ANNALS OF THE UNIVERSITY OF CRAIOVA
Series: Automation, Computers, Electronics and Mechatronics, Vol. 14 (41), No. 1, 2017

Two Lessons on Recurrent Neural Networks
1. Basic Features and Architectures

Daniela Danciu*

** Department of Automation and Electronics, University of Craiova, A.I. Cuza, 13, Craiova, RO-200585, Romania (e-mail: ddanciu@automation.ucv.ro).*

Abstract: The main idea of this survey split into two lectures is motivated by the intensive and extensive development of the Recurrent Neural Networks (RNNs) research branch of the Artificial Intelligence (AI) domain. Due to their cyclic interconnections, RNNs are Neural Networks (NNs) which involve dynamics. More specific, RNNs can have very rich spatial and temporal behaviors which include fixed-point multiple equilibria, oscillations (self-oscillations, but also forced oscillations), time-delays, synchronization and even chaotic behaviors. For these reasons RNNs can be used to model complex cognitive functions such as associative memories, but also decision making, classification, sorting as well as formalized problem solving tasks. The first lecture presents RNNs with a special focus on their main features, the artificial neuron and the most used architectures. The second lecture will further discuss the RNNs from the points of view of the qualitative behavior (considering the local properties but also the global behavior) and their main applications.

Keywords: Recurrent Neural Networks, Hopfield networks, Cohen-Grossberg networks, KWTA neural networks, Cellular Neural Networks, Multiple equilibria

1. INTRODUCTION

The first "brick" at the foundation of the Artificial Neural Networks (ANNs) field but also at the Artificial Intelligence (AI) domain can be considered the book by the philosopher and psychologist Wiliam James [James (1890)]. He stated in his book the first principles of the correlated learning and also of the associative memory, almost suggesting the idea of the neuron's activity dependence by the sum of input signals [Eberhart and Dobbins (1990)]. The next milestone of the field, a half-century later, is the first model of the artificial neuron introduced by McCulloch and Pitts (1943). In 1949, D. O. Hebb brought four contributions to the field opening the way for another important direction in NNs and AI fields, that of learning algorithms. His contributions are as follows [Eberhart and Dobbins (1990)]: (a) the idea of connections – stating that within a NN "information is stored in the synapses weights", (b) the idea of proportional dependence of a synaptic weight learning rate by the product of activation values of the neurons; (c) the symmetry of the synaptic weights (which is not in accordance with the reality even if it is used in the Artificial NN field; (d) "the cell assembly theory which states [...] that if simultaneous activation of a group of weakly connected cells occurs repeatedly, these cells tend to coalesce into a more strongly connected assembly". The development of the ANNs field has knew further both evolution and stagnation periods until the beginning of the '80s of the XX century, when the background for the first implementation of a neural network has been established. This turning point in the history of the field marked the beginning of a creative effervescence in the developing of the NNs field but also of the AI domain.

Among the Artificial Intelligence devices, Neural Networks (NNs) are computational architectures that represent simplified versions of the biological brain from the point of views of structure, signalling, functionalities as well as signal processing. Neural networks provide a high flexibility in answering to the implementation requirements induced by a particular application, i.e. these structures can be implemented as hardware devices in VLSI technology, or as hybrid hardware-software devices or also as software applications embedding a certain level of parallelism.

All these NNs features recommend them for a new paradigm in information processing – by means of neural computers able to solve both non-formalized and formalized problems, as an alternative to the conventional von Neumann computers. As it is described by Galushkin (2010), neural computers can be either problem-oriented or universal ones and can be implemented as hybrid analog-digital learning machines such that: (i) the multidimensional operations on the threshold basis can be performed by the "fast" analog part, (ii) the neural algorithms for the adjustment of the NNs' synaptic weights can be implemented either in the "fast" analog form or in the "low-speed" form of either specialized digital circuits or software components.

The description of a neural network can be discussed at both the micro- and macro-levels. At the micro-level – the artificial neuron level – it is important to point out the neuron structure and its activation function. At the macro-level it is important to analyse the neural network topology, its qualitative properties (in case they exhibit dynamics) and their suitable applications. These key items shape the structure of this survey which focuses on the wide class of neural networks that embed feedback connections – the Recurrent Neural Networks (RNNs). Consequently, Section 2 of this paper presents the key features of the artificial neuron as well as of the main two topologies of the artificial neural networks while Section 3 considers the main RNNs structures. In Section 4 the qualitative behaviours of RNNs, viewed as nonlinear dynamical systems, are briefly discussed – this subject being one of the second lecture sections. Some concluding remarks will end the paper.

2. ARTIFICIAL NEURAL NETWORKS. MAIN CHARACTERISTICS

The computational power of the biological brain is a result of its "architecture". ANNs – as brain simplified versions – can be described as highly parallel structures of elementary processing units (artificial neurons) exhibiting emergent computational capabilities such as learning and generalization, but also fault tolerance and slow performance degradation due to distributed encoding and redundancy of information on a certain area of neurons and synapses [Kosko (1992)]. The artificial neurons are arranged in successive layers such that pieces of information from the previous layers – but also possibly from the next layers– are simultaneously processed/filtered (parallel computation) by the neurons of a certain layer. Such a parallel-sequential processing architecture of a neural network ensures the emergence of new computational capabilities, i.e. an increased and enriched computational power which is not specific to a single neuron. Nevertheless, the neuron structure and function are also key factors for acquiring these collective properties of the whole neural network regardless its type – natural or artificial.

2.1 The Artificial Neuron

Taking into account the main features of the biological neuron, an artificial neuron i has multiple inputs and only one output (Fig. 1). One can identify, as in the case of natural neuron, three basic regions:

i) the receiving region: the input signals x_j – which include the outputs of other neurons within the network $y = \overline{1.m}$ as well as either an external stimulus I_i or a bias – are collected and the neuron's *post-synaptic signal* is computed as a weighted sum of these inputs (pre-processing stage); the synaptic weights w_{ij} are real numbers showing the strength and the excitatory/inhibitory effect to the i^{th} neuron, i.e. $w_{ij} > 0$ shows an excitatory effect of the input x_j while its effect will be inhibitory for $w_{ij} < 0$;

ii) the processing region: the post-synaptic signal is processed via a certain neuron *activation function f* and, as a result, an *action potential* can be released;

iii) the transmission region: the neuron output signal y, i.e. its action potential, is further transmitted via synaptic connections to other neurons within the network but also, possible, to the neuron itself (*self-feedback*).

Usually, the output of an artificial neuron is computed as

$$y = f(r) = f\left(\sum_{j=0}^{m} w_j x_j\right) \tag{1}$$

but also more complex architectures and formulae exist, these incorporating self-feedback connections at the synapses level or at the activation function level or at the output level as well – for more details, the reader is sent to the paper by Tsoi and Back (1994) and the references therein.

The activation (transfer) function f has as main role the limitation of the domain of variation for the neuron output signal to a pre-specified domain, performing at the same time a specific operation on the post-synaptic signal r such as filtration or selection to enumerate just a few.

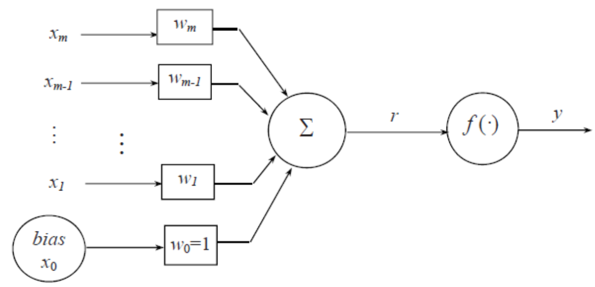

Fig. 1. The basic structure of an artificial neuron.

The most used neuron activation functions, as being the more similar to the natural ones, are: the threshold function, piecewise linear functions, the sigmoid function and the Gaussian function.

i) *Threshold function* – restricts the neuron output domain to only two values: $\{0,1\}$ in case of a binary threshold function or $\{-1,+1\}$ in case of a bipolar threshold function. Mathematically, the threshold function reads as

$$f(r) = \begin{cases} \alpha, r \geq p \\ \beta, r < p \end{cases} \tag{2}$$

where for the two aforementioned cases $\alpha = 1$, $p = 0$ and $\beta = 0$ or $\beta = -1$, respectively – see also Fig. 2

ii) *Bipolar ramp function* – is a bounded, non-decreasing, piecewise-linear function, with angular points where its derivative has discontinuities – see Fig. 3. The bipolar ramp function

$$f(r) = \begin{cases} 1, & r \geq 1 \\ r, & |r| < 1 \\ -1, & r \leq -1 \end{cases} \tag{3}$$

is a globally Lipschitzian function, with the Lipschitz constant $L = 1$, i.e. it verifies

$$0 < \frac{f(\sigma)}{\sigma} \leq L, f(0) = 0 \tag{4}$$

iii) *Sigmoidal functions* – are bounded, continuous differentiable, monotonically increasing and globally Lipschitzian functions which give a nonlinear graded response within a pre-specified interval, usually $[-1,1]$ or $[0,1]$. The sigmoidal function is wide used for modeling natural processes within different scientific fields such as biology, chemistry, sociology etc. There are several examples of sigmoidal functions – see for instance [Danciu and Răsvan (2001)]. The logistic function (Fig. 4) is a sigmoid function with the Lipschitz constant $L = 1$

$$f(\lambda r) = \frac{1}{1 + e^{-\lambda r}} \tag{5}$$

Remark that as the parameter λ increases, the shape of the sigmoid function approaches the shape of the threshold function.

iv) *Gaussian function* – is a nonlinear radial function used in probabilistic neural networks. Considering the dispersion v^2 the Gaussian function reads as

$$f(r) = e^{-\frac{r^2}{v^2}} \tag{6}$$

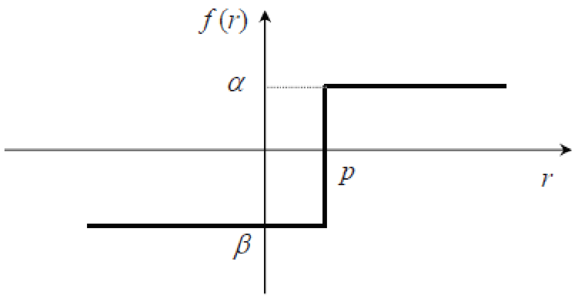

Fig. 2. The threshold function.

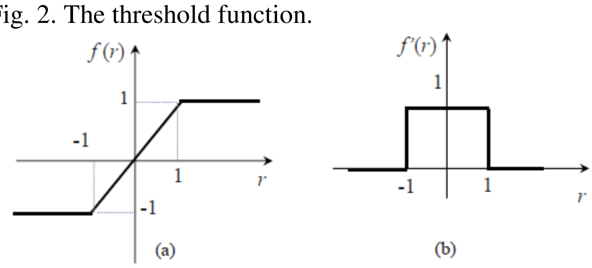

Fig. 3. The bipolar ramp function (a) and its derivative (b).

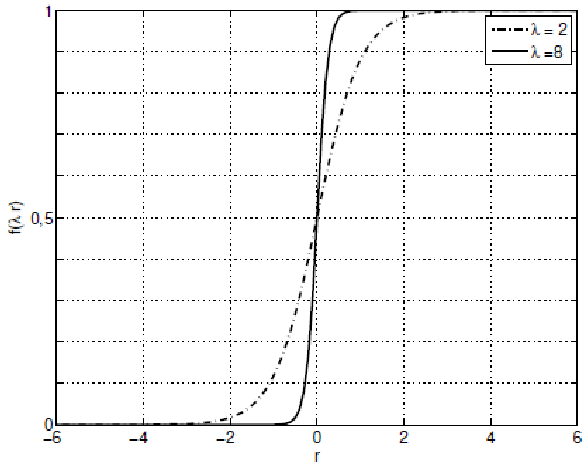

Fig. 4. The shape of a sigmoid function.

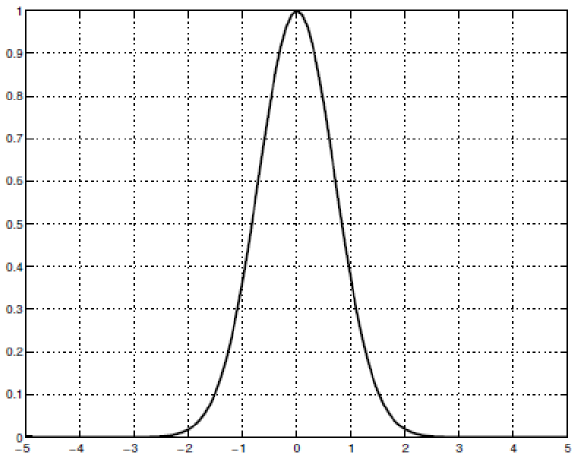

Fig. 5. The Gaussian function.

2.2 The Main Topologies of Artificial Neural Networks

The NNs topology refers to the neurons number and distribution on layers as well as to the layers number and distribution within a network, but also to the interconnections types and the way the information flows through the network. The elementary units of an ANN are organized in one or several layers. The neurons within a layer are similar regarding two aspects: the input signals have the same source and all the neurons have the same updating dynamics. Within a network the interconnections can be: (a) intra-layer, when the neurons of the same layer are interconnected, and (b) inter-layer, when the neurons of different layers are interconnected.

From the point of view of the information "flow" direction through a network, two are the basic topologies for the wide class of ANN: the feedforward and the feedback topologies.

- *The feedforward topology* of a NN means that the information passes the network only in one direction – from its inputs to its outputs. The *Feedforward Neural Networks* (FNNs) have one input layer, one or several hidden layers and one output layer. Being structures with no cyclic interconnections, the FNNs lack dynamics. In Fig. 6 is an example of a FNN having a 3-neurons input layer I_x, one 2-neurons hidden layer H_y and a 3-neurons output layer O_z.

- *The feedback topology* refers to the existence of cyclic interconnections, i.e. connections from the forward layers neurons to the backward layers neurons as well as self-feedback or intra-layer connections as it is shown in Fig. 7. *Recurrent Neural Networks* are neural networks with feedback topology, thus these networks are dynamical systems with complex spatial and temporal behaviors.

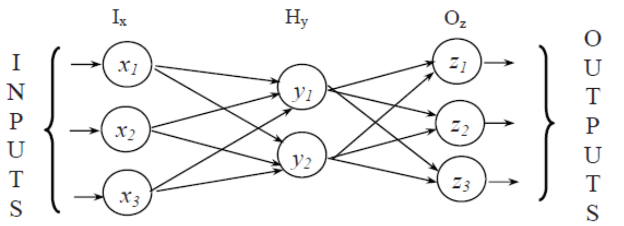

Fig. 6. An example of a Feedforward Neural Network with one hidden layer.

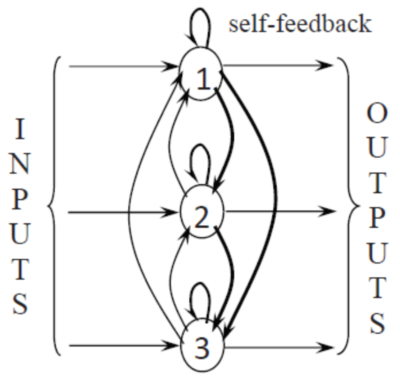

Fig. 7. An example of a full-interconnected Recurrent Neural Network with one layer.

In the sequel we shall present the most known and used RNNs and we shall discuss their specific topologies and the induced qualitative properties.

3. RECURRENT NEURAL NETWORKS

We restrict ourselves to five architectures for RNNs, those who have had a high impact to the technical world. The idea of including recurrence in the neural networks architecture was suggested in the ’70s papers [Grossberg (1988)] of the XX century, but the physicist J. J. Hopfield [Hopfield (1982, 1984)] has been the researcher which pointed out the role of the feedback connections for obtaining some complex desirable behaviors for NNs: “All our interesting results arise as consequences of the strong back-coupling.” Also, J. J. Hopfield “identified network structures and algorithms that could be generalized and that had a high degree of robustness. Significantly, he pointed out throughout his papers that his ideas could be implemented in integrated circuitry, [and] presented his networks in a manner that was easy for engineers and computer scientists to understand”. He defined the energy E of a network and showed “that the algorithm of changing Vi (activation values – our remark) [...] makes E decrease and that eventually a minimum E is obtained. In other words, he proved that the network has stable states.” [Eberhart and Dobbins (1990)].

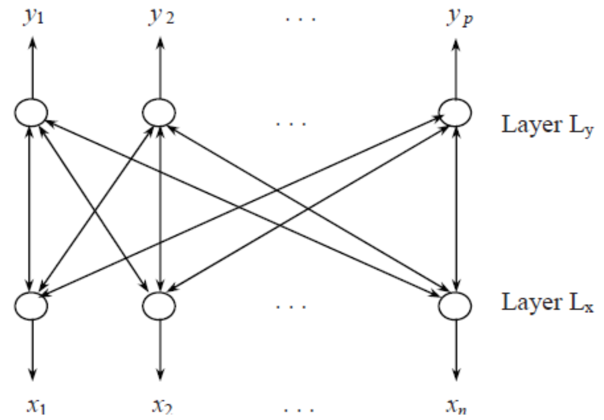

Fig. 8. A Bidirectional Associative Memory. (Source: Kosko (1992))

3.1 Hopfield Neural Networks (HNN)

Introduced by Hopfield (1982, 1984),HNNs are auto-associative neural networks with one layer of full interconnected neurons (Fig. 7) where the information passes between neurons back and forth until an overall equilibrium state is attained, this state being also the output of the network. The interconnection matrix $W = [w_{ij}]$ is symmetric; the original version of the HNN has zero entries on the principal diagonal, i.e. all the neurons have no self-feedback $w_{ij} = 0, i = \overline{1,m}$ – this choice being suggested by some biological facts regarding the low level activity of neurons.

The network dynamics of the continuous-time HNN is described by the system of ordinary differential equations

$$\begin{aligned} \dot{x}_i &= -a_i x_i - \sum_{j=1}^{m} w_{ij} y_i + I_i, i = \overline{1,m} \\ y_i &= f_i(x_i) \end{aligned} \tag{7}$$

where x_i is the state of the neuron, y_i is its output, w_{ij} , $j = \overline{1.m}$ are the synaptic weights from the neurons j to the neuron i, a_i models the passive decay rate of the neuron state to the resting state and the activation function $f(\cdot)$ is a nonlinear sigmoidal function.

The feedback connections and the nonlinear activation function of the neuron lead to some emergent computational capabilities for the HNN, which made it an *associative memory*, or more specific a *content addressable memory* [Hopfield (1982)]. This means that HNN can learn some patterns – given in the form of some binary vectors. The learning procedure is an off-line computation of the synaptic weights, values which remain fixed once the “learning process” is finished. Thus, the "memory" patterns become the stable equilibria of the nonlinear dynamical system. As a result, if a distorted or partial version of a "memory" pattern is presented at the network input, then the evolution of the network state will be such that its output will eventually have the same form as that "memory" pattern. From the point of view of dynamical systems this means that, given an initial condition within the attraction basin of a stable equilibrium having as coordinates the stored pattern, the state trajectory will eventually converge to that equilibrium point [Hopfield (1984)].

As limitations of the HNN wemention: (a) the number of stored patterns is limited by the network capacity to about 15% of the total number of neurons and (b) in order to avoid spurious equilibria, the Hamming distance between the "memory" patterns (equilibria) has to be about 50% [Hopfield (1982)].

3.2 Bidirectional Associative Memory (BAM)

BAM has been proposed by Kosko (1988) as an extension of HNN to the case of two layers NNs with full inter-layers and no intra-layer connections – see Fig. 8. As it is described in [Kosko (1988, 1992)], due to the bidirectional connections, if one denotes W the interconnection matrix from the layer L_x to the layer L_y, then the reverse connections are described by W^T and thus, the overall interconnection matrix is symmetric with the form

$$W^* = \begin{bmatrix} 0 & W \\ W^T & 0 \end{bmatrix} \tag{8}$$

The mathematical model of a discrete-time BAM reads as [Kosko (1992)]

$$\begin{cases} x_{k+1}^i = \sum_{j=1}^{p} w_{ij} R_j\left(y_k^j\right) + I_i, i = \overline{1,n} \\ y_{k+1}^j = \sum_{i=1}^{n} w_{ij} S_i\left(x_k^i\right) + J_j, j = \overline{1,p} \end{cases} \tag{9}$$

where the neuron activation functions are binary threshold functions of the form

$$S_i\left(x_k^i\right) = \begin{cases} 1, & x_k^i > U_i \\ S_i\left(x_{k-1}^i\right), & x_k^i = U_i \\ 0, & x_k^i < U_i \end{cases}$$
$$R_j\left(y_k^j\right) = \begin{cases} 1, & y_k^j > V_j \\ R_j\left(y_{k-1}^j\right), & y_k^j = V_j \\ 0, & y_k^j < V_j \end{cases} \tag{10}$$

with U_i, V_j and I_i, J_j arbitrary-valued thresholds and inputs, respectively. As a heteroassociative neural network, BAM learns the associations between pairs of patterns $\left\{\left(\overline{\boldsymbol{x}}_i, \overline{\boldsymbol{y}}_j\right)\right\}_{i=\overline{1,n}}$, $\overline{\boldsymbol{x}}_i \in \{-1,1\}^n \in \mathbb{R}^{n\times 1}$, $\overline{\boldsymbol{y}}_j \in \{-1,1\}^p \in \mathbb{R}^{p\times 1}$, i.e. the learned pairs of patterns become the "memory" patterns of the BAM. One says that a pair of stored patterns $(\overline{\boldsymbol{x}}, \overline{\boldsymbol{y}})$ is bidirectional stable, i.e. it is a bidirectional stable equilibrium, if the state trajectories starting from any input pairs of patterns within the attraction basins of the two vectors $\overline{\boldsymbol{x}}, \overline{\boldsymbol{y}}$ will eventually attain that "memory" pair pattern. When a bidirectional equilibrium$(\overline{\boldsymbol{x}}, \overline{\boldsymbol{y}})$ is attained, the same signal will pass unchanged back and forth between the two layers and the output signals vectors of the network will be the same as the "memory" vectors [Kosko (1992)].

3.3 Cohen-Grossberg Competitive Neural Networks (CGNN)

The compact form of the Cohen-Grossberg neural network mathematical model is [Cohen and Grossberg (1983)]

$$\dot{x}_i = a_i(x_i)\left[b_i(x_i) - \sum_{j=1}^{n} c_{ij} d_j\left(x_j\right)\right], i = \overline{1,n}, c_{ij} = c_{ji} \tag{11}$$

Grossberg (1988) has showed that taking into consideration different forms for the functions a_i, b_i and d_i one may obtain additive dynamics ("Additive Model networks" such as HNN, BAM) as well as multiplicative dynamics ("Shunting Model networks") described by the general equations

$$\begin{aligned} \dot{x}_i = &-A_i x_i + (B - C x_i)\left[I_i + \sum_{k=1}^{n} D_{ki} f_k(x_k) y_{ki} z_{ki}\right] \\ &- (E + F x_i)\left[J_i + \sum_{k=1}^{n} G_{ki} g_k(x_k) y_{ki} z_{ki}\right], i = \overline{1,n} \end{aligned} \tag{12}$$

where the first term describes the passive decay of the neuron activity, the second term refers to the excitatory signals (external stimulus I_i and the total excitatory feedback from network neurons) while the third term considers the inhibitory signals in the same manner. Also, Grossberg (1988) has showed that by considering appropriate forms of (12) one can obtain the population biology models Volterra-Lotka and Gilpin-Ayala, the Eigen-Schustermodel formacromolecular evolution, the steady state Hartline-Ratliff model of Limulus retina, the Hodgkin-Huxley model of the neuron membrane.

3.4 K-Winner-Takes-All Neural Networks (KWTA)

From the point of view of its architecture, KWTA neural network is a special type of Hopfield network where the neurons activation functions are nonlinear sigmoidal functions with high gains, λ. This feature ensures its selectivity and order preserving properties in the sense that given N constant input signals $d_1, \ldots, d_N$, a KWTA network will be able to provide a N-dimensional output having its first $1 \leq K \leq N-1$ elements with positive values and ordered according to the K-highest input signals. To be more specific, if we consider the input signals $d_1, \ldots, d_N$ in a σ-ordered sequence $d_{\sigma(1)} > d_{\sigma(2)} > \cdots > d_{\sigma(N)}$, then the output signal vector $\overline{y}$ will be such that $\overline{y}_{\sigma(i)} > 0, i = \overline{1,K}, \overline{y}_{\sigma(i)} < 0, i = \overline{K+1,N}$, i.e. the binary sequence $\{\underbrace{1,\ldots,1}_{K},\underbrace{-1,\ldots,-1}_{N-K}\}$ will select the corresponding equilibrium corner of the closed hypercube $\overline{H} = \{-1,1\}$ – the state space for the KWTA networks dynamics. In this way the steady state output variables $\bar{y}_i$ are well delimited from 0 and the network ensures some robustness with respect to the parameter uncertainty [Calvert and Marinov (2000), Danciu and Răsvan (2009)].

KWTA networks are useful in various applications requiring high processing rates, and can be implemented as subsystems in complex systems for decision making applications, list sorting, digital processing [Brockett (1991)].

3.5 Cellular Neural Networks (CNNs)

Introduced by L.O. Chua and his collaborators [Chua and Yang (1988b), Chua and Yang (1988a)] as "a novel class of information-processing systems" of type "large-scale nonlinear analog circuits that process signals in real time", CNNs are ANN of identical cells regularly distributed in *n*-dimensional layers. The "cell" is the elementary processing unit of a CNN and can be described as a nonlinear dynamical system having several kinds of inputs, one output and only local interconnections within a r–radius neighborhood. An important characteristic of CNNs is that these local interconnections, being identical for at least all the inner cells of the network, can be casted into some so-called "cloning templates". Consequently, the cell-based neural networks are desirable for those applications which require parallel processing by means of a huge number of elementary units [Chua and Roska (1993)].

Considering a CNN structured as a two-dimensional array of cells arranged on a regular rectangular grid indexed by $i = \overline{1,N}, j = \overline{1,M}$, the canonical form for the cell dynamics is [Gilli et al. (2002)]

$$\begin{aligned}\dot{x}_{ij}(t) = &\sum_{(k,l)\in N_r(i,j)} T^A_{ij,kl}\big(x_{ij}, x_{kl}\big) f(x_{kl}) \\ &+ \sum_{(k,l)\in N_r(i,j)} T^B_{ij,kl}\big(u_{ij}, u_{kl}\big) u_{kl} \\ &+ \sum_{(k,l)\in N_r(i,j)} T^C_{ij,kl}\big(x_{ij}, x_{kl}\big)\, x_{kl} + I_{ij}\end{aligned} \tag{13}$$

where x_{ij} is the ij^{th} cell state variable, u_{kl} are the control variables from the neighboring cells, $N_r(i\ j)$ is the r–neighborhood of the local interactions of the ij^{th} cell, $T^A_{ij,kl}\big(x_{ij}, x_{kl}\big)$ and $T^C_{ij,kl}\big(x_{ij}, x_{kl}\big)$ are the output and the state feedback cloning templates "that in general might be space-variant nonlinear functions of the state variables x_{ij} and x_{kl}", $T^B_{ij,kl}\big(u_{ij}, u_{kl}\big)$ is the control template, I_{ij} is a bias or an external stimulus and the $C^\infty(\mathbb{R})$ $f(\cdot)$ is a bipolar ramp function of type (3).

The three important features of CNNs – local interconnections, cloning templates and different types of inputs – lead to a certain flexibility in implementation: they can be software emulated or software/hardware implemented on a digital basis, but also they can be hardware implemented in the VLSI technology.

The CNNs main applications include image processing (feature extraction, noise removal, edge detecting etc.) as well as formalized problems solving – the type of problems which can be casted in the form of some high-dimensional systems of ordinary differential equations (ODEs) displaying a certain grade of regularity and similarity thus allowing the identification of some "cloning templates".

4. QUALITATIVE PROPERTIES OF RNN – AN OVERVIEW

A key feature of both natural and artificial neural networks is that their proper work is conditioned by two types of dynamics: "learning dynamics" and "intrinsic dynamics". The "learning dynamics" refers to the gradual adjustment of the synaptic weights during the training process in order the NN to learn performing a specific task – in other words, this process ensures the transfer of the new information in the NN's Long Term Memory (LTM) [Grossberg (1988)]. The "intrinsic dynamics" is induced by the learning process and refers to the neural network regarded as a nonlinear dynamical system.

Usually, the two dynamics are not considered together in the designing stage when a special attention is devoted to only achieving some "useful goals" by the network via learning. It is not compulsory that the resulted "intrinsic dynamics" should have the desirable properties in order to ensure a "good behavior" of the network in achieving the designing goals. Hence, these properties have to be checked *a posteriori* on the mathematical model of the synthesized NN viewed as a dynamical system [Danciu and Răsvan (2007)].

In respect to the "intrinsic dynamics", it is worth mentioning that the *emergent computational capabilities* of a RNN can be achieved provided that it has multiple equilibria. Thus, when the network attains one of its stable equilibria this means that it has solved the task associated with that equilibria – such as selection, detection, decision, classification etc. In the case of dynamical systems with multiple equilibria, the usual local concepts of stability (Lyapunov, asymptotic and exponential stability) are not sufficient for an adequate description. Accordingly, the analysis have to be done within the framework of *Stability Theory* as well as within the framework of the *Qualitative Theory of Systems with Several Equilibria*.

The second framework allows evaluation of the systems global behavior, the specific properties being: dichotomy, global asymptotics and the gradient behavior – for definitions and the main results, the reader is sent to [Reitmann et al. (1992)]. Here we shall give a summary description of these concepts [Răsvan (1998), Danciu (2011)]: (i) *dichotomy* – all bounded solutions tend to the equilibrium set; (ii) *global asymptotics* – all solutions tend to the equilibrium set; (iii) *gradient-like behavior* – the set of equilibria is stable in the sense of Lyapunov and any solution tends asymptotically to some equilibrium point. Let us mention that from the aforementioned properties, the gradient behavior is the most desirable for a proper work of the network in achieving the "designing goals".

As RNNs model the natural brain, there are also other behaviors of interest which can be analyzed within the two frameworks. We mention here two equally important such behaviors: (i) the oscillatory behavior (self-oscillations or forced-oscillations) which involves such dynamics as rhythmicity as well as synchronization of the oscillatory responses with the time-varying external stimuli and (ii) the dynamics affected by time-lags in signal transmission within the network.

5. SOME CONCLUSIONS

In this first lecture the main features and architectures of the Recurrent Neural Networks are introduced. It was shown that due to the feedback interconnections, RNNs are dynamical networks with complex evolutions within their state spaces and that these evolutions depend on the permanent regimes they acquire through the "learning process".

Thus, this first lecture laid the background to further discuss within the second lecture the conditions which ensure those qualitative properties which allow RNNs to proper fulfill the tasks they are designed for. Further, having these "good properties" ensured, one can discusses the main applications of RNNs – the types already introduced in Section 3 of this first lecture. As usual, some concluding remarks will end the second lecture and, thus, will end the survey on RNNs.

REFERENCES

Brockett, R. (1991). Dynamical systems that sort lists, diagonalize matrices and solve linear programming problems. *Lin. Algeb. Applicat.*, 146, 79–91.

Calvert, B. and Marinov, C. (2000). Another k-winners-takeall analog neural network. *IEEE Trans. Neural Networks,* 11, 829–838.

Chua, L. and Roska, T. (1993). The cnn paradigm. *IEEE Trans. Circuits Syst*. I, 40(3), 147–156.

Chua, L. and Yang, L. (1988a). Cellular neural networks: applications. *IEEE Trans. Circuits Systems*, 35(10), 1273– 1290.

Chua, L. and Yang, L. (1988b). Cellular neural networks: theory. *IEEE Trans. Circuits Systems*, 35(10), 1257–1272.

Cohen, M.A. and Grossberg, S. (1983). Absolute stability of pattern formation and parallel storage by competitive neural networks. *IEEE Trans. Systems, Man and Cybernetics*, 13, 815–825.

Danciu, D. (2011). Bio-inspired systems. several equilibria. qualitative behavior. In J.C. et. al. (ed.), *Proc. 10th International Work-Conference on Artificial Neural Networks IWANN*, volume 6692 of LNCS, 573–580. Springer-Verlag Berlin Heidelberg.

Danciu, D. and Răsvan, V. (2001). Steady state "almost linear" behavior of delayed hopfield type neural networks. *In Proc. 13th International Conference on Control Systems and Computer Science CSCS*, 210–213. Romania.

Danciu, D. and Răsvan, V. (2007). Dynamics of neural networks - some qualitative properties. In Proc. *9th Int. Work- Conference on Artificial Neural Networks IWANN*, 8–15. Spain.

Danciu, D. and Răsvan, V. (2009). Gradient like behavior and high gain design of kwta neural networks. In J.C. et al. (ed.), *Proc. 10th InternationalWork-Conference on Artificial Neural Networks IWANN*, volume 5517, 24–32. Springer.

Eberhart, R. and Dobbins, R. (1990). *Neural Network PC Tools. A Practical Guide*. Elsevier Inc, Academic Press.

Galushkin, A. (2010). *Neural Network Theory*. Springer.

Gilli, M., Roska, T., Chua, L., and Civalleri, P. (2002). Cnn dynamics represents a broader class than pdes. *I. J. Bifurcation and Chaos*, (10), 2051–2068.

Grossberg, S. (1988). Nonlinear neural networks: Principles, mechanisms, and architectures. *Neural Networks*, 1, 17–61.

Hopfield, J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proc. Nat. Acad. Sci.*, 79, 2554–2558.

Hopfield, J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. *Proc. Nat. Acad. Sci.,* 81, 3088–3092.

James, W. (1890). *The Principles of Psychology*. Henry Holt and Company, New York.

Kosko, B. (1988). Bidirectional associative memories. *IEEE Trans. Systems, Man and Cybernetics*, 18(1).

Kosko, B. (1992). *Neural Networks and Fuzzy Systems*. Prentice-Hall International, London.

McCulloch,W.S. and Pitts,W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5, 115–133.

Reitmann, V., Smirnova, V., and Leonov, G. (1992). *Non-Local Methods for Pendulum-Like Feedback Systems*. Vieweg- Teubner Verlag.

Răsvan, V. (1998). Dynamical systems with several equilibria and natural liapunov functions. *Archivum mathematicum*, 34(1), 207–215.

Tsoi, A.C. and Back, A.D. (1994). Locally recurrent globally feedforward networks: A critical review of architectures. *IEEE Transactions on Neural Networks*, 5(2).