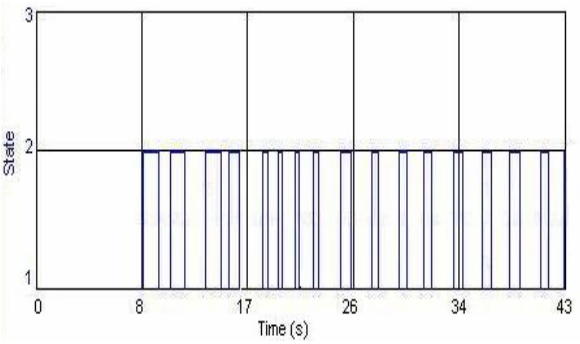


Fig. 6. Discrete transitions

In Fig. 5 it is presented the levels evolution in the two tanks. In figure 6 it is presented the discrete transitions.

VI. Conclusions

In this paper, the problem of modeling, control and implementation of control scheme for two tank plant it is approached.

A hybrid control scheme offers a good performance tracking for each of outputs of a MIMO system, even if the number of outputs is larger than number of inputs.

The control scheme combines a classical control (PI control) and a logical based control. The proposed control scheme was implemented on PAC, using ladder diagrams.

Finally, were presented some experimental results, in order to demonstrate the performance of the proposed hybrid control scheme.

References

[1] P. J. Antsaklis, X. D. Koutsoukos, "Hybrid system control", in *Encyclopedia of Physical Science and Technology*, volume 7, pp. 445-458. Academic Press, third edition, 2002.

[2] J. Malmborg, J. Eker, "Hybrid control of a double tank system", in *6th IEEE Conference on Control Application*, Hartford, Connecticut, 1997.

[3] A. Borshchev, Y. Karpov, V. Kharitonov, "Distributed Simulation of Hybrid Systems with AnyLogic and HLA", in *Future Generation Computers Systems,* 18(6), pp. 829-839, 2002.

[4] T. Hirmajer, M. Fikar, "Optimal Control of a Hybrid Dynamical System: Two Coupled Tanks", Technical Report, TH0001, 2002

[5] F. Stinga, A. Soimu, D. Popescu, "Hybrid Control system for a Two-Tank Plant", in *Proc. 11th Int. Carpathian Control Conf*., Eger, 2010, pp. 451-454.

[6] C. Lazar, D. Vrabie, S. Carari, „Automated system with PID controllers (in Romanian)", in Ed. Matrix Rom, Bucharest, 2004.

[7] G Allen-Bradley / Rockwell Automation. 1768 CompactLogix Controllers – User manual. Available: www.ab.com

[8] Allen-Bradley / Rockwell Automation. RSLogix 5000. Available: www.ab.com

[9] Quanser Consulting Inc. Two tanks system. 1998.

Questions generation management system for e-learning

Cosmin Stoica Spahiu
University of Craiova, Faculty of Automation, Computers and Electronics
Email: Stoica.Cosmin@software.ucv.ro

Abstract— **The e-learning platforms became very useful tools in the last years for the students' education in different activity domains. In this paper it is presented the current implementation stage of an e-Learning platform designed as a collaborative environment for students, professors, secretaries and administrators. It is a role based platform, where each role can perform specific tasks. An element of novelty for this platform is that professors can define tags in the courses they have added and the system will automatically generate questions. The accomplished studies indicated that the students substantially appreciate the e-learning method, due to the facilities: the facile information access, a better storage of the didactic material, the curricula harmonization between universities, personalized instruction.**

I. Introduction

TAKING into consideration the high number of learning material existing in electronic format, the importance of the testing and evaluation systems has increased in the past decade. The accomplished studies indicated that the students substantially appreciate the e-learning method, due to the facilities offered [1], [2], [3], [6], [7]: the facile information access, a better storage of the didactic material (the curricula harmonization between universities, personalized instruction, informational content standardization, real time access to qualitative information resources, friendly interfaces) but they don't consider it as a replacement of the traditional learning which has other advantages [8].

An essential aspect in the learning process (either electronic or traditional) is the possibilities to evaluate the students and to provide them electronic material. Most of these systems use tests that were generated by teachers. These tests permit a good evaluation and pursuance of the student evolution.

One of the domains where the platform presented in this paper is intended to be used is the medicine domain where the images obtained during investigations plays an important role (investigations from endoscopy, echography, tomography, etc.).

As the examples presented during classes are reduced to minimum, the professors must have the possibility to add different images with different diagnosis and the students should have the possibility to retrieve these images either based on their diagnosis (text-based retrieval), either based on their similitude (content-based retrieval). This way they can compare different situations where images are somehow different but have the same diagnosis (what are the details needed to obtain the diagnosis) or images similar but with different diagnosis (what makes the differences).

These operations will be implemented based on the color, and texture characteristics similarity.

Another aspect very important for the professors is to have the possibility to add courses to the platform and to test the understanding degree of the course. One of the best possibilities is to ask questions from the studied course.

The paper presents the current stage of the implementation of a tool that can manage multimedia content and can generate questions automatically based on the tags defined by the professor. In order to generate questions, the professor can add new tags, delete the existing ones and generate questions specifying the part of the course that should be used for questions.

The structure of the paper is: in the Section II it is presented the e-learning platform design. In the Section III it is presented the questions generation application, the architecture of the application, and in the last part, the conclusions.

II. General Description of the Platform

The platform is currently in the development stage. It contains two main modules: one for testing, and one for learning.

The testing module gives the possibility for the professor to define tags and questions templates that will be used to automatically generate questions from courses added to the platform.

The learning module manages all the courses added by the professor into the system.

The platform is role based and has defined four roles:

- secretary – who has absolute rights for the information added to databases,
- professors – who have absolutes rights over the disciplines assigned to them,
- students – who can only download courses and take tests.
- administrator – manages the structures of the databases and the users

Secretary can manage sections, professors, disciplines and students. The secretaries have also the task to set up the structure of study years for all sections.

The main task of a professor is to manage the assigned disciplines. The professor sets up chapters for each assigned

The support of the The National University Research Council under Grant CNCSIS IDEI 535 is gratefully acknowledged.

discipline by specifying the name and the course document and manages test and exam questions for each chapter.

The platform offers students the possibility to download course materials, take tests and exams and communicate with other involved parties like professors and secretaries.

III. Testing Module – Questions Generator

A. Technical Description

The module used for testing gives the possibility to the professor to define the most important aspects in each course using tags, and the system will automatically generate questions based on these tags.

The system is implemented in java and the following classes have been used: POIDoc – class that opens and returns the content of a MS Word document. This class uses Apache POI Project. Imports 3 jar files: poi-3.0.1.jar, poi-contrib-3.0.1.jar, poi-scratchpad-3.0.1.jar [8].

The POI project consists of APIs for manipulating various file formats based upon Microsoft's OLE 2 Compound Document format using pure Java. We can read and write MS Word, Excel files using Java.

POI is your Java Word solution as well as your Java Excel solution. HWPF is the name of our port of the Microsoft Word 97 file format to pure Java. However, it does not support the new Word 2007, .docx file format, which is not OLE2 based [8].

For basic text extraction, we make use of org.apache.poi.hwpf.extractor.WordExtractor. It accepts an input stream or a HWPFDocument.

B. General Description

In this paragraph there are presented concepts and working style for the Test Creator software tool. The main window is organized in several sub-windows, each of them permitting some operations in a very simple manner. It is easily compatible with any learning domain: engineering, medical, or economical. In order to generate questions, there should be followed 3 steps [8]:

1. Defining tags or questions categories
2. Defining templates for a specific tag
3. Parsing the text in order to generate the templates

The basic idea for generating questions based on the course material is to define a list of tags, chosen by the teacher. Each tag represents a class of questions that are applicable for certain theoretical notions from the course. All tags in a category have similar formatting.

For example, the <DEFINE> tag will be used to formulate some questions where the student has to define some concepts. The <EXAMPLE> tag will be used for questions where it should be presented an example for a specific concept.

In the TestCreator module there is a sub-window where there are presented the tags that already exists in the database.

They can be updated using insert/delete operations. The most important thing is that the teacher has total freedom in choosing these tags, as he considers being most suitable for his course domain [8].

For each tag, the teacher defines one or several forms of a question, suitable for a specific category. These forms of the questions were called templates (figure 1).

For example, for the <DEFINE> tag it can be defined the following template: "DEFINE #". For the tag <WHAT IS>, the template can be "What is a/an #". The use of "#" sign represents for this application the reserved word or phrase to which it is applied the tag.

One of the problems was to have several forms for the same tag in order to give the possibility to the teacher to create questions that are correct from the syntactic and semantic point of view, or another situation that appears frequently refers to the possibility to formulate the same questions both in the native language and in an international language.

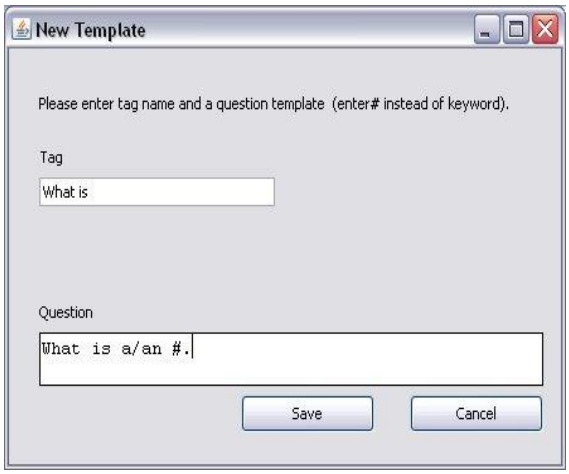


Fig. 1 Defining the question template

The final step is represented by the questions generation. The professor has to load first the course material in the main window. Then, in order to generate questions, he has to select the keyword or key-phrase from the text and then to select the tag and template.

The "#" sign will be replaced with the chosen word/phrase from the course.

For example, for the DEFINE # template, and the keywords "Boyce-Codd normal form" there will be generated the question: DEFINE Boyce-Codd normal form.

As soon as it was generated, the question is displayed in another window of the software tool. The teacher has to decide if it is correct and if it should be kept and stored in the database (figure 2) [8].

The questions generated for each course are displayed in a special window and the professor has the possibility to delete, update and store them in the database.

C. The Structure of the Database

For the database needed in this platform it was used the MySQL Database Server.

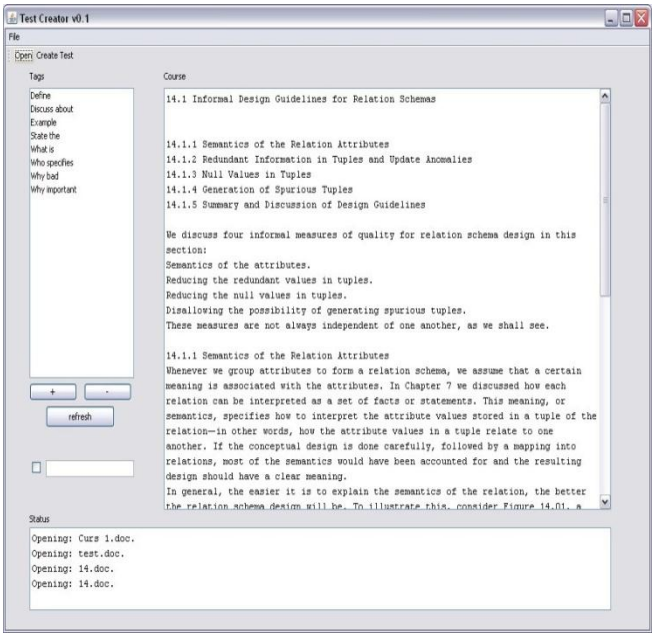


Fig. 2 The main window of the software tool

The database defined for the application includes the following tables:

a) Users
b) Students
c) Disciplines
d) AsociatedDisciplines
e) Tags
f) Courses
g) Templates
h) Questions
i) Results

Where:

"Users" table stores information about users defined in the system. Each registered student must have his own password and username.

"Students" table stores personal information about the students loaded into the system.

"Disciplines" and "AsociatedDisciplines" tables stores information about the courses available into the system and how they are associated to each student.

The "Tags" table is the one that stores information about existing query categories.

The "Templates" table has a connection of 1:m with the table Tags. It is used to store for each query category, different forms of the queries.

The "Course" table stores information about electronic courses and their chapters. Actually these chapters will be loaded into the application in order to generate questions from them.

Between Courses table and Templates table there is a m:m connection. In the relational model this will lead to a new relation, called "Questions" where it will be stored the effective content of the generated query.

The "Results" table stores the results obtained by students to different tests.

II. Conclusions and Future Work

The paper presented the work in progress of a software tool for the e-learning process that has as an element of originality the methods for generating automatic questions from course materials.

It is a role based platform where the professor can define tags and templates that will be used in query generation.

The Test Creator module permits generation of questions based on electronic materials used by the students. The solution implies for teachers to have a series of defined tags and templates that they have to manage. These tags can be used to generate questions automatically.

The platform will be extended adding methods for multimedia content management, especially for images management. These methods are considered to be useful in the medicine domain where the examples presented to classes are reduced to minimum and the students need support with images from real cases.

In order to find the most important part in a course material it will be created a list of concepts and a Concept map.

For these steps of the process it should be used a third party software tool specialized in concept maps aspects.

It is preferable to include concept maps concepts in the learning process for two reasons:

- you can be sure that you will have questions about all important concepts existing in the course
- you can monitor learning activity to be sure that students learn meaningful and not only several separate aspects, with no connection between.

References

[1] A. Andrenucci, E. Sneiders, "Automated Question Answering: Review of the Main Approaches", in *Proceedings of the 3rd International Conference on Information Technology and Applications (ICITA'05)*, July 4-7, Sydney, Australia, IEEE, Vol. 1, 2005, pp.514-519

[2] J. McGough, J. Mortensen, J. Johnson, S. Fadali "A web-based testing system with dynamic question generation". *LNCS 1611-3349*, 2008, pp. 242-251

[3] W. Wang, H. Tianyong, L. Wenyin, "Automatic Question Generation for Learning Evaluation in Medicin", in *LNCS Volume 4823*, 2008, pp.242-251

[4] L. Vecchia, M. Pedroni, "Concept Maps as a Learning Assessment Tool" in *Issues in Informing Science and Information Technology*, Volume 4.

[5] E. McDaniel, B. Roth, M. Miller, "Concept Mapping as a Tool for Curriculum Design", in *Issues in Informing Science and Information Technology*

[6] C. Jonathan, A. Gwen, E. Maxine Eskenazi, "Automatic question generation for vocabulary assessment", in *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, Vancouver, British Columbia, Canada, 2005, pp.819-826.

[7] D.D. Burdescu, M. C. Mihaescu, "Building a decision support system for students by using concept maps", in *Proceedings of International Conference on Enterprise Information Systems (ICEIS'08)*, Barcelona, Spain, 2008

[8] L. Stanescu, C. Stoica Spahiu, A. Ion, A. Spahiu, "Question generation for learning evaluation", *Proceedings of First International Symposium on Multimedia – Applications and Processing* (MMAP'08), 2008.

[9] C. Badica, M. Teodorescu, C. Spahiu, C. Fox, A. Badica, "Integrating Role Activity Diagrams and Hybrid IDEF for Business Process Modeling using MDA", *Proceedings of Synasc 2005*, Timişoara, Romania, 2005

[10] L. Stanescu, D.D. Burdescu, C. Stoica Spahiu, "Using R-Trees in Content-Based Region Query with Spatial Bounds", *Proceedings of Synasc*, Timişoara, Romania, 2005

Person Tracking in Video Surveillance Systems Using Kalman Filtering

Suliman C., Cruceru C., Macesanu G. and Moldoveanu F., *Member, IEEE*

Abstract— **In this paper we have developed a Simulink based model for monitoring contacts in a video surveillance sequence. To correctly identify a contact in a surveillance video, we have used the Lucas-Kanade optical flow algorithm. The position and the behavior of the correctly detected contact was monitored with the help of the traditional Kalman filter. Here we compare the results obtained from the optical flow with the ones obtained from the Kalman filter, and we show the correct functionality of the Kalman filter based tracking. The tests were performed using video data taken with the help of a fix camera. The tested algo-rithm has shown promising results.**

I. Introduction

THE problem of using vision to track and understand the behavior of humans is a very important one. The main applications that it has are in the areas concerning human-robot interaction [6], robot learning, and video surveillance.

Here we try to focus our attention on video surveillance systems. A high level of security in public places is an extremely complex challenge. A number of technologies can be applied to various aspects of security, including biometric systems, screening systems, and video surveillance systems. Nowadays video surveillance systems act as large-scale video recorders, analog or digital. These systems serve two main purposes: to provide a human operator with images to detect and react to potential threats and recording for future investigative purposes.

From the perspective of real-time detection, it is well known that the human's visual attention drops below acceptable levels even if that operator is a trained one in the task of visual monitoring. Video analysis technologies can be applied to develop smart surveillance systems that can aid the operator in the detection and in the investigatory tasks.

For surveillance applications, the tracking problem is a fundamental component. In video surveillance one of the most used method for tracking contacts is the particle filter [7][10][11][13]. Another well known method in the research community is the use of the traditional Kalman filter [9]. In many cases the use of this type of filter is sufficient. This is due to the controlled indoor and outdoor environments that are used in the studies.

Many papers in the literature detail methods that track single persons only [5][10], but there are also many authors that describe different methods for the detection and tracking of multiple persons [2][3][4][11]. Most of these methods involve as testing grounds indoor environments [1][3][7][13] as well as outdoor environments [2][4][7][9], where these methods are applied to track groups.

The objective of this paper is the development of a video surveillance system capable of tracking multiple persons in an outdoor environment. In Section II we describe the structure of the proposed video surveillance system. In Section III we present the method used for contact detection and the method used for the extraction of useful data from the video feed. Section IV describes the Kalman filter algorithm applied in our case. In Section V and VI we present the results obtained from the Simulink model's simulation, the conclusions drawn from this study and the possible future developments.

II. Surveillance System Structure

In this paper we examine the feasibility of using the optical flow algorithm in conjunction with the Kalman filter algorithm [9][12] for tracking multiple contacts in a surveillance scene. In order to create an algorithm that is able to track a contact in a scene, three different, large-scale task must be accomplished (see Fig. 1). First the algorithm needs to take an incoming surveillance video signal and segment it into a stream of frames where contacts are distinguished from the background of the scene. The next step is the tracking of the contacts throughout the video sequence. Finally, the resulting track must be processed in order to analyze the contact's behavior.

For the segmentation process of the incoming video signal, the optical flow algorithm developed by Lucas and Kanade was used [8]. The optical flow algorithm approximates the movement of the contact in the current frame as referenced to the previous frame. By determining the motion of objects, one can distinguish between the contact and the background of the scene. After careful tuning and processing, the output of the segmentation process is passed to the Kalman filter algorithm for further processing.

The Kalman filter is a recursive, adaptive filter that operates

Manuscript received August 31, 2010. This paper was supported by the Sectoral Operational Programme Human Resources Development (SOP HRD), fi-nanced from the European Social Fund and by the Romanian Government under the contract number POSDRU/6/1.5/S/6.

Suliman C. is with the Automatics Department, "*Transylvania*" University of Brasov, Brasov, Romania (e-mail: caius.suliman@unitbv.ro).

Cruceru C. is with the Automatics Department, "*Transylvania*" University of Brasov, Brasov, Romania (e-mail: cri.cruceru@yahoo.com).

Macesanu G. is with the Automatics Department, "*Transylvania*" University of Brasov, Brasov, Romania (e-mail: gigel.macesanu@unitbv.ro)

Moldoveanu F. is with the Automatics Department, "*Transylvania*" University of Brasov, Brasov, Romania (e-mail: moldof@unitbv.ro).

in the state space. It is well known for its ability to track objects in a timely and accurate manner. The tracking algorithm developed in this paper is able to process multiple contacts.

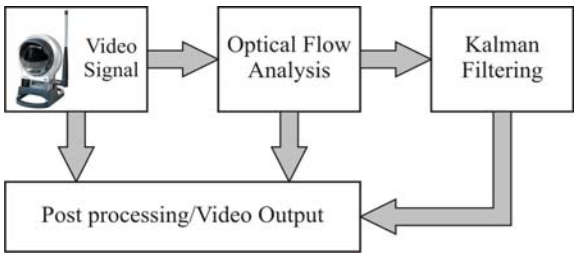


Fig. 1. The surveillance system structure.

III. Person Detection and Position Estimation

In Fig. 2 we will present the main component parts of the Optical Flow Analysis block.

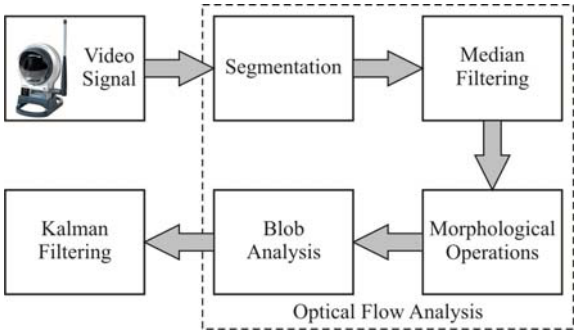


Fig. 2. The optical flow analysis block.

A. Optical flow analysis

One of the important blocks presented in the above scheme is the so called *optical flow analysis* block. The main purpose of this block is to determine the existence of possible contacts in the incoming video signal and process them in such manner that the Kalman filter will be able to estimate their position with minimal error.

In the following we will describe the functionality for each component of the optical flow analysis block.

1) Segmentation

In our case, the term *segmentation* is used to describe the process through which a video signal passes to become a series of binary images. At the output of this sub-block, each of the resulting binary images will contain black and white areas. The black areas correspond to the portion of the frame where no motion was detected, and the white areas correspond to the portion of the frame where motion was detected.

Because we use the optical flow to detect motion, the video signal needs to be converted to the intensity color space. To estimate the optical flow between two images we use the algorithm developed by Lucas and Kanade [8]. In our case this algorithm is used to compute the optical flow between the current frame and the previous one. This is one of the tunable parameters used in our experiments. The method proposed by Lucas and Kanade woks by assuming that optical flow is constant in an $m \times m$ window (mask), with $m > 1$. This window is centered on the pixel. The numbering of the pixels begins from 1 to n, where $n = m^2$. From this we will have the following set of equations:

$$\begin{bmatrix} I_{x1} & I_{y1} \\ I_{x2} & I_{y2} \\ \vdots & \vdots \\ I_{xn} & I_{yn} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} -I_{t1} \\ -I_{t2} \\ -I_{t3} \end{bmatrix} \quad (1)$$

In the above equations we can see that we have only two unknowns (u and v). The solution for the above system of equations will have the following form:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \sum I_{xi}^2 & \sum I_{xi} I_{yi} \\ \sum I_{xi} I_{yi} & \sum I_{yi}^2 \end{bmatrix} \begin{bmatrix} -\sum I_{xi} I_{ti} \\ \sum I_{yi} I_{ti} \end{bmatrix}, i = 1...n \quad (2)$$

From the above we can say that the optical flow can be obtained by computing all of the image derivatives for all the values of i.

Another important tunable parameter used in the optical flow estimation is the threshold for noise reduction. It is used for eliminating the small movements between frames. The higher its value, the less small movements impact the optical flow computing. Our experiments pointed out that the optimal value for the smoothness factor is 0.005.

Before the processed video signal exits the segmentation sub-block it is compared with a certain threshold to keep only what interests us from the video feed.

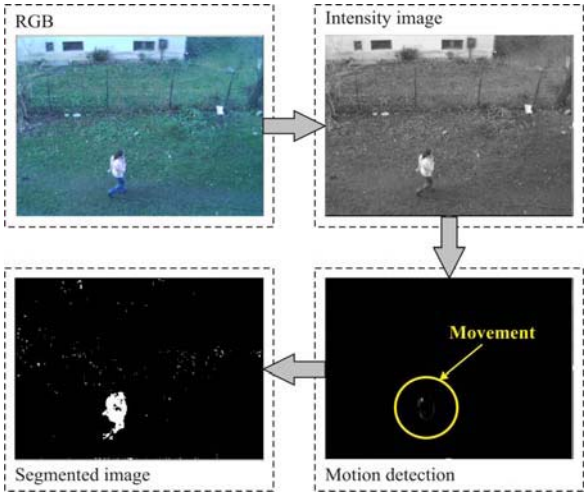


Fig. 3. The segmentation process.

2) Median Filtering

One of the biggest problems that optical flow has it's that it is very sensitive to changes in illumination or to the quality of the video. This sensitivity conduces in erroneous blobs appearing in individual frames. If these blobs are

large, that means that they are approaching the average size

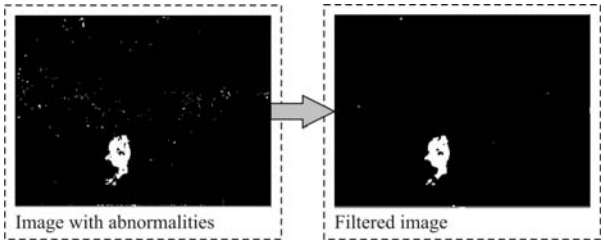


Fig. 4. Median filtering.

of the contact that we are trying to track, they can create problems for successful morphological operations.

The main reason for choosing the median filter is that most of these abnormalities, the erroneous blobs, appear in singular frames and they do not appear again for several more frames. The median filter is used to decrease the effect of these abnormalities while still maintaining the information of the correctly detected contacts (see Fig. 4).

3) Morphological Operations

The morphological operation will process the video signal coming from the output of the median filtering sub-block in such way that all erroneous blobs residing in the image are eliminated and all and only the correctly detected blob is maintained and classified as a real contact. The main morphological operations used by us in this study are the *erosion* and *dilatation*. Optimal erosion is achieved when the structuring element keeps at least the remnant of a blob for all correct contacts. If we use a sub-optimal structuring element for the erosion and dilatation operations, a valid contact could be lost completely, or an erroneous blob could be tracked. Both these errors can produce significant barriers to optimal dilatation. Optimal dilation is obtained when the structuring element merges all remnants of a single blob into one contact. If a sub-optimal structuring element is used for dilation, one contact could be viewed as multiple contacts or multiple contacts could be viewed as one contact. After an optimal structuring element for erosion was determined, each frame was eroded using the chosen structuring element. Determination of the optimal structuring element for dilation was similar to that of erosion. Each frame was dilated with a square structuring element.

An infinite number of possibilities exist for size and shape of structuring elements. Depending on the data used, the size and shape of the optimal structuring element could vary significantly.

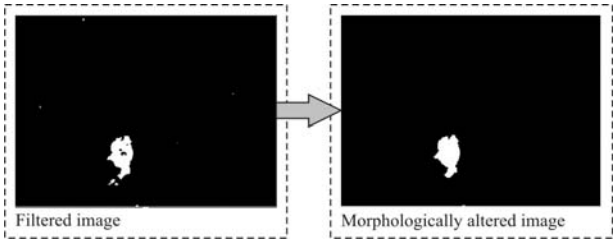


Fig. 5. Morphological operations.

After the median filtering the erosion operation removed great part of the remaining erroneous blobs residing in the image, thus deciding that they were not contacts. Further dilation has created a solid blob out of the area where motion was detected, and this blob will be tracked as a contact (see Fig. 5).

4) Blob Analysis

The main functionality of the blob analysis sub-block is to determine the minimum size of a blob and the maximum number of blobs that will be used in the Kalman position estimation step. By setting the minimum blob size we obtain a new level of protection against abnormalities by specifying a minimum size that a blob must have in order to be correctly tracked. Thus, any blob that doesn't fulfill this condition will not be tracked.

The other tunable parameter of the blob analysis sub-block, the maximum number of blobs, is used to set the number of Kalman filters to be used in the tracking process. In our case this parameter was set to 1.

B. Kalman Filtering

Filtering is a very used method in engineering and embedded systems. A good filtering algorithm can reduce the noise from signals while retaining the useful information. It estimates the states of linear systems. This type of filter works very well in practice and that is why it is often implemented in embedded control system and because we need an accurate estimate of the process variables. The discrete Kalman filter is characterized by both a process model and a measurement equation.

The process model is characterized by the assumption that the present state, x_k, can be related to the past state, x_{k-1}, as follows:

$$x_k = \Phi_k x_{k-1} + w_k , \tag{2}$$

where w_k is assumed to be a discrete, white, zero-mean process noise with known covariance matrix, Q_k; Φ_k represents the state transition matrix which determines the relationship between the present state and the previous one.

In our case we try to estimate the current state of an contact based on its last known state. Here, the state vector consists of a two-dimensional position expressed in Cartesian coordinates, a two-dimensional velocity and a two-dimensional acceleration. By considering a constant acceleration, the state transition matrix can be determined from the basic kinematic equations as follows:

$$s_k = s_{k-1} + v_{k-1}t + \frac{1}{2}a_{k-1}t^2 , \tag{3}$$

$$v_k = v_{k-1} + a_{k-1}t , \tag{4}$$

$$a_k = a_{k-1} , \tag{5}$$

where s is defined to be the contact's position, v is its velocity, a is the contact's acceleration and t is the sampling period. In a matrix form, the above equations can be written as:

$$\begin{bmatrix} s_{x,k} \\ s_{y,k} \\ v_{x,k} \\ v_{y,k} \\ a_{x,k} \\ a_{y,k} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 & 0.5 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0.5 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_{x,k-1} \\ s_{y,k-1} \\ v_{x,k-1} \\ v_{y,k-1} \\ a_{x,k-1} \\ a_{y,k-1} \end{bmatrix}. \quad (6)$$

Here, the subscripts x and y refer to the direction of the contact's position, velocity and acceleration in the two-dimensional plane. The value of the sampling period is set to 1. From the above equation the state transition matrix, Φ_k, is:

$$\Phi_k = \begin{bmatrix} 1 & 0 & 1 & 0 & 0.5 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0.5 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (7)$$

The measurement equation is defined as:

$$z_k = H_k x_k + v_k, \quad (8)$$

where z_k represents the measurement vector, v_k is assumed to be a discrete, white, zero-mean process noise with known covariance matrix, R_k. The matrix H_k describes the relationship between the measurement vector, z_k, and the state vector, x_k. Given the fact that the state vector is of length six and the measurement vector is of length two, the matrix H_k must be of length six by two:

$$H_k = \begin{bmatrix} 1 & 0 & 1 & 0 & 0.5 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0.5 \end{bmatrix}. \quad (9)$$

From the process model and measurement equation it results that the Kalman filter attempts to improve the prior state estimate using the incoming measurement which has been corrupted by noise. This improvement can be achieved by linearly blending the prior state estimate, $\hat{x}_{k-1}$, with the noisy measurement, z_k, in:

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H_k\hat{x}_k^-). \quad (10)$$

Here $\hat{x}_k^-$ means the a-priori estimate; K_k is known as the blending factor. The minimum mean squared error of the estimate is obtained when the blending factor assumes the value of the Kalman gain:

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1}, \quad (11)$$

where P_k is known as the state covariance matrix. Generally the state covariance matrix is a diagonal matrix. The state covariance matrix is determined from the a-priori state covariance matrix as follows:

$$P_k = (I - K_k H_k) P_k^-. \quad (12)$$

After the Kalman gain has been computed, and the state and state error covariance matrices have been updated, the Kalman filter makes projections for the next value of k. These projections will be used as the a-priori estimates during processing of the next frame of data.

$$\hat{x}_{k+1}^- = \Phi_k \hat{x}_k, \quad (13)$$

$$P_{k+1}^- = \Phi_k P_k \Phi_k^T + Q_k. \quad (14)$$

The above equations are the projection equations for the state estimate and for the state covariance matrix.

The main role of the Kalman filtering block is to assign a position estimation filter to each of the measurements entering the system from the optical flow analysis block. Therefore we have a filter for each of the detected contacts. For an easy implementation of the Kalman filter in Simulink, we wrote an embedded Matlab function. This method is often used when the function that needs to be implemented is more easily to express in Matlab's symbolic language than in Simulink's graphical language.

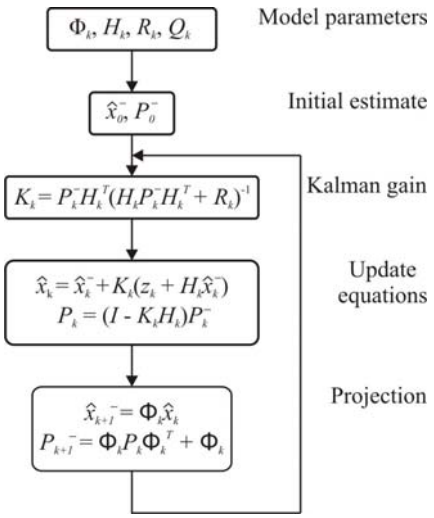


Fig. 6. The Kalman filter algorithm.

IV. Post Processing

The last block that we will discuss is the post processing/video output block. This block was used to process the output from the optical flow analysis block and the output of the Kalman filtering block.

The post processing block is composed of three video output sub-blocks. The first sub-block is used only to view the original video signal.

The second sub-block is used to visualize the resulting signal from the optical flow analysis block and to allow the user to be sure of the correct functionality of the optical flow analysis block. This sub-block is in direct connection with the blob analysis sub-block, block that produces the coordinates for a bounding box. This bounding box is a rectangle drawn around each correctly detected blob. The user is able to watch in real-time which contact in the video

feed is being sent to the Kalman filtering block. If rectangles are not surrounding the correctly detected contacts in an image, this thing means that the optical flow analysis bloc is not working properly. Figure 7a presents the resulting output of the optical flow video viewer sub-block. We present only four frames taken at 17.5 FPS of each other. It can be clearly seen that a contact was detected and a bounding box was correctly superimposed on the contact. The same thing happens in figure 7b where two contacts have been detected. The last sub-block discussed in this section is the Kalman filtering video viewer. This sub-block is in direct connection with the Kalman filtering block. This block produces at its output a matrix containing the position of the detected contact. The output is used by the Kalman filtering video viewer to draw markers in the video. These markers are represented here by red filled circles. The user is able to see in real-time if the detected contact is correctly tracked by the Kalman filter. If a marker doesn't follow consistently a contact we can say that the Kalman filtering block isn't working properly. Figure 8a presents four frames resulting from the Kalman filtering video viewer, taken like in the previous case, at 17.5 FPS of each other. We can clearly see that the marker is correctly tracking the detected contact thus confirming that the Kalman filter is working properly. The Kalman filter is even capable to track the contact that is leaving at some time the visual field of the camera and then correctly reassign the marker to the contact that reenters in the visual field. Figure 8b presents the tracking process for two correctly detected contacts.

V. Conclusions and Future Work

There are two main factors that affect the problem of tracking: the accuracy to distinguish between contacts passing through the scene and the speed to process the video feed in real-time. In this paper we have shown that with the help of the optical flow and Kalman filter algorithms it is possible to detect and track a person passing through a scene.

The video signal used in our experiments it is provided by a Linksys WVC200 PTZ IP video camera at a resolution of 240x320. The entire experiment was conducted using an Intel Core2Duo T9300 computer with 4 GB of RAM.

From the optical flow analysis used in our research we have deduced that there is an inevitable trade-off between the accuracy and speed of processing. To accurately distinguish a contact that passes through a scene, the computational time of the optical flow algorithm must be increased. If this increase of the processing time is too large, the algorithm will not operate in real-time.

(a)

(b)

Fig. 7. Output sample from the optical flow video viewer.

(a)

(b)

Fig. 8. Output sample from the Kalman filtering video viewer.
(a) tracking a contact that passes through the camera's field of view; (b) tracking two contacts that are passing through the camera's field of view.

The Kalman filter algorithm presented in this research was able to correctly process a contact and to correctly assign a filter to the processed contact. After reviewing the results we deduced that the algorithm performed quite well showing a moderate consistency in tracking. Due the success with the data used in our experiments, any inconsistencies in the tracking process can be traced back to the fluctuations in performance of the optical flow algorithm.

Future research in the area of surveillance systems should be focused in two directions. First, research should be made to determine an objective measure of performance of the optical flow algorithm and to see if other existing algorithms are better suited to accurate, real-time processing of a video signal. The second research should be focused in the area of determining contact behavior, and in areas such as merging contacts into groups or dividing groups into separate contacts.

For a future research we will try to make a comparison between the presented segmentation method for contact detection and other segmentation methods .

ACKNOWLEDGMENT

This paper is supported by the Sectoral Operational Programme Human Resources Development (SOP HRD), financed from the European Social Fund and by the Romanian Government under the contract number POSDRU/6/1.5/S/6.

REFEERENCES

[1] M. A. Ali, S. Indupalli, B. Boufama, "Tracking Multiple People for Video Surveillance," *First Intern. Workshop on Video Processing for Security*, June 2006.

[2] B. Benfold, I. Reid, "Guiding Visual Surveillance by Tracking Human Attention," *Proc. of the 20th British Machine Vision Conf.*, September 2009.

[3] L.M. Fuentes, S.A. Velastin, "From tracking to advanced surveillance," *Proc. of the Intern. Conf. on Image Processing*, vol. 3, pp. 121–124 , September 2003.

[4] C. C. Hsieh, S. S. Hsu, "A Simple and Fast Surveillance System for Human Tracking and Behavior Analysis," *Proc. of the 3rd Intern. IEEE Conf. on Signal-Image Technologies and Internet-Based System*, pp. 812–818, December 2007.

[5] F. Jean, R. Bergevin, A.B. Albu, "Body tracking in human walk from monocular video sequences," *Proc. of the 2nd Canadian Conf. on Computer and Robot Vision*, pp. 144–151, May 2005.

[6] N. Koenig, "Toward real-time human detection and tracking in diverse environments," *Proc. of the 6th IEEE Intern. Conf. on Development and Learning*, pp. 94–98, July 2007.

[7] S. Kong, M.K. Bhuyan, C. Sanderson, B.C. Lovell, "Tracking of Persons for Video Surveillance of Unattended Environments," *Proc. of the 19th Intern. Conf. on Pattern Recognition*, pp. 1–4, December 2008.

[8] B. Lucas, T. Kanade, "An iterative image registration technique with an application to stereo vision," *Proc. of the Intern. Joint Conf. on Artificial Intelligence*, pp. 674–679, 1981.

[9] W. Niu, L. Jiao, D. Han, Y. F. Wang, "Real-time multiperson tracking in video surveillance," *Proc. of the 4th Pacific Rim Conf. on Multimedia*, vol. 2, pp. 1144–1148, December 2003.

[10] A.W. Senior, G. Potamianos, S. Chu, Z. Zhang, and A. Hampapur, "A comparison of multicamera person-tracking algorithms," *Proc. IEEE Int. Works. Visual Surveillance*, May 2006.

[11] J. Wang, Y.g Yin, H. Man, "Multiple Human Tracking Using Particle Filter with Gaussian Process Dynamical Model," *EURASIP Journal on Image and Video Processing*, vol. 2008, Article ID 969456, 10 pages, 2008.

[12] G. Welch, G. Bishop, "An Introduction to the Kalman Filter,", *Technical Report: TR95-041*, University of North Carolina, 2006.

[13] J. Yao, J.M. Odobez, "Multi-Camera 3d Person Tracking With Particle Filter In A Surveillance Environment," *Proc. of the 16th European Signal Processing Conf.*, August 2008.

Intelligent Trajectory Tracking in Sliding Mode Based Wheeled Mobile Robot Control

Vasiliu Grigore, Mihalcea Ionut, Radjabov Serghei, Filipescu Adriana

Abstract — This article presents three different solutions to increase the accuracy of a trajectory planning. The first is the use of infrared sensors, than trajectory calculation using Bézier curves and sliding mode. In this study a scenario in which obstacles are detected by infrared sensors is proposed. The robot performs obstacle avoidance through the use of Fuzzy Logic.

Keywords: ***Mobile robot, Trajectory tracking, Sliding Mode, Laser range senso, Fuzzy Logic.***

I. INTRODUCTION

The word *robot* is commonly defined as a mechanical device capable of performing human tasks, or behaving in a human-like manner [1].

For an autonomous mobile robot performing a navigation-based task in a vague environment, to detect and to avoid encountered obstacles is an important issue and a key function for the robot body safety as well as for the task continuity. Obstacle detection and avoidance in a real world environment that appears so easy to humans - is a rather difficult task for autonomous mobile robots and is still a well-researched topic in robotics [2].

The progress in the field of robots has been painfully slow. Robotics is still a cottage industry, even considering the special-purpose automatons now in wide use in automotive manufacturing.

II. SHARP IR RANGE FINDER

A. DESCHIPTION

The Sharp IR (Infrared) Range Finder works by the process of triangulation, the process of determining the location of a point by measuring angles to it from known points at either end of a fixed baseline, rather than measuring distances to the point directly. The point can then be fixed as the third point of a triangle with one known side and two known angles. Triangulation today is used for many purposes, including surveying, navigation, metrology, astrometry, binocular vision, and gun targetting.

Vasiliu Grigore, professor at the "Dunarea de Jos" University, Galati, Romania, Regiment11Siret str, no 29bis, C37, ap.29, (e-mail: vasiliugrigore3@yahoo.com).

Mihalcea Ionut, student at the the "Dunarea de Jos" University, Galati, Romania, Traian Vuia str, no 50, J6, ap.45, (e-mail: mihalceaionutvalentin@gmail.com).

Radjabov Serghei, student at the the "Dunarea de Jos" University, Galati,(e-mail: serghei.radjabov@gmail.com).

Filipescu Adriana, student at the "Dunarea de Jos" University, Galati, Romania, Armata Poporului str, no 12, L6, ap.49, (e-mail: adriana_filipescu@yahoo.ca).

The use of triangles to estimate distances goes back to ancient times. In the 6th century BC the Greek philosopher Thales is recorded as using similar triangles to estimate the height of the pyramids by measuring the length of their shadows at the moment when his own shadow was equal to his height [3].

A pulse of light (wavelength range of 850nm +/-70nm) is emitted and then reflected back. When the light returns it comes back at an angle that is dependent on the distance of the reflecting object. Triangulation works by detecting this reflected beam angle , by knowing the angle, distance can then be determined.

The IR range finder receiver has a special precision lens that transmits the reflected light onto an enclosed linear CCD [4] array based on the triangulation angle. The CCD array then determines the angle and causes the rangefinder to then give a corresponding analog value. Additional to this, the Sharp IR Range Finder circuitry applies a modulated frequency to the emitted IR beam. This ranging method is almost immune to interference from ambient light, and offers amazing indifference to the color of the object being detected. In other words, the sensor is capable of detecting a black wall in full sunlight with almost zero noise.

A major problem/advantage with the Sharp IR rangefinder is beam width. Unlike sonar, its fairly thin, meaning to detect an object the sensor must basically point directly at that object.

The beam width is the same diameter as the lens on the Sharp IR range finder transmitter. As the IR detector was moved away from the obstacle, the beam fades and the diameter expands [5].

B. THE OUTPUT

The Sharp IR has a non-linear output. This means that as the distance increases linearly (by set increments), the analog output increases/decreases non-linearly. The image bellow is a typical expected output from your range finder. Notice the strange kink in the beginning of the graph. This is because the range finder is not capable of detecting very short distances.

For effectively use of Sharp IR Range Finder a voltage output versus distance chart is needed (fig. 2.1). Without a chart should be run an experiment that measures distance versus the output analog value. By placing an object in front of the sensor the distance is measured, then look at the *printf* output reading.

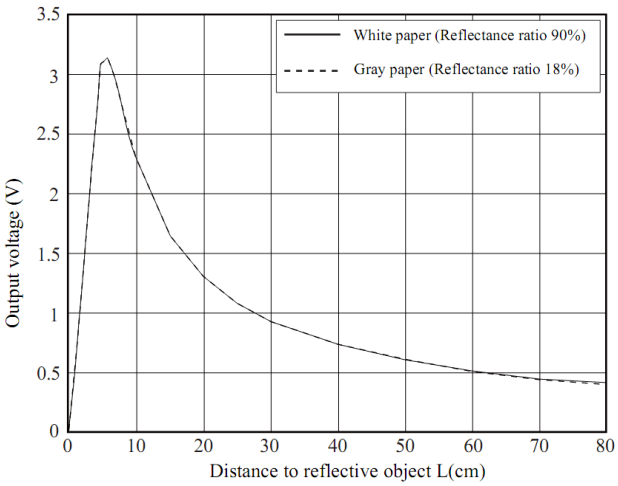


Figure 2.1: Voltage output versus distance

To minimize any noise the experiment was run in the environment where the robot will operate, in the faculty lounge – this way all ambient conditions are the same for highest accuracy. This is a very important step for calibrating any sensor.

C. EXPERIMENTAL DISADVANTAGES

One major issue with the Sharp IR Range Finder is when going below the minimum sensor range. This is when an object is so close the sensor cannot get an accurate reading, and it tells the robot that a really close object is really far. This is bad, as the robot then proceeds to ramp up in speed for a messy collision. Sonar also has this minimum range problem. The solution to this problem is to NOT put the sensor flush in the front of robot. Positioning the sensor into the robot's front the minimum sensor range is not accesible (fig 2.2).

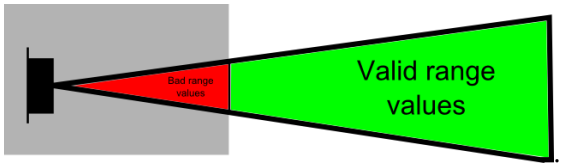


Figure 2.2 : Minimum sensor range

Another problem is the *narrowness* of the IR beam. In reading sharp details and getting high accuracy, a thin beam is ideal. But the problem with a thin beam is that if it is not pointed exactly at the object, the object is therefore invisible.

In contrast to the IR Range Finder would be the sonar. Sharp IR Range Finder beam has the widest portion about 16 cm wide.

An issue that these range finders have in common with sonar is cross interference. This means that the signal emitted by one sensor can potentially be read by another sensor and therefore give you bad readings. However, unlike sonar which have sound signals that can bounce off of multiple walls, we just need to make sure that the IR beams do not cross in parallel, but this makes sense because we have redundant sensors if the two beams cross (fig 2.3, fig. 2.4).

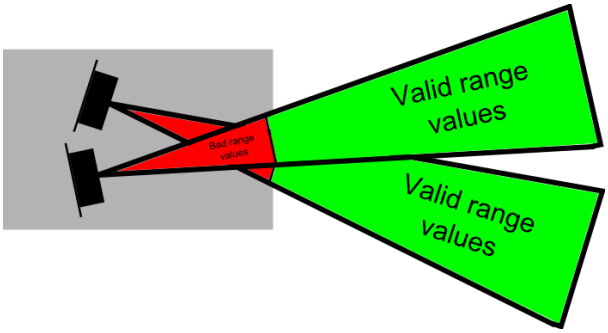


Figure 2.3: Technique 1 with the Sharp IR Range Finder

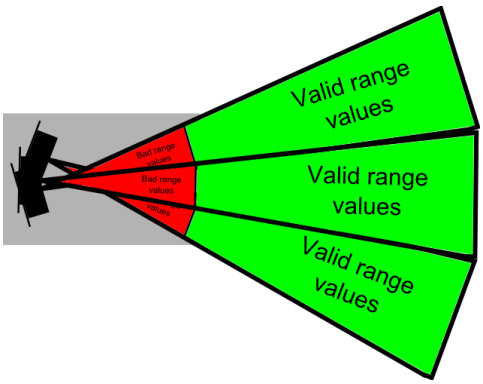


Figure 2.4: Technique 2 with the Sharp IR Range Finder

Therefore the use of only four sensors, limits the possibilities of obstacle recognition and the reaction time of the robot.The blind areas were eliminated using fuzzy techniques and diferent sensor arrangements.

D. FUZZY LOGIC

The robot can perform obstacle avoidance through the use of fuzzy Logic First, the outputs of the IR sensors first have to be classified as close, mid, or far, based on the fuzzification rule (fig. 2.5).

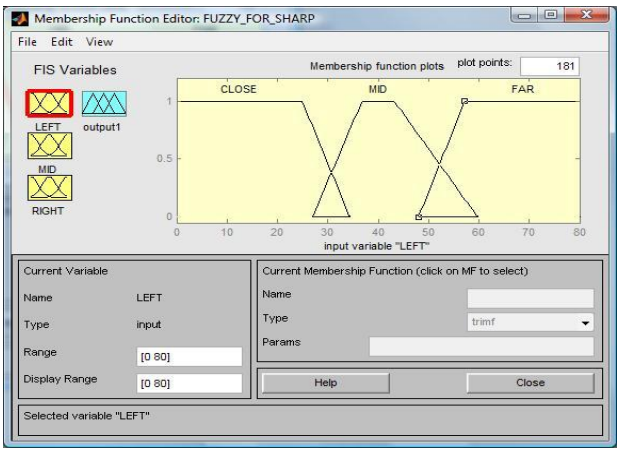


Figure 2.5 : Fuzzy logic

After that was created a fuzzy logic table to be able to identify the appropriate behavior based on the fuzzy output obtained from the fuzzification rules [6].

Table 2.1 : Fuzzy Logic table.

	Right Sensor/Middle Sensor								
Left Sensor	RF/MF	RF/MM	RF/MC	RM/MF	RM/MM	RM/MC	RC/MF	RC/MM	RC/MC
LF	F	L	B	L	L	B	L	L	B
LM	R	R	B	R	R	B	L	L	B
LC	R	R	B	R	R	B	B	B	B

LF = Left far LM =Left mid LC = Left close
RF = Right far RM = Right mid RC = Right close
MF = Middle far MM =Middle mid MC = Middle close
L = Left turn; R= Right turn; B = Back; F = Forward.

F = LF*RF*MF
B = (RF*MC+RM*MC+RC*MC) * (LF+LM+LC) + LC * (RM*MC+RC*MF+RC*MC)
R = (LM+LC)*(RF*MF+RF*MM+RM*MF+RM*MM)
L=LF*(RF*MM+RM*MF+RM*MM+RC*MF+RC*MM) + (LF+LM) * (RC*MF+RC*MM)

III. BEZIER CURVES

Bezier curves [7] are named after their inventor, Dr. Pierre Bezier [8]. Engineers may find it most understandable to think of Bezier curves in terms of the center of mass of a set of point masses. For example in the figure 3.1, 3.2, consider the four masses $m_0, m_1, m_2, m_3, m_4, m_5$ located at points P_0, P_1, P_2, P_3, P_4, P_5.

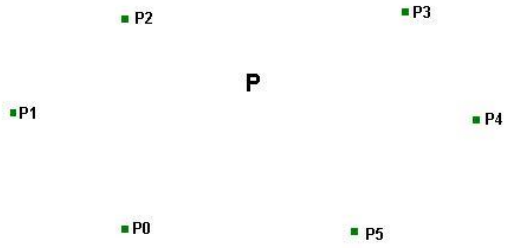


Figure 3.1: Center of mass of six points.

$$P = \frac{m_0 P_0 + m_1 P_1 + m_2 P_2 + m_3 P_3 + m_4 P_4 + m_5 P_5}{m_0 + m_1 + m_2 + m_3 + m_4 + m_5} \quad (3.1)$$

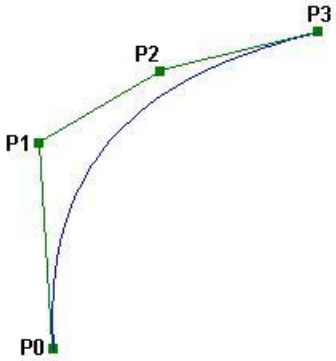


Figure 3.2: Cubic B´ezier curve.

These variable masses m_i are normally called blending functions and their locations P_i are known as control points or Bezier points. By drawing straight lines between adjacent control points, as in a dot to dot puzzle, the resulting figure is known as a control polygon. The blending functions, in the case of Bezier curves, are known as Bernstein polynomials.

Bezier curves of any degree can be defined. Figure 3.3 shows sample curves of degree one through four. A degree n Bezier curve has n +1 control points whose blending functions are denoted $B_i^n(t)$ (3.2) where

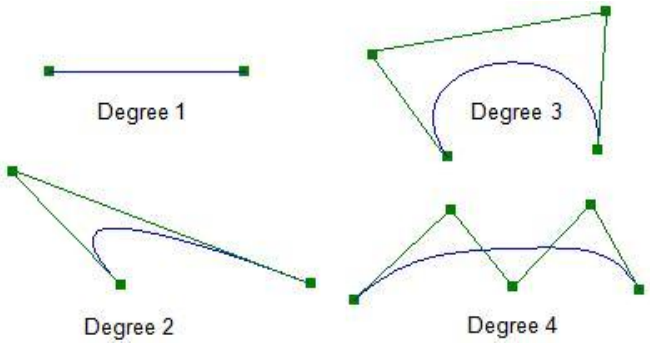


Figure 3.3: Bézier curves of various degree.

$$B_i^n(t) = \binom{n}{i}(1-t)^{n-i}t^i, i = 0,1,2,\dots,n \quad (3.2)$$

Recall that $\binom{n}{i}$ is called a *binomial coefficient* and is equal to $\frac{n!}{i!(n-i)!}$.
The equation of a Bezier curve(3.3) is:

$$P(t) = \sum_{i=0}^{n}\binom{n}{i}(n-1)^{n-i}t^i P^i \quad (3.3)$$

In figure 3.4 based on Bezier curve implemented in a JavaScript platform we generated the x and y coordinates of the curve.

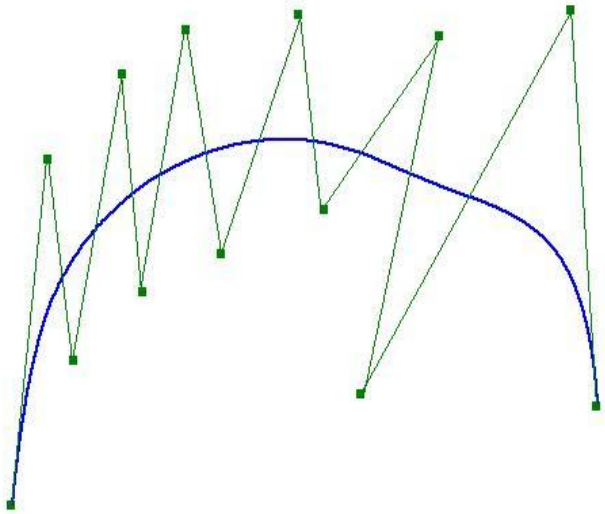


Figure 3.4: Generated B´ezier curve.

To view if the curve generated in JavaScript is the same in MATLAB was plotted the x and y vectors (fig 3.5).

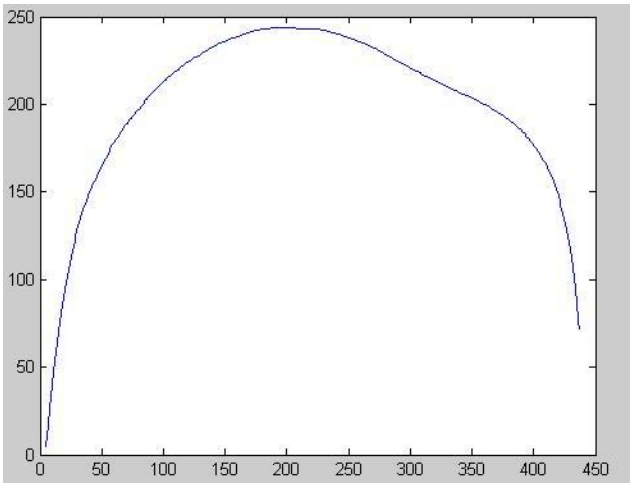


Figure 3.5: Plot x and y vectors

IV. TRAJECTORY TRACKING [9]

Trajectory tracking – when the vehicle is required to track a time parameterized reference. It is known that direct kinematics provides the position and orientation of the wheeled mobile robot [WMR] in the space. Inverse kinematics provides the sets of linear and angular velocity variables that will satisfy the same WMR position and orientation. Some of objectives of trajectory planning are to plan a collision free paths and avoid stationary obstacles.

Trajectory planning schemes helps to interpolate or approximate the desired path by a class of polynomial functions and generate a sequence of time based control set points for the control of the robot from the initial location to the goals (fig.4.1).

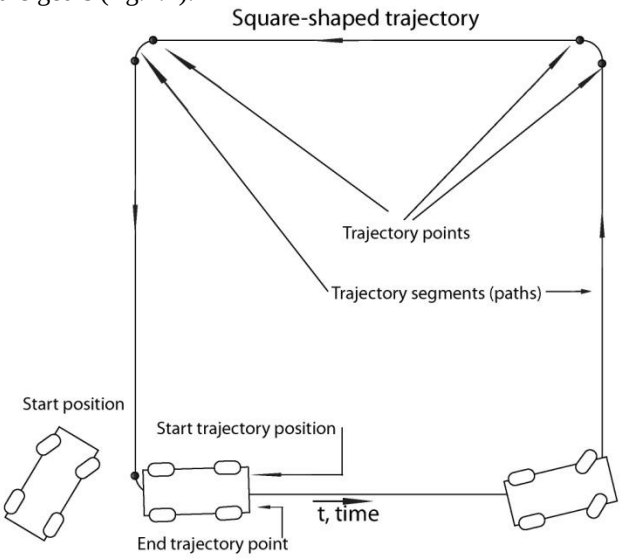


Figure 4.1. Trajectory tracking illustration

To have a smooth movement, the trajectory must be twice differentiable to give a continuous velocity and acceleration.

As a result, curve fitting is an integral part of trajectory planning where the path is broken un into straight and curved segments and velocity with angular velocity are individually planned for each segment. A number of techniques are used in curve fitting, including the use of B-splines [10], cubic splines, clothoids, etc.

The experiments contain some constraints such as upper velocity, acceleration limits, maximum curvature. Path end points and angles will be specified in Cartesian coordinates. Cartesian trajectory planning [11] can be represented like trajectories via points which is composed of a series of straight-line, constant-velocity trajectories connected via points. Position and orientation of robot at the end-link frame have to be interpolated from beginning to end of each trajectory segment.

In the end must be obtained a velocity profile generated by trajectory planner software which will be included in c++ project. The project is based on a console application. To connect to a robot and communicate with it Aria library are used. For better result of trajectory tracking was used sliding mode trajectory tracking algorithm.

A. Sliding Mode Trajectory Tracking Control For Two Drive Wheels And Two Direction Wheels WMR [12]

Sliding mode for trajectory tracking control is used to design a robust controller which will be capable to reduce a large class of disturbance. In order to correctly track desired trajectory, sliding mode controller implemented on WMR is using three types variables in it's calculations:

- variables from velocity profile
- variables from encoders (odometry)
- the error vector

A motion model of a mobile robot represented like simplified nonholonomic system (1) :

$$\begin{cases} \dot{x_r}(t) = v_r(t) * cos\phi_r(t) \\ \dot{y_r}(t) = v_r(t) * sin\phi_r(t) \\ \dot{\phi_r}(t) = \frac{v_r}{L} * tan\,\delta_r(t) \end{cases} \quad (1)$$

where x_r and y_r are the Cartesian coordonates of the rear axle midpoint, v_r is the velocity of this midpoint, ϕ_r is the mobile robot heading angle, L is the interaxle distance, δ_r the front wheel angle, which is the control variable to steer the vehicle.

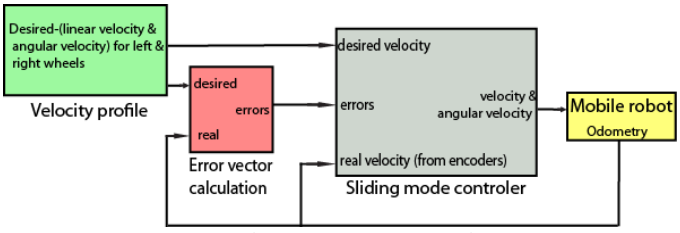


Figure 4.2 : A model of sliding mode controller for mobile robot

$$\begin{bmatrix} x_e \\ y_e \\ \phi_e \end{bmatrix} = \begin{bmatrix} cos\phi_d & sin\phi_d & 0 \\ -sin\phi_d & cos\phi_d & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} x_r - x_d \\ y_r - y_d \\ \phi_r - \phi_d \end{bmatrix} \quad (2)$$

where (x_r, y_r, ϕ_r) - are real robot position and heading (odometry), (x_d, y_d, ϕ_d) - are desired position and heading (velocity profile).

$$\begin{cases} \dot{x}_e = -v_d + v_r * cos\phi_e + y_e * \frac{v_d}{L} * tan\delta_d \\ \dot{y}_e = v_d * sin\phi_e - x_e * \frac{v_d}{L} * tan\delta_d \\ \dot{\phi}_e = \frac{v_r}{L} * tan\delta_r - \frac{v_d}{L} * tan\delta_d \end{cases} \quad (3)$$

where v_d - linear velocity, δ_d - front wheel angle.

Important observation is that $|\phi_e|$ should be smaller than $\pi/2$, which result that the robot orientation should not be perpendicular to the desired trajectory.

To calculate sliding surfaces was used just two control variables (y_e, ϕ_e) of three (x_e, y_e, ϕ_e) trajectory tracking variables (4),(5).

$$s_1 = \dot{x}_e + k_1 * x_e \quad (4)$$

$$s_2 = \dot{y}_e + k_2 * y_e + k_0 * sgn(y_e) * \phi_e \quad (5)$$

where k_0, k_1, k_2 - positive constant parameters, x_e, y_e and θ_e are the trajectory tracking errors calculated by equations (3).

B. Simulation

In the experiment PowerBot mobile robot tracked square-shaped trajectory with side length of square 2 [m]. To create velocity profile of this trajectory was used software developed in MATLAB. Velocity and acceleration are calculated in dependence of segments length and limits, where limits are specified for different robots. Limits should be configured in dependence of robot maximum velocity and acceleration or desired maximum velocity and acceleration.

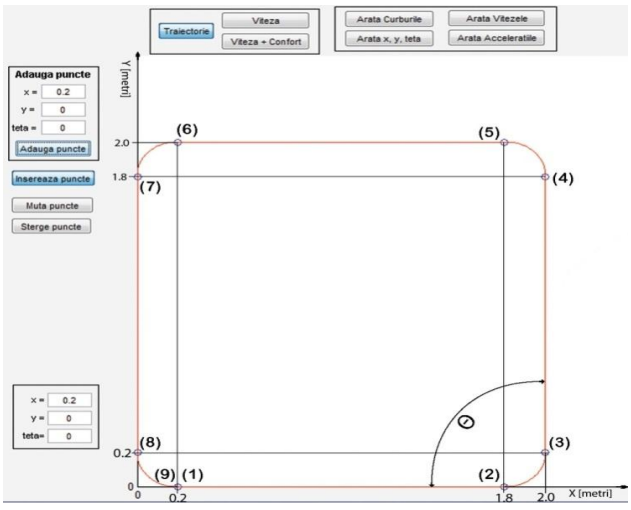


Figure 4.2: Square-shaped trajectory with side length of 2 [m].

The trajectory starts at point (1) with coordonates [0.2 0] and is composed of 9 points, which will be achieved ascending, from point (1) to point (9)-end of trajectory point.

PowerBot is a wheeled mobile robot with two drive wheels and two direction wheels [Fig. 4.3 (a)] with center of mass located not on the center of axes, which result that robot cannot to turn 90 degrees on the same place. In case of robot with center of mass located on center of axes, it is capable to turn around on the same place [Fig. 4.3 (b)].

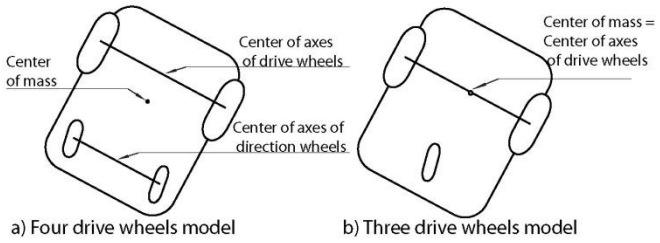


Fig. 4.3 : Models of robots with different center of mass

During experiments was determined that optimal maximum linear velocity and angular velocity of PowerBot is 0.6 [m/s] and 1.4 [rad/s] respectively. Was noted that this limits are not servomotors limits, and just optimal limits used in this experiments.

Before running the program in real-time, this was simulated on MobileSim robot simulator designed for MobileRobots/ActivMedia platforms and their environments.

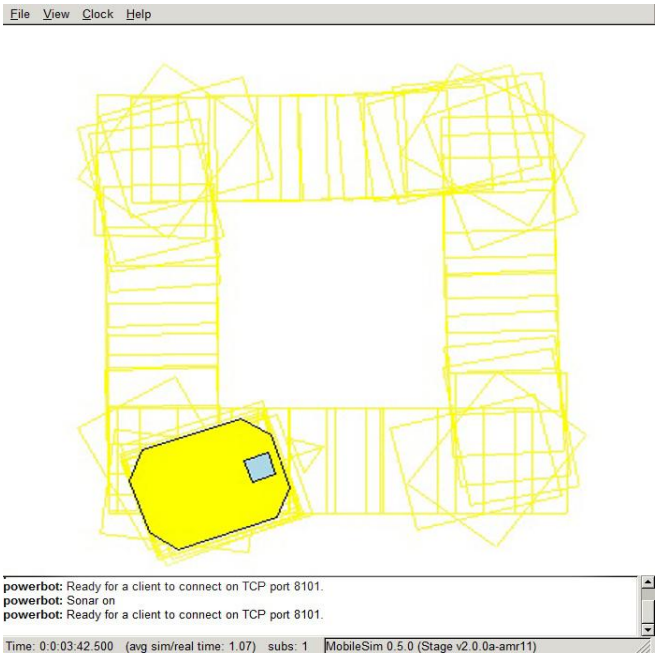


Figure 4.4: Simulation result of square-shaped trajectory tracked by PowerBot.

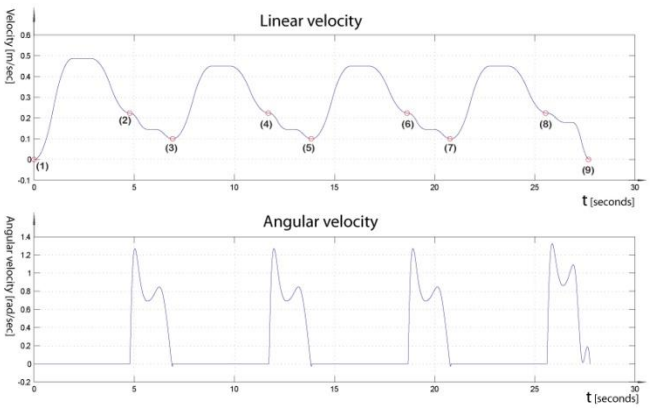


Fig. 4.5: Plots of linear and angular velocity

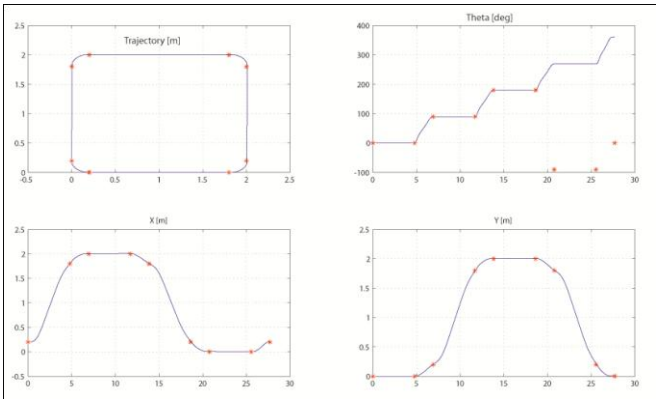


Figure 4.6: Plots of Trajectory, Theta [deg], X [m], Y [m]

RESULTS OF REAL-TIME EXPERIMENTS

Comparasion of simulation and real-time results

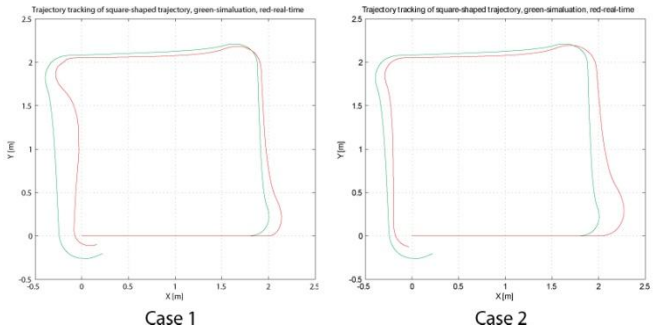


Figure 4.7. Plot of simulated and real-time tracked trajectory (case 1 and 2). Green-simulated trajectory, Red-trajectory in real-time

V. CONCLUSIONS

The Sharp IR Range Finder can be used for mapping by placing it on a servomotor. Then the servomotor will rotate to a degree, takes a distance reading, and records it. Then it moves to the next angle, takes another reading, and then records it. This will be done untill we obtain an array of distances of objects and the corresponding angles.

The Bézier curves helps to make the trajectory planning for the robot with the generated x and y vectors. As the number of segments increases the Bézier curve becomes smoother.

In conclusion the study of presented two cases of trajectory tracking showed that angular velocity on sharp corners should be reduced in order to avoid skidding and to obtain smoother and better trajectory tracking. In perspective, trajectory planner software should be improved in order to offer automatic detection of sharp corners and as result, generation of more accurate velocity profile.

REFERENCES

[1] GORDON McCOMB, "The Robot Builder's Bonanza", 2nd ed., Malestrom

[2] Saydo Soumare, "Real-Time Obstacle Avoidance by an Autonomous Mobile Robot using an Active Vision Sensor and a Vertically Emitted Laser Slit", Japan

[3] Diogenes Laërtius, "Life of Thales", The Lives and Opinions of Eminent Philosophers

[4] http://en.wikipedia.org/wiki/Charge-coupled_device

[5] http://www.societyofrobots.com, "SENSORS - SHARP IR RANGE FINDER"

[6] Phan Vu, "Phantastic Vudoo Contraption", University of Florida

[7] http://en.wikipedia.org/wiki/Bézier_curve

[8] http://en.wikipedia.org/wiki/Pierre_Bézier

[9] Imran Waheed and Reza Fotouhi (2008). Trajectory and temporal planning of a wheeled mobile robot on an uneven surface. Cambridge University Press

[10] March 14, 2005. An Introduction to B-Spline Curves. Thomas W. Sederberg

[11] Lecture Notes (Fall 2009). Chapter 6. Trajectory Planning. CS 5310/6310 & ME 5220/6220 Introduction to Robotics. Available: http://www.eng.utah.edu/~cs5310/chapter6.pdf

[12] Razvan Constantin Solea (March 2009). SLIDING MODE CONTROL APPLIED IN TRAJECTORY-TRACKING OF WMR's AND AUTONOMOUS VEHICLE. A DISSERTATION SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING OF COIMBRA UNIVERSITY

Dynamic Stability for Hydropower Plant Systems

Matei Vinatoru, *Member, IEEE*

***Abstract*—In hydro power plants from Romania, there is a major interest for the implementation of digital systems for monitoring and control, replacing the conventional control systems for power, frequency and voltage. Therefore is necessary to develop mathematical models capable to accurately describe both dynamic and stationary behaviour of the hydro units, in order to be able to implement digital control algorithms. Moreover, it is necessary to implement systems for monitoring and control of hydro power plants in a cascade system along a river, in order to optimise the use of the river resources. This paper presents the possibilities of modelling and simulation of the hydro power plants and performs an analysis of different control structures and algorithms.**

I. Introduction

THIS paper approaches the instability problems that may occur at hydro-electric groups and determination of the operating regimes that shall be provided by the automatic control system. For the study of the dynamic stability, the following shall be considered:

-For the hydro-plants with reservoir and surge tank, the hydraulic part of the group (the penstock from the reservoir to the turbine) introduces time constants much greater than the turbine and generator, thus the overall dynamic of the assembly is driven by the pressure losses on the hydraulic circuit.

-For run-of-the water plants where the pressure losses on the penstock are negligible, the stability is influenced by the dynamic of the turbine and generator, whose time constants change with the power generated by the group. In this case, the feedback due to the link between power, speed and flow require adequate control structures, usually adaptive ones, to maintain the entire system in the stability domain.

Additionally, in Romania there are hydro-electric plant assemblies along a river, which require SCADA systems to coordinate the overall operation of the system in order to optimise the usage of the entire hydro-electric potential of the river. Every lake has its own hydro power plant and the water used through the turbines of one plant is sent to the next lake, thus using the entire hydraulic potential offered by the geographical area.

The automatic control of an ensemble of hydro-electric plants built on a water course must provide the following functions:

- The optimal conversion of hydraulic energy to electric energy that can be ensured during a variable flow of the water by maintaining the water level in the reservoir at a maximum; and the production of electric energy will also be variable.

- Maintaining of the hydro-generators in the stability area considering the variable flow trough turbines and balancing the energy that can be produced and the energy consumed in the energetic system.

- The correlation of the flow trough turbines for the plants built in a cascade along the river and, therefore, also their energy production depending on the water flow from one reservoir to another.

- The automatic starting and stopping of groups (solving also the synchronization) considering the energy demands and the water available at an optimal use.

The paper proposes the development of simple control algorithms for the hydro-energetic groups based on simplified transfer functions that must also ensure the robustness in exploitation, when the groups' target is modified according to the requirements of hydraulic arrangements and to the energetic system demands.

II. The Dynamic Stability for Hydropower Plants with Reservoir and Surge Tank

Different construction of hydropower systems and different operating principles of hydraulic turbines make difficult to develop mathematical models for dynamic regime, in order to design the automatic control systems [1,2,5,6]. Also, there are major differences in the structure of these models. Moreover, there are major differences due to the storage capacity of the reservoir and the water supply system from the reservoir to the turbine. The dynamic model of the plants with penstock and surge tank is more complicated than the run-of-the-river plants, since the water feed system is a distributed parameters system. This paper present a solution for the modelling of the hydraulic systems and the design of the control system.

As it is know, the hydraulic power available at the turbine is a function of the water flow trough the turbine Q_t and the net head H_n

$$Ph = \gamma \, Q_t \, H_n \tag{1}$$

In the long run operation of the hydro power plants with reservoir and surge tank, it results from the relation (1) that the power generated by the turbine is maximum when the head H_n is maintained at maximum value, for variations of the river flow due to meteorological conditions. In this case, the power generated by the turbine will be only a function of the water flow trough the turbine.

M. Vinatoru is with the Department of Automation, University of Craiova, Faculty of Automation, 13 A. I. Cuza str., 200396 Craiova, Romania, Phone: (04) 0251-437541, Fax: (04) 0251-437541, (e-mail: vinatoru@automation.ucv.ro), WWW: http://www.ace.ucv.ro.

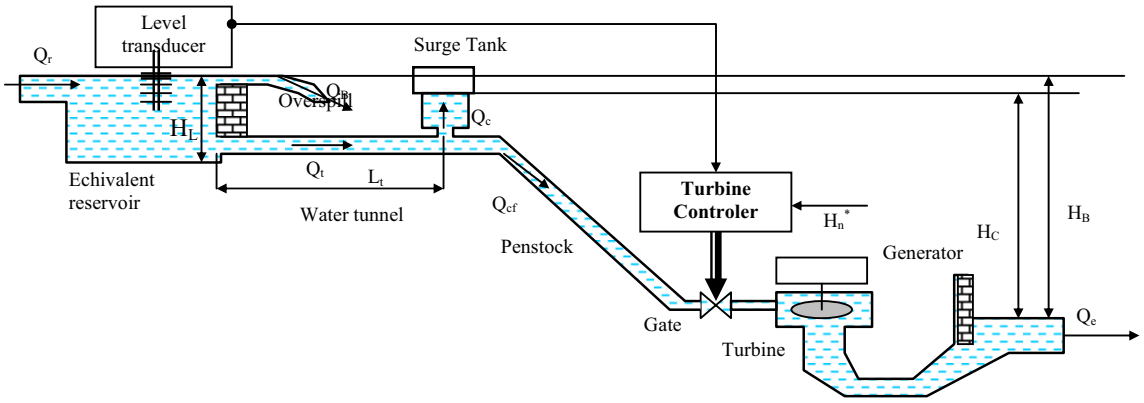


Figure 1. Simplified model of the hydropower plant

The set-point for the water level controller is generated by the monitoring system, has a certain variation profile and is a function of the required power:

$$H_n^* = \frac{P_n^*}{\eta Q_t} \tag{2}$$

In figure 1 is presented the simplified hydraulic diagram of a hydropower plant with reservoir and surge tank. Using this structure, a case study is presented, regarding the stability of the control structure.

A. Mathematical modelling of the control system

For the hydro power plant with reservoir and surge tank, (simplified diagram presented in figure 1), we will consider the followed areas with mass and energy accumulations (reservoir, water tunnel, surge tank, penstock and hydro-turbine). For these dynamic elements we developed the non-linear model presented in the following equations [1, 2].

a) The continuity equation for the water in the reservoir

$$A_L \cdot \frac{dH_B}{dt} = Q_r - Q_t - Q_w \tag{3}$$

b) The energy balance equation for the water tunnel between the reservoir and the surge tank:

$$\frac{L_t}{gA_t} \cdot \frac{dQ_t}{dt} + C_t |Q_t| Q_t + C_C |Q_c| Q_c = H_B - H_c \tag{4}$$

where $\frac{L_t}{gA_t} \cdot \frac{dQ_t}{dt} = \frac{L_t}{g} \cdot \frac{dv_t}{dt}$ is the inertial force of the water mass in the tunnel expressed in hydrostatic units, $C_t/Q_t/Q_t$ and $C_t/Q_c/Q_c$ are the hydraulic pressure loss due to friction with the tunnel walls, respectively due to the cross-section reduction at the entrance in the surge tank, H_B-H_C is the driving force that moves the water in the tunnel and the surge tank.

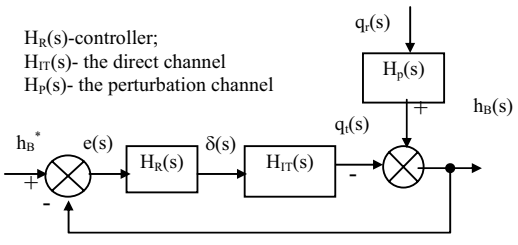


Figure 2. The block diagram of level controller

c) The continuity equation for the surge tank

$$A_c \cdot \frac{dHc}{dt} = Q_c = Q_t - Q_{cf} \tag{5}$$

d) The energy equation for the penstock and the turbine

We can neglect the water mass accumulation. Therefore, from the energy balance equation, the behaviour of the penstock and the turbine can be described by the water flow computed from the following relation:

$$Q_v = x \cdot Q_o \sqrt{\frac{H_c + C_c |Q_c| Q_c}{H_o}} \tag{6}$$

In these equations the variables have the following meanings -A_L is the area of the horizontal surface of the equivalent reservoir, considered as having a constant depth H_L. $H_L.A_L=V_{real}$, Q_r is the river water flow, Q_W is the discharging overspill water flow, L_t is the length of the tunnel between the reservoir and the surge tank, A_t is the cross-section of the tunnel, Q_t is the water flow in the tunnel, Q_c is the surge tank water flow, H_B is the water height in the reservoir, H_C is the water height in the surge tank, A_C is the cross-section of the surge tank, Q_{cf} is the water flow through the penstock, H_0 is the water level in the surge tank when the gate is completely open, Q_0 is the water flow for a completely open gate and $0 \le x \le 1$ is the opening factor for the gate (controller output).

e) The equation of the controller

The control system structure for the water level in the reservoir is presented in figure 2. It is recommended to use a PI or PD controller. In our case we will use a PI controller. In this control system, the water level in the reservoir is compared with a set-point determined by the operational conditions. The equation of the PI controller can be written in differential form:

$$\frac{dx}{dt} = \frac{K_R}{T_i}(H_f - H_{ref}) + K_R \frac{d(H_f - H_{ref})}{dt} \quad (7)$$

where: H_{ref} is the set-point level, H_f is the water level in the reservoir, K_R, T_i are the proportional constant and the integral constant of the PI controller.

B. Stability analysis

In the equations (3)-(7), which describe the dynamic behaviour of the control system for the hydropower unit, some parameters such as loss coefficients C_t and C_c can change due to different flow regimes and due to building parameters of the feeding system. Therefore is necessary to study the system stability in the operation point (small stability in Lyapunov way) and to determine the stability domain in the operational parameters plane.

Equations (3)-(7) will be linearised around the operation point, with non-dimensional relative variation of variables: q_t, h_B, h_c, δ and q_r (Q_{r0}=Q_{t0}):

$$\begin{aligned} q_t &= \frac{Q_t - Q_{t0}}{Q_{t0}}; \quad h_B = \frac{H_B - H_{B0}}{H_{B0}} \\ h_c &= \frac{H_c - H_{c0}}{H_{c0}}; \delta = \frac{x - x_0}{x_0}; q_r = \frac{Q_r - Q_{r0}}{Q_{r0}} \end{aligned} \quad (8)$$

We will introduce the following supplementary reference parameters, $\beta = \frac{x_0 T_i}{H_{ref}}$; $p = \frac{C_t Q_{t0}^2}{H_{c0}}$, (the slope of the controller output for a step variation of the error and the ratio between the losses on the feeding pipes and the hydraulic pressure at the base of the surge tank).

According with the previous relations and considering that the spill-over flow is zero (Q_B=0), following linearisation of equations (3)-(7), the linear state equations, expressed with non-dimensional variables defined in (8), can be written in operational Laplace forms:

$$\begin{aligned} h_B(s) &= -\frac{1}{sT_L} \cdot q_t(s) + \frac{1}{sT_L} \cdot q_r(s); \\ q_t(s) &= -\frac{2p}{sT_t} \cdot q_t(s) + \left(\frac{p+1}{sT_t}\right) \cdot h_B(s) - \frac{1}{sT_t} \cdot h_c(s) \end{aligned} \quad (9)$$

$$h_c(s) = \frac{1}{sT_c} \cdot q_t(s) - \frac{1}{2sT_c} \cdot h_c(s) - \frac{1}{sT_c} \cdot \delta(s) \quad (10)$$

$$\delta(s) = \frac{K_I + K_R s}{s} \cdot h_r(s) - \frac{K_I + K_R s}{s} \cdot h_{BT}(s) \quad (11)$$

where: $T_t = \frac{L_t \cdot Q_{t0}}{g \cdot H_{s0} \cdot A_t}$, $T_c = \frac{H_{c0} \cdot A_c}{Q_{t0}}$, $T_L = \frac{H_{ref} \cdot A_L}{Q_{t0}}$

These equations can be processed to obtain the transfer functions on the direct channel and on the perturbation channel.

$$h_B(s) = -\frac{T_t \cdot s + 2p}{P(s)} \cdot \delta(s) + \frac{(T_t \cdot s + 2p) \cdot H_n(s)}{P(s)} \cdot q_r(s) \quad (12)$$

The characteristic polynomial of the open-loop system is:

$$\begin{aligned} P(s) &= 2T_L T_c T_t^2 \cdot s^4 + T_L T_t (8pT_c + T_t) \cdot s^3 + \\ &+ \left[8p^2 T_c T_L + 2(2p+1) T_L T_t + 2(p+1) T_c T_t\right] \cdot s^2 + \\ &+ \left[4p(p+1)(T_L + T_c) + (p+1) T_t\right] \cdot s + (2p^2 + 3p + 1) \end{aligned} \quad (13)$$

Replacing (11) in (12), after several simple transforms, it results the close-loop system equation:

$$\begin{aligned} h_B(s) &= \frac{(T_t s + 2p)(K_I + K_R s)}{s \cdot P(s) + (T_t s + 2p)(K_I + K_R s)} \cdot h_r(s) + \\ &+ \frac{s \cdot (T_t s + 2p) \cdot H_n(s)}{s \cdot P(s) + (T_t s + 2p)(K_I + K_R s)} \cdot q_r(s) \end{aligned} \quad (14)$$

The characteristic polynomial of the closed-loop system is:

$$\begin{aligned} L(s) &= s \cdot P(s) + (T_t s + 2p)(K_I + K_R s) = \\ &= 2T_L T_c T_t^2 \cdot s^5 + T_L T_t (8pT_c + T_t) \cdot s^4 + \\ &+ \left[8p^2 T_c T_L + 2(2p+1) T_L T_t + 2(p+1) T_c T_t\right] \cdot s^3 + \\ &+ \left[4p(p+1)(T_L + T_c) + (p+1) T_t + T_t K_R\right] \cdot s^2 + \\ &+ (T_t K_I + 2pK_R + 2p^2 + 3p + 1) s + 2pK_I \end{aligned} \quad (15)$$

For an exact analysis of the hydropower unit dynamic behaviour it can be considered the example of a hydropower plant with reservoir and surge tank having a low installed power but a high storage capacity in the reservoir.

Experiment. *The hydropower system parameters: The water volume in the reservoir V_L=4,8.10^6 m^3; -The equivalent depth of the reservoir (considered constant) H_L=60m; The equivalent reservoir surface area is A_L=8.10^{4}m^2; The length and diameter of the pipe L_t=9650m, D_t=3,6m and the cross-section A_t=10m^2; The surge tank has a diameter D_c=5,4m with a cross-section A_c=23m^2; The gross nominal head H_B=260m, H_{Bmax}=266m, H_{Bmin}=245m; The gross nominal head at the surge tank H_{c0}=230m; The nominal flow Q_{t0}=36m^3/s; The length of the penstock L_{cf}=205m.*

Using the previous data, the time constants and equations' coefficients can be determined using relation (11):

$T_t = 15$; $T_c = 150$; $T_L = 6.10^5 s$, $p = 0{,}13$; $C_t = 0{,}023$

$$Q_{v0} = Q_{t0} = x_0 Q_0 \sqrt{\frac{H_{c0}}{H_0}}; \Rightarrow \frac{x_0 Q_0}{\sqrt{H_0}} = \frac{Q_{t0}}{\sqrt{H_{c0}}} = \frac{36}{\sqrt{230}} = 2{,}34$$

The steady state value for the controller output x_0 is determined using the maximum flow through the penstock Q_{vmax}=56m^3/s and is obtained for x_0=1:

$x_0 = 0{,}64$; iar $Q_0 / \sqrt{H_0} = 3{,}7$.

$$\begin{aligned} L(s) &= 405.10^8 s^5 + 153.10^7 s^4 + 348{,}53.10^5 s^3 + (35{,}27.10^4 + \\ &+ 15.K_R) s^2 + (15.K_I + 0{,}26.K_R + 4238) s + 0{,}26.K_I \end{aligned} \quad (16)$$

Using the stability criteria Routh-Hurwitz for the

characteristic polynomial (15, 16) we obtain a series of inequalities for the tuning parameters K_i and K_R as follows:

- from the block diagram of the control system presented in figure 2, it can be seen that the transfer coefficient on the direct channel is negative, $K_R<0$;
- from the inequality $a_0=0,26.K_I>0$ it results that the integral parameter shall be positive;
- from the Hurwitz determinant of second order results: $a_0a_1 =0,26K_I(15K_I+0,26K_R+4238)>0, \Rightarrow \lfloor K_R \rfloor<57,69K_I+16300$

which gives us a very large range for the tuning parameters of the controller;

- from the Hurwitz determinant of third order results:

$$K_I<(1495*10^7+152270*K_R+3,9*K_R^2)/(3072*10^3-225*K_R)$$

Solving previous inequality graphically, we obtain $K_R\in(0\ -100)$, $0<K_i<(4866\ -4826)$ and again, the range for the tuning parameters is large.

During the real operation of the hydropower unit, even if the control system remains stable, the oscillations in the hydraulic unit shall be avoided since they can generate huge over pressure in the penstock. In order to highlight these oscillations, a simulation of the hydropower unit was performed using different values for the tuning parameters. The block diagram in figure 2 was used for simulation. This diagram was preferred for simulation since it allows highlighting the variation of some specific parameters of the system, such as the level in the surge tank and the flow variation in the penstock Q_t. The simulation results are represented in the figures 3 and 4. In figure 3 is presented the system output for a 10% variation of the set-point for the reservoir water level, with no limitation on the flow control channel. Oscillations of the flow can be observed, due to the big differences between the time constants in the simulation scheme. To avoid these flow oscillations, the variation of the flow control output is limited in order to avoid big differences between the subsequent commands sent to the gate. The simulation results are presented in figure 4.

III. MODELLING OF THE HYDRAULIC SYSTEM FOR RUN-OF-THE-RIVER HYDROPOWER PLANTS

These types of hydropower plants have a low water storage capacity in the reservoir; therefore the plant operation requires a permanent balance between the water flow through turbines and the river flow in order to maximize the water level in the reservoir for a maximum efficiency of water use. Next, we will determine the mathematical model for each component of the hydropower system.

a) Hydraulic turbine. The hydraulic turbine can be considered as an element without memory since the time constants of the turbine are much smaller than the time constants of the reservoir, penstock, and surge chamber, if exists, which are series connected elements in the system.

As parameters describing the mass transfer and energy transfer in the turbine we will consider the water flow through the turbine Q and the momentum M generated by the turbine and that is transmitted to the electrical generator. These variables can be expressed as non-linear functions of the turbine rotational speed N, the turbine gate position Z, and the net head H of the hydro system.

$$Q = Q(H, N, Z) \quad M = M(H, N, Z) \tag{17}$$

Through linearization of the equations (17) and (18) around the steady state values, we obtain:

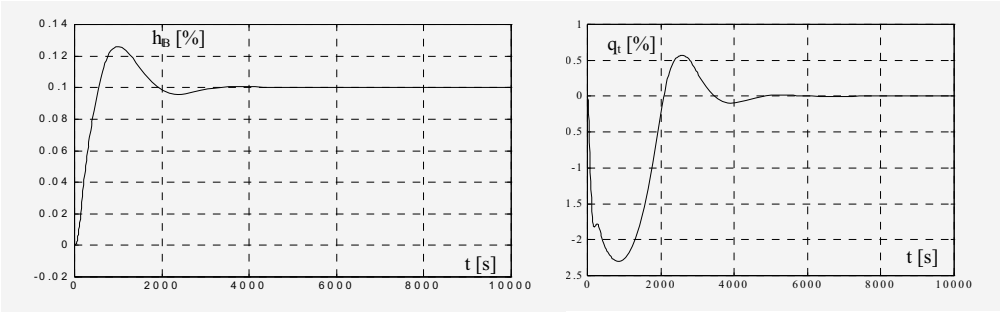


Figure 3. The response of the control system for a 10% variation of the set-point

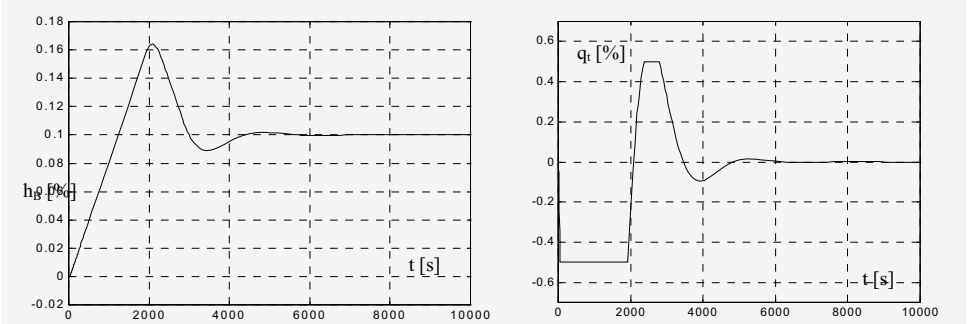


Figure 4. The response of the control system with flow limitation

$q=a_{11}\,h + a_{12}\,n + a_{13}\,z \quad m=a_{21}\,h + a_{22}\,n + a_{23}\,z$ (18)

where the following notations were used: $q=\Delta Q/Q_0$, $n=\Delta N/N_0$, $m=\Delta M/M_0$, $h=\Delta H/H_0$, $z=\Delta Z/Z_0$ which represent the non-dimensional variations of the parameters around the steady state values.

b) The hydraulic feed system. The hydraulic feed system has a complex geometrical configuration, consisting of pipes or canals with different shapes and cross-sections. Therefore, the feed system will be considered as a pipe with a constant cross-section and the length equal with real length of the studied system. In order to consider this, it is necessary that the real system and the equivalent system to contain the same water mass.

The dynamic pressure loss can be computed considering the inertia force of the water exerted on the pipe cross-section:

$$F_i = -ma = -LA\rho\, a = -AL\rho \frac{dv}{dt} = -AL\rho \frac{dQ}{dt} \qquad (19)$$

where L is the length of the penstock or the feed canal, A is the cross-section of the penstock, ρ is the specific density of water (1000Kg/m^3), a is the water acceleration in the equivalent pipe.

The dynamic pressure loss can be expressed as:

$$H_d = \frac{F_i}{A} = -L\rho \frac{dv}{dt} = -AL\rho \frac{dQ}{dt} \qquad (20)$$

or in non-dimensional form:

$$h_d = -T_w \frac{dq}{dt} \qquad h_d(s) = -s\, T_w\, q(s) \qquad (21)$$

where T_W is the integration constant of the hydropower system and the variables have the following meaning:

$$h_d = \frac{\Delta H_d}{H_{d0}}, \quad q = \frac{\Delta Q}{Q_0}, \quad T_w = \frac{Q_0}{H_{d0}} \cdot \frac{\rho L \cdot \sum l_i}{\sum l_i A_i} \qquad (22)$$

It must be noted that this is a simplified method to compute the hydraulic pressure loss, which can be used for run-of-the-river hydropower plants, with small water head. If an exact value of the dynamic pressure is required, then the formulas presented in [8], sub-chapter 8.4 "The calculation of hydro energy potential" shall be used. Replacing (21) in (19) and (20) and doing some simple calculations, we obtain:

$$q(s) = \frac{a_{12}}{1 + a_{11}T_w s} n(s) + \frac{a_{13}}{1 + a_{11}T_w s} z(s) \qquad (23)$$

$$h_d(s) = -\frac{a_{12}T_w s}{1 + a_{11}T_w s} n(s) - \frac{a_{13}T_w s}{1 + a_{11}T_w s} z(s) \qquad (24)$$

$$m(s) = \left(a_{21} - \frac{a_{12}T_w s}{1 + a_{11}T_w s}\right) n(s) + \left(a_{23} - \frac{a_{13}T_w s}{1 + a_{11}T_w s}\right) z(s) \qquad (25)$$

The mechanical power generated by the turbine can be calculated with the relation $P=\eta.\gamma.Q.H$ (see [8] sub-chapter 8.5 "Hydraulic turbines"), and $P=M.\omega=2\pi.M.N$, which can be used to obtain the linearized relations for variations of these values around the steady state values:

$$p = \eta.g.Q_o.h + \eta.g.H_o.q \qquad (26)$$

$$n = \frac{2\pi N_o}{P_o} p - \frac{(2\pi N_o)^2}{P_o} m \qquad (27)$$

where η is the turbine efficiency, and γ, Q, and H were defined previously, $P_0=M_0.\omega_0$ is the steady state power generated by the turbine for a given steady state flow Q_0 and a steady state head H_0, and N_0 is the steady state rotational speed. Using these relations, the block diagram of the hydraulic turbine, for small variation operation around the steady state point, can be determined and is presented in figure 5.

For an ideal turbine, without losses, the coefficients a_{ij} resulted from the partial derivatives in equations (23 - 25) have the following values: $a_{11}=0,5$; $a_{12}=a_{13}=1$; $a_{21}=1,5$; $a_{23}=1$. In this case, the transfer functions in the block diagram are given by the following relation:

$$H_{qn} = H_{qz} = \frac{1}{1 + 0,5.T_w s}, \; H_{hn} = -\frac{T_w s}{1 + 0,5.T_w s} \qquad (28)$$

$$H_{hz} = -\frac{T_w s}{1 + 0,5.T_w s}, \; H_{mn} = \left(1,5 - \frac{T_w s}{1 + 0,5.T_w s}\right) \qquad (29)$$

$$H_{mz}(s) = \left(1 - \frac{T_w s}{1 + 0,5.T_w s}\right) = \frac{1 - 0.5.T_w s}{1 + 0,5.T_w s} \qquad (30)$$

Experiment. *Let consider a hydroelectric power system with the following parameters: Water flow (turbines): $Q \in (500 \div 1000)$ m^3/s, $Q_N=725$ m^3/s; Water level in the reservoir: $H \in (17 \div 38)$ m, $H_N=30$ m; -The equivalent cross-section of the penstock $A=60$m^2; -Nominal power of the turbine $P_N=178$MW$=178.000$kW; -Turbine efficiency $\eta=0,94$; -Nominal rotational speed of the turbine N=71,43 rot/min; -The length of the penstock $l=\Sigma l_i=20$m;*

It shall be determined the variation of the time constant T_W for the hydro power system.

For the nominal regime, using relation (12), where $\Sigma l_i=20$m, the time constant of the system is: $T_w = 0,82$ s.

Next we will study the variation of the time constant due to the variation of the water flow through the turbine for a constant water level in the reservoir, H=30m, as well as the variation due to the variable water level in the reservoir for a constant flow Q=725 m^3/s.

In table I, column 3 and figure 2 a) are presented the values and the graphical variation of the time constant T_W for the variation of the water flow between 500 m^3/s and 110 m^3/s, for a constant water level in the reservoir, H=30m. In table 8.12 column 4 and figure 2 b) are presented the values and the graphical variation of the time constant T_W for the variation of the water level in the reservoir, for a constant water flow, Q=725m^3/s.

It can be seen from the table or from the graphs that the time constant changes more than 50% for the entire operational range of the water flow through the turbine or if the water level in the reservoir varies. These variations will create huge problems during the design of the control system for the turbine, and robust control algorithms are recommended.

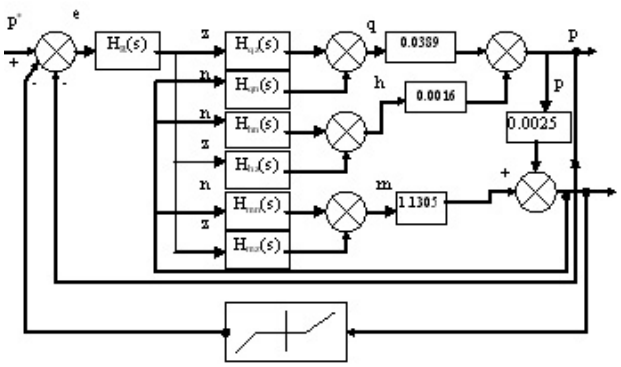


Figure 5. Control system for experiment

Table I. Variations of the time constant of the hydro system

H	Q	Tw(H=30m)	Tw(Q=725mc/s
17	1135,46	1,286065	1,449101
20	965,14	1,093155	1,231736
23	839,25	0,950569	1,071075
26	742,42	0,8408882	0,947489
29	665,62	0,7538998	0,849473
32	603,21	0,6832217	0,769835
35	551,51	0,6246598	0,703849
38	507,97	0,5753446	0,648282

In figure 5 is presented the block diagram of the turbine's power control system, using a secondary feedback from the rotational speed of the turbine. It can be seen from this figure that a dead-zone element was inserted in series with the rotational speed sensor in order to eliminate the feedback for ±0,5% variation of the rotational speed around the synchronous value. The constants of the transfer functions had been computed for a nominal regime T_W=0,8s. The optimal parameters for a PI controller are: K_R=10, T_I=0,02s. The results of the turbine simulation for different operational regimes are presented in figure 6, for a control system using feedbacks from the turbine power and rotational speed, with a dead-zone on the rotational speed channel for ±0,5% variation of the rotational speed around the synchronous value (a) Power variation with 10% around nominal value, b) Rotational speed variation for power control).

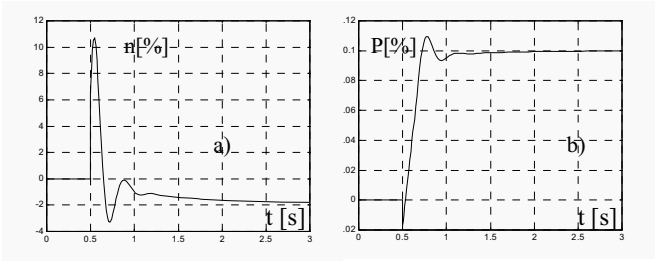


Fig. 6. Turbine Control simulations with feedbacks from turbine power and rotational speed

In figure 7 are presented the variations of the turbine power (graph a) and rotational speed (graph b) for the control system using a feedback from the turbine power but no feedback from the rotational speed .

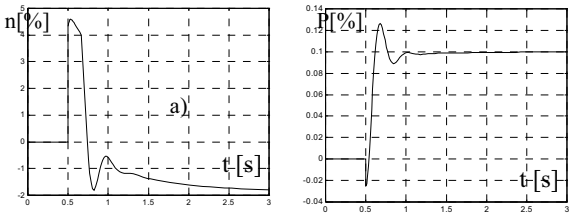


Fig. 7 Control structure with only power feedback,
a) Power variation with 10% around nominal value,
b) Rotational speed variation for power control

This oscillation has no significant influence on the performance of the system but would have lead to permanent perturbation of the command sent to the turbine gate.

IV. CONCLUSIONS

The possibility of implementation of digital systems for monitoring and control for power, frequency and voltage in the cascade hydro power plant was discussed. The simplified mathematical models, capable to accurately describe dynamic and stationary behaviour of the hydro units were developed and simulated and these results were compared with the experimental results. Finally, a practical example was used to illustrate the design of the controller and to study the system stability.

REFERENCES

[1] G., D Ferrari-Trecate, Mignone, D. Castagnoli, M. Morari, Hybrid *Modelling and Control of a Hydroelectric Power Plant*, CH –0802, Institut fur Automatik, ETH- Zurich, (2003).

[2] C. Henderson, *Yue Yang Power Station – The Implementation of the Distributed Control System,* GEC Alsthom Technical Review, Nr. 10, 1992.

[3] G. I. Krivchenko, *Hydraulic machines: Turbine and Pumps,*2nd ed. ISBN 1-56670-001-9, CRC Press, London (1994).

[4] M. Vînătoru, E. Iancu, C. Vînătoru, *Control, Monitoring and Protection of the Turbine and Generator System*, International Symposium on System Theory, Robotics, Computers and Process Informatics, SINTES 9, Craiova , (1998).

[5] O. F. Jimenez, M. H. Chaudhry Water level Control in Hydropower Plants, Journal of Energy Engineering, Vol118, No. 3 Dec. 1992.

[6] H. Weber, V. Fustik, F. Prillwitz, A. Iliev, Practically oriented simulation model for the Hydro Power Plant "Vrutok" in Macedonia, Balkan Power Conference, 19. – 21.06.2002, Belgrade

[7] I.C.E Felix -Bucharest S.A, *SCADA System for Hydropower cascade from Midle Olt,* http:// www.felix.ro/apl.html.

[8] M. Vinatoru, *Industrial processes control, vol. II,* Ed. Universitaria Craiova, 2005.

Mathematical model proposed for simulating the influence on air quality of the three ash dumps that affect Craiova, Romania

Gabriel Vladut, Monica Mateescu, Liana Simona Sbirna and Sebastian Sbirna

***Abstract*—Monitoring air quality has become a significant factor in the environment management, with increasing interest in health protection. The present study is focused on the local environment impact of the three ash dumps which affect the air quality in Craiova (one of the major cities of Romania): CET I (Isalnita), CET II (Simnic) and CET Turceni, from Gorj county. Interest is invested in predicting the average concentration of dangerous particulate matter with a diameter less than 10 μm at various distances from an air pollution source, such as an ash dump. In this paper, we propose a theoretical model to simulate the air pollution with PM10 caused by such a point source.**

I. Introduction

AIR pollution is a major environmental risk to health and it is estimated to cause approximately two million premature deaths worldwide per year. Exposure to air pollutants is largely beyond the control of individuals and requires action by public authorities at national, regional and even international level.

II. About PM10

SUSPENDED particulate matter (PM10) is a dangerous urban pollutant. It is a complex mixture of particles of various origin and chemical composition, smaller than 10 μm in aerodynamic diameter, usually consisting of smoke and dust from industrial processes, incineration of refuse, heat and power generation, road traffic, construction, agriculture, as well as plant pollen and other natural sources.

PM10 is considered a serious air pollution problem in urban agglomerations, because of its detrimental impact on human health and living standards.

The European Union has established air quality standards for PM10, in order to improve the quality of ambient air.

The former Directive 30/1999, as well as the later Directive 50/2008 state that since January 1st 2005 the daily average values of PM10 concentration may not exceed 50 μg/m^3 in more than 35 days a year and the annual average value of PM10 concentration may not exceed 40 μg/m^3.

Manuscript received June 17, 2010.

G. Vladut is with the S. C. IPA CIFATT Craiova company, 12 Stefan cel Mare Street, Craiova. He is an IEEE Member (e-mail: office@ipacv.ro).

M. Mateescu is with the Faculty of Chemistry, University of Craiova, which is situated on 107I Calea Bucuresti Street, Craiova (e-mail: monica.daniela.mateescu@gmail.com).

L. S. Sbirna is with the Faculty of Chemistry, University of Craiova, 107I Calea Bucuresti Street, Craiova (e-mail: simona.sbirna@gmail.com).

S. Sbirna is with Dolj Environment Protection Agency, 1 Petru Rares Street, Craiova (e-mail: s.sbirna@gmail.com, phone: 0721-329-779).

III. About Craiova

CRAIOVA is Romania's sixth largest city and capital of Dolj County. It is situated near the east bank of the Jiu river in the middle of Oltenia region at an average altitude of 100 m and it the most important city of Oltenia.

It is situated at 44°20′00″ North latitude and 23°49′00″ East longitude.

Specific relief is mainly plain. To the North a slight influence of the hills is observed, whereas to the South it exhibits meadow characteristics.

According to the last Romanian census, from 2002, there were 302 600 people living within the city of Craiova, making it the sixth most populous city in Romania.

It is a longstanding political center, located at approximately equal distances from the Southern Carpathians (North) and the Danube River (South).

IV. Monitoring PM10 in Craiova

WITHIN Craiova urban area, the air quality is continuously monitored. The concentration of different air pollutants is recorded by five modern monitoring station provided by the European Community, whose symbols are DJ1, DJ2, …, DJ5 (DJ is the abbreviation for Dolj county, whose capital city is Craiova).The results recorded by these stations are validated by experts from the Dolj Environment Protection Agency, who process the data.

PM10 concentration is recorded by four of these stations. As an example, in Fig. 1 we present a graphic showing the variation of this parameter during the year 2009 recorded by DJ3. We may observe that, sometimes, even values exceeding 200 μg/m^3 were recorded.

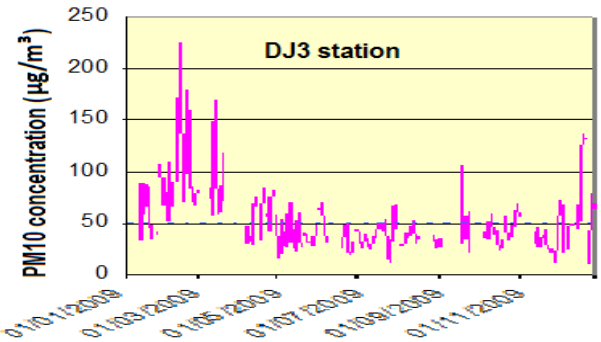


Fig. 1. PM10 concentration - data recorded at DJ3 station in 2009

In order to verify the results of our theoretical study, we must present a graphic that relates the PM10 concentration to the wind velocity.

Such a graphic is shown in Fig. 2.

The most reliable data are gathered by using air pollution information recorded by DJ3 and meteorological information provided by DJ 4 (they are situated at less than 10 km away from each other on the same road), because DJ4 incorporates a meteorological station, but unfortunately it is the only station that does not record PM10 concentration.

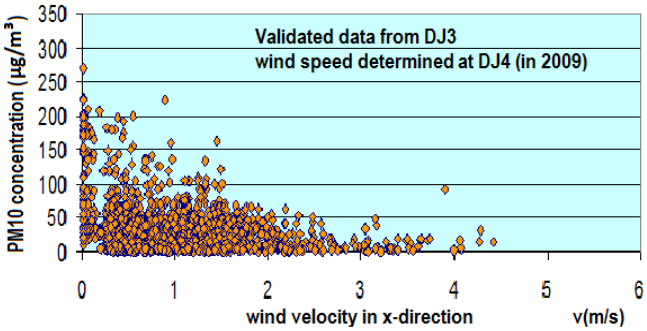


Fig. 2. Correlation between PM10 concentration measured at DJ3 and wind velocity in *x* direction, measured at DJ4

The atmospheric dynamism is a particular one: DJ4 station is located in the Jiu Valley corridor, being affected by air streams orientated from NW towards SE direction.

Craiova is submitted to the influence of three ash dumps, namely CET I (Isalnita), CET II (Simnic) and CET Turceni, from Gorj county.

This is the reason why, in this paper, a mathematical model is proposed for simulating the influence of a PM10 source at different spatial points.

V. Mathematical Model

As input data, we used validated values of different parameters provided by the Dolj Environment Protection Agency, whose continuously monitoring stations (all equipped with analogous instruments based on a modern technology, reporting hourly or daily data) measure PM10 level in Craiova, as well as meteorological parameters, mainly wind direction and wind velocity (the low average value of the wind velocity indicates a poor capacity of carrying away the dangerous particulate matter).

A tridimensional diffusion model has been developed for computing the concentration of PM10 originated from an ash dump (a particular point source that continuously disperses particles).

It is based on a partial differential equation which is constructed by using Fick's law. Under these circumstances, consideration of variable wind speed and variable dispersion coefficients is only possible by applying limiting assumptions.

In the model we propose, variation of PM10 concentration with distance obeys a logarithmic correlation near the source, as one may see in Fig. 3.

Unfortunately, no analytical method for solving diffusion equations considers the logarithmic profile of wind velocity.

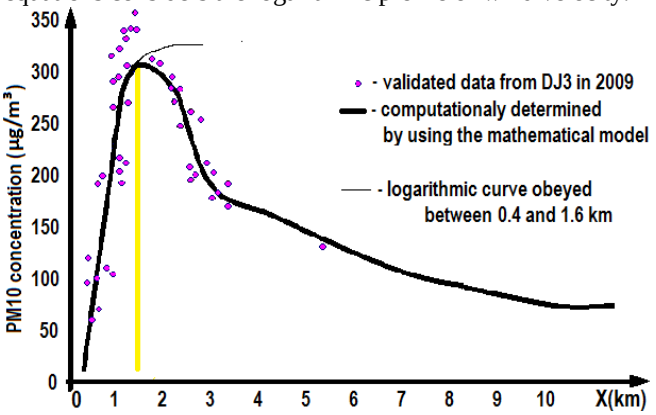


Fig. 3. Logarithmic variation of PM10 concentration on *x* direction

Thus, numerical solution methods including better and more complete atmospheric parameters within the model should be used (high-speed computers make a numerical method the best and most appropriate method for solving diffusion equations).

Any chemical reaction of the pollutants may be ignored in this analysis, but settling on the earth has to be considered.

Pollutants move horizontally in the wind direction (this direction is denoted as *x*, while X stands for the distance corresponding to the point at which a particle is carried away from the source).

PM10 particles diffuse into the atmosphere in the y and z directions, chosen so that (*x*, *y*, *z*) represents a cartesian coordinate system. With increasing distance from the source, dispersion profile changes.

Diffusion in wind direction is ignored.

Boundary conditions must be taken into account: particle concentration is considered zero before point source; particles settle on the ground with terminal velocity V; particle diffusion in the vertical *z* direction is ignored beyond the mixing height; particle penetration in the horizontal *y* direction is ignored after a certain distance, which may be calculated knowing the wind velocity.

The continuity equation for particles – taking diffusion into account – may be represented as:

$$v\frac{\partial[PM10]}{\partial x}-\frac{\partial}{\partial y}\left(D_y\frac{\partial[PM10]}{\partial y}\right)-\frac{\partial}{\partial z}\left(D_z\frac{\partial[PM10]}{\partial z}\right)=0 \qquad (1)$$

v being the wind velocity in *x* direction, [PM10] – the ambient particle concentration, D_y and D_z – the dispersion coefficients in *y* and *z* directions.

It shows that, in computing the continuity equation for predicting particle concentration, wind velocity and PM10 dispersion coefficients must be known.

"The finite volume method", incorporating the Powerlaw scheme (proposed by Patankar [1]) and a stretched grid in the *x* direction was employed to obtain the numerical solution of the continuity equation for particles.

To identify the dispersion coefficients and wind profile, the surface roughness and wind speed in the mixing height must first be determined.

VI. Results and Discussions

Since determining the surface roughness is difficult, in this study the surface roughness parameter is considered to be 0.05 m (based on local topographical conditions).

Regarding atmospheric stability, the particle diffusion domain in the *y* direction is reached at about 2000 m.

For all the results presented, the height of the mixing layer is 250 m, whereas the height of the surface layer is 80 m.

Thus, with *dy*=20 m and *dz*=10 m, there are 100 elements in the *y* and *z* directions, respectively.

Generally, each section will involve 25×100 cells (i.e., 2500 cells). The domain of particle dispersion in the *x* direction is 10000 m.

To reduce the time required for calculation, the sizes of *dx* will be diverse, i.e., in a point situated close to the ash dump, where rapid changes of concentration occur, values of *dx* will be lower, while in a point situated far away from it they will be higher.

Instead of directly solving the continuity equation, the atmosphere is considered to be divided into small elements. A mass conservation equation is written for each element [2].

A small volume of fluid, with dx, dy and dz dimension, is considerate as unit cell [3], which is located in point (a, b, c).

PM10 concentrations are supposed to be known in points (a-1,b,c), (a+1,b,c), (a,b-1,c), (a,b+1,c), (a,b,c-1), (a,b,c+1), but the concentration is supposed to be unknown in (a, b, c).

Thus, a mass conservation equation for a fluid element in point (a, b, c) may be written in order to determine it:

$$v_a dydz[PM10]_{a-1.b.c} - v_a dydz[PM10]_{a.b.c} = \frac{(D_y)_a dxdz}{dy}[PM10]_{a.b.c} + \frac{(D_z)_{a-1/2} dxdy}{dz}[PM10]_{a.b.c} + \frac{(D_z)_{a+1/2} dxdy}{dz}[PM10]_{a.b.c} - \frac{(D_y)_a dxdz}{dy}[PM10]_{a.b-1.c} \quad (2)$$
$$\frac{(D_y)_a dxdz}{dy}[PM10]_{a.b+1.c} - \frac{(D_z)_{a-1/2} dxdy}{dz}[PM10]_{a.b.c-1} - \frac{(D_z)_{a+1/2} dxdy}{dz}[PM10]_{a.b.c+1}$$

which may be rewritten:

$$[PM10]_{a.b.c} = A/B \quad (3)$$

where:

$$A = v_a dydz[PM10]_{a-1.b.c} + \frac{(D_y)_a dxdz}{dy}[PM10]_{a.b-1.c} + \frac{(D_y)_a dxdz}{dy}[PM10]_{a.b+1.c} + \frac{(D_z)_{a-1/2} dxdy}{dz}[PM10]_{a.b.c-1} + \frac{(D_z)_{a+1/2} dxdy}{dz}[PM10]_{a.b.c+1} \quad (4)$$

whereas

$$B = v_a dydz + \frac{(D_y)_a dxdz}{dy} + \frac{(D_z)_{a-1/2} dxdy}{dz} + \frac{(D_z)_{a+1/2} dxdy}{dz} \quad (5)$$

For the elements near the ground, diffusion toward the ground is not considered. Instead, it is assumed that pollutants settle on the ground with terminal velocity.

On the other hand,

$$\frac{\partial}{\partial z}\left(D_z \frac{\partial[PM10]}{\partial z}\right) = V \frac{\partial[PM10]}{\partial z} \quad (6)$$

V being the particle terminal velocity, calculated according to the formulae proposed by Crawford [4]:

$$V = \frac{\rho d^2 g}{18\eta} \quad (7)$$

where ρ is the particle density (kg·m^{-3}), d is the particle diameter (m), g is the gravity acceleration (m·s^{-2}) and η stands for the viscosity (kg·m^{-1}·s^{-1}).

VII. Graphical Representation

In order for the concentration profile obtained in the ground surface layer to be more clearly understood, a reliable graphical representaton is required. Different perspectives are shown in Fig. 4 and Fig 5 [5].

Both of them use a code of colors for designating the concentration values of PM10 (μg/m^3) – as one might see on the right side of each figure – and they are both given in the cartesian coordinates (*x*, *y*, *z*) - *x* and *y* representing the distances from the ash dump (*x* is the wind velocity main direction), whereas *z* represents the concentration of the particulate matter.

The first one (which seems to be bidimensional) shows the "map" seen from "above", i.e. from a point that is situated on the *z* axis.

The second one offers three perspectives of the concentration profile, as explained bellow:

(a) shows an usual perspective in the (*x*, *y*, *z*) cartesian coordinates;

(b) presents another perspective, obtained after an anticlockwise rotation of the coordinate system by a quarter of a turn about the *z* axis, represented by the matrix equation:

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} a_{11}a_{12}a_{13} \\ a_{21}a_{22}a_{23} \\ a_{31}a_{32}a_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} \quad (8)$$

the nonzero terms in the matrix being: $a_{12} = 1$, $a_{21} = -1$, $a_{33} = 1$;

c) gives one more perspective, obtained by rotating the cartesian system (*x*', *y*', *z*') by half a turn about the same axis, rotation which is represented by the equation:

$$\begin{pmatrix} x'' \\ y'' \\ z'' \end{pmatrix} = \begin{pmatrix} b_{11}b_{12}b_{13} \\ b_{21}b_{22}b_{23} \\ b_{31}b_{32}b_{33} \end{pmatrix} \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} \quad (9)$$

the nonzero terms in the matrix being: $b_{11} = b_{22} = -1$, $b_{33} = 1$.

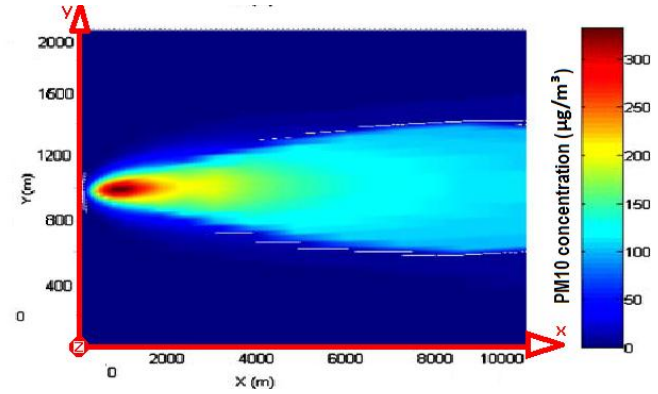
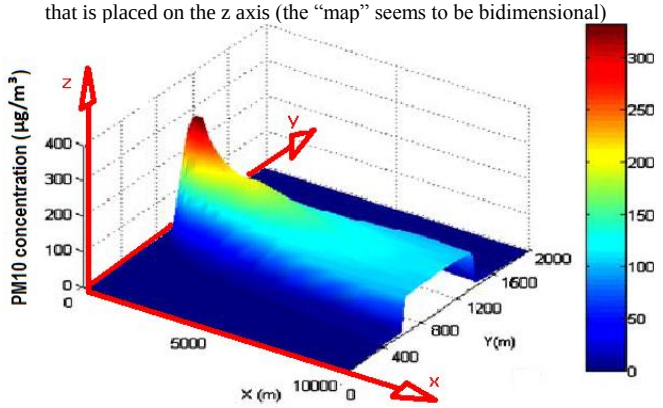
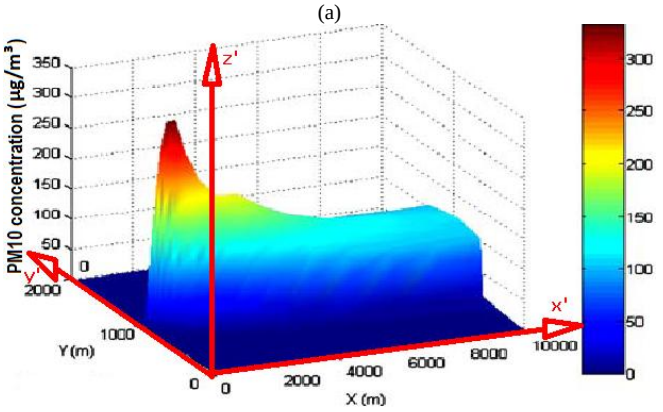


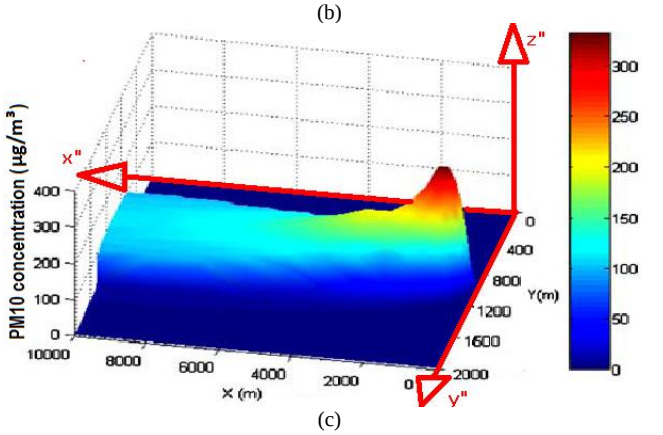
Fig. 4. View of the concentration profile obtained by an observer that is placed on the z axis (the "map" seems to be bidimensional)



(a)



(b)



(c)

Fig. 5. Other three perspectives on the concentration profile obtained:
(a) in the (x, y, z) cartesian coordinate system;
(b) in the (x, y, z) cartesian coordinate system;
(c) in the (x, y, z) cartesian coordinate system,
the cartesian systems (*x*, *y*, z), (*x*', *y*', *z*'), (*x*", *y*", *z*") being described above

VIII. Conclusion

THIS study has shown that it is necessary to monitor the air quality, especially the concentration of those pollutants that may cause major health injury.

A comparisson between practical and theoretical data is useful in predicting the air pollution, so that measures can be taken to diminish it [6]-[13].

The paper proposed a theoretical manner to simulate the way the urban air may be afected by a pollution source, such as an ash dump (Craiova is affected by the emissions from three ash dumps: CET I - Isalnita, CET II - Simnic and CET Turceni, from Gorj county).

It has been shown that PM10 daily average in the ambiental air of Craiova city at distances between 0.4 and 1.6 km from an ash dump is often higher than the European Union thershold of 50 µg/m³ (which may not be exceeded more than 35 times a year).

Therefore, urgent measures for reducing air pollution have to be taken in Craiova, as well as in other developed cities in Romania, for our country to get the exoneration from the European Union, as far as air pollution is concerned and, of course, for us to improve our own life quality.

As far as the model itself is concerned, the good agreement between the theoretical results and the measured data shows that this model can be a powerful tool in predicting PM10 concentrations.

IX. References

[1] V. S. Patankar, "*Numerical heat transfer and fluid flow*," New York: McGraw Hill Book Company, 1980.

[2] D. M. Moreira, M. T. Vilhena, D. Buske and T. Tirabassi, "The GILTT solution of the advection – diffusion equation for an inhomogeneous and nonstationary PBL," *Atmospheric Environment*, vol. 40, pp. 3186-3194, 2006.

[3] S. Baroutian, A. Mohebbi, and A. Soltani Goharrizi, "Measuring and modeling particulate dispersion: A case study of Kerman Cement Plant," *J. Hazardous Materials*, vol. 136, pp. 468-474, 2006.

[4] M. Crawford, "*Air pollution control theory*," New York: McGraw - Hill Book Company, 1976.

[5] E. Canepa, and C. F. Ratto, "Algorithms to simulate the transport of pollutant elements: A model validation exercise and sensitivity analysis", *Environmental Modeling and Software*, vol.18, pp. 365-372, 2003.

[6] M. Delgado, J. L. Verdegay and M. Vila, "A general model for fuzzy linear programming," *Fuzzy Sets Syst.*, vol. 29, pp. 21–29, 1989.

[7] D. Dubios and H. Prade, "Qualitative possibility theory and its applications to constraint satisfaction and decision under uncertainty," *Int. J. Intell. Syst.*, vol. 17, no. 1, pp. 45–53, 2002.

[8] G. H. Huang and N. B. Chang, "The perspectives of environmental informatics and systems analysis," *J. Environ. Inform.*, vol. 1, pp. 1–6, 2003.

[9] L. Liu, G. H. Huang, Y. Liu and G. A. Fuller, "A fuzzy-stochastic robust programming model for regional air quality management under uncertainty," *Eng. Optim.*, vol. 35, no. 2, pp. 177–199, 2003.

[10] I. A. Maqsood, "Two-stage interval-stochastic programming model," *J. Air Waste Management Assoc.*, vol. 53, pp. 540–552, 2003.

[11] M. Yeomans, "Combining simulation with evolutionary algorithms," *J. Environ. Inform.*, vol. 5, pp. 11–29, 2007.

[12] H. I. Jager HI, A. W. King, N. H. Schumaker, T. L. Ashwood and B. Jackson, "Spatial uncertainty analysis of population models," *Ecol. Model*, vol. 185, pp. 13–27, 2005.

[13] V. Guttal and C. Jayaprakash, "Spatial variance and spatial skewness: leading indicators of regime shifts in spatial ecological systems," *Theor. Ecol.*, vol. 11, pp. 450–460, 2009.

INDEX OF AUTHORS

LIST OF REVIEWERS

Waseem ABBAS
Mihail ABRUDEAN
Paweł ADAMCZYK
Cristian AMARANDEI
Florin ANTON
Alexandru ARCHIP
Vadim AZHMYAKOV
Costin BADICA
Marcelin Iulian BALASCA
Peter BARANYI
Marian BARBU
Andrzej BARTOSZEWICZ
Theodor BORANGIU
Corneliu BOTAN
Antoneta BRATCU
Marius BREZOVAN
Adrian BURLACU
Cristian BUTINCU
Simona CARAIMAN
Sergiu CARAMAN
Petru CASCAVAL
Emil CEANGA
Daniela CERNEGA
Dorian COJOCARU
Mirel COSULSCHI
Mitica CRAUS
Vladimir CRETU
Daniela DANCIU
Catalin DIMON
Radu DOBRESCU
Viorel DUGAN
Eva-Henrietta DULF
Alexandru DUMITRACHE
Eugen DUMITRASCU
Teodor DUMITRIU
Tiberius DUMITRIU
Nicolae Iulian ENESCU
Heinz ERBE
Lavinia FERARIU
Laurentiu FRANGU
Gerhard FREILING
Emilia FRIDMAN
Eugen GANEA
Cornelia GORDAN
Adriana GRAVA
Cristian GRAVA
Nguyen HOAI NAM
Daniela HOSSU
Eugen IANCU
Radu IBANESCU
Augustin Iulian IONESCU
Cosmin IONETE
Dan ISTRATE
Nick IVANESCU
Alexander KAMACHKIN
Vladimir L. KHARITONOV
Marius KLOETZER
Laszlo KOCZY
Peter KORONDI
Corneliu LAZAR
Mircea LAZAR
Gheorghe LAZEA
Florin LEON
Rogelio LOZANO
Cristian MAHULEA
Nicolae MARASESCU
Gheorghe MARIAN
Constantin MARIN
Mihaela-Hanako MATCOVSCHI
Hamid MEDJAHED
Letitia MIREA
Mihai MOCANU
Sabine MONDIE
Nicolae NEAGU
Ovidiu NEAMTU
Mircea NITULESCU
Aleksandra NOWACKA-LEVERTON
Sorin OLARU
Nejat OLGAC
Virginia Ecaterina OLTEAN
Doru PANESCU
Octavian PASTRAVANU
Stefan Gheorghe PENTIUC
Emil PETRE
Catalin PETRESCU
Rustem POPA
Dan Gh. POPESCU
Dorin POPESCU
Dumitru POPESCU
Elvira POPESCU
Mihai POSTOLACHE
Cosmin POTERAS
Radu-Emil PRECUP
Stefan PREITL
Andrei PRICOP
Dorina PURCARU
Michał PYTASZ
Pedro RODRIGUEZ-AYERBE
Monica ROMAN
Andrei RUSAN
Sergey RYVKIN
Belem SALDIVAR
Bogdan SAPINSKI
Adrian SCHIOP
Dan SELISTEANU
Dorin SENDRESCU
Razvan SOLEA
Veaceslav SPINU
Alexandru STANCU
Liana STANESCU
Radu STEFAN
Florin STINGA
Viorel STOIAN
Adrian-Mihail STOICA
Cristina STOICA
Florin STOICAN
Florentina UNGUREANU
Gabriela VARVARA
Patricio VELA
Erik VERRIEST
Matei VINATORU
Mihail VOICU
Yorai WARDI
Dan WU
Deryck YEUNG
Tony YEZZI
Alexei ZHABKO

LOCAL PARTNERS AND SPONSORS

PARTNERS

ANCS
Autoritatea Nationala pentru Cercetare Stiintifica
(National Authority for Scientific Research)

ROMTEK SRL

SPONSORS

IBM Romania

SOFTRONIC Romania

DOLSAT CONSULT

SMC Romania

SINTEC MEDIA Romania

TOP EDGE Romania

POLISEA Romania

ISSN 2068 – 0465