Path Following with Collision Avoidance and Velocity Constraints for Multi-Agent Systems

Ionela Prodan * **Sorin Olaru** * **Cristina Stoica** *
Silviu-Iulian Niculescu **

* *SUPELEC Systems Sciences (E3S) - Automatic Control Department, 3 rue Joliot Curie, F-91190, Gif-sur-Yvette cedex, France (e-mail: {Ionela.Prodan, Sorin.Olaru, Cristina.Stoica}@supelec.fr).*
** *Laboratory of Signals and Systems, SUPELEC - CNRS, 3 rue Joliot Curie, Gif-sur-Yvette, France (e-mail: {Ionela.Prodan, Silviu.Niculescu}@lss.supelec.fr).*

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.

Keywords: Multi-agent systems, cooperative control, Model Predictive Control, non-convex constrains

1. INTRODUCTION

Questions about achieving a formation of a group of agents and how to ensure that all the agents avoid collision both among the group and with the obstacles around them arise when dealing with multi-agent systems Richards and How (2005). The goal of this paper is to control 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 (Mesbahi and Egerstedt (2010), Blondel et al. (2005), Olfati-Saber and Murray (2002)). Among the applications, can be cited the characterization of pedestrian behavior in the crowd (Helbing et al. (2000), Fang et al. (2010)). 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 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.

This paper considers 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 (Camacho and Bordons (2004), Rossiter (2003), Mayne et al. (2000), Maciejowski (2002), Bemporad and Morari (1999)). This paper deals with 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. The design of vehicle formation through the use of MPC is detailed in Dunbar and Murray (2002). Other approaches can be found in aerospace applications, where MPC is applied to spacecraft formation keeping Manikonda et al. (1999), 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 particular, in Bemporad and Morari (1999), the authors considered such an approach based on the use of auxiliary binary variables together with MPC, with examples in hybrid control systems. In Richards and How (2005), the collision and obstacle avoidance are included in the trajectory planning of spacecraft vehicles, but the velocity constraints are not taken into account.

The present paper considers a two-dimensional environment for the group of agents, with supplementary non-convex velocity constraints that can also be handled by adding binary variables.

An important contribution to 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 receding horizon mixed integer optimization problem. Using the predicted control laws, 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 (Van Nieuwstadt and Murray (1998), De Doná et al. (2009)). In De Doná et al. (2009), the use of MPC is combined with the 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.

This remaining paper is organized as follows. Section 2 introduces the agent dynamics and the reference trajectory generation mechanism. The constrained predictive control problem is then summarized in. Section 3 deals with the linear reformulation of the state constraints for the real-time optimization problem. Section 4 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 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 5, illustrative simulation results are presented. Finally, some conclusions are drawn in Section 6.

Throughout the paper, the following notations are used. An intersection of finitely many half-spaces, 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
$Q \succ 0$ - Q is a strictly positive definite matrix
δ^c - binary variables $\{0, 1\}$

2. 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 possible 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:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{v_x} \\ \dot{v_y} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\dfrac{\mu}{m} & 0 \\ 0 & 0 & 0 & -\dfrac{\mu}{m} \end{bmatrix} \begin{bmatrix} x \\ y \\ v_x \\ v_y \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \dfrac{1}{m} & 0 \\ 0 & \dfrac{1}{m} \end{bmatrix} \begin{bmatrix} u_x \\ u_y \end{bmatrix}, \qquad (1)$$

where (x, y) are the position coordinates and (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:

$$\dot{\xi}^i(t) = A_c \xi^i(t) + B_c u^i(t), \ i = 1 : N_a, \qquad (2)$$

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_s by considering the time instants $t_k = kT_s$:

$$\xi^i_{k+1} = A\xi^i_k + Bu^i_k, \quad k \in \mathbb{N}, \quad i = 1 : N_a, \qquad (3)$$

where $\xi_k = \xi_{t_k}$, $u_k = u_{t_k}$. The pair (A, B) is given by:

$$A = e^{A_c T_s}, \quad B = \int_0^{T_s} e^{A_c(T_s - \theta)} B_c d\theta$$

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:

$$P(\xi^i) = \{\xi \in \mathbb{R}^{n_\xi} : H(\xi - \xi^i) \leq K\} \qquad (4)$$

where $H \in \mathbb{R}^{n_c \times n_\xi}$, $K \in \mathbb{R}^{n_c \times 1}$, and n_c is the number of hyperplanes. 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 completeness, the problem of generating a reference trajectory for the linear system (1) is next summarized, along the line in Van Nieuwstadt and Murray (1998).

B. Trajectory generation

The idea is to find a trajectory $(\xi(t), u(t))$ that steers the model (1) from an initial state x_0 to a final state x_f, over a fixed time interval $[t_0, t_f]$. Using the flatness theory introduced by Van Nieuwstadt and Murray (1998) the system is parameterized in terms of a finite set of variables $z(t)$ and a finite number of their derivatives:

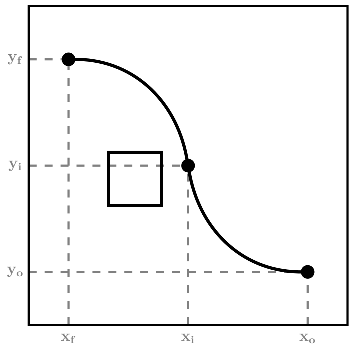

Fig. 1. A flat trajectory for obstacle avoidance

$$\begin{aligned} \xi(t) &= \xi(z(t), \dot{z}(t), \cdots, z^{(q)}(t)), \\ u(t) &= u(z(t), \dot{z}(t), \cdots, z^{(q)}(t)), \end{aligned} \tag{5}$$

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 the class of polynomial functions is used. Based on the parametrization (5) and imposing boundary constraints for the evolution of the differentially flat systems De Doná et al. (2009) 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):

$$\begin{aligned} \xi^{ref}(t) &= \xi(z^{ref}(t), \dot{z}^{ref}(t)), \\ u^{ref}(t) &= u(\dot{z}^{ref}(t), \ddot{z}^{ref}(t)), \end{aligned} \tag{6}$$

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 redesigning of the flat trajectory.

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 $u^* = \left\{u_{k|k}, u_{k+1|k}, \cdots, u_{k+N-1|k}\right\}$ that minimizes the finite horizon quadratic objective function $V_N(\xi_k, u^*)$:

$$\begin{aligned} V_N = {} & \xi^T_{k+N|k} P \xi_{k+N|k} + \sum_{l=1}^{N-1} \xi^T_{k+l|k} Q \xi_{k+l|k} + \\ & + \sum_{l=0}^{N-1} u^T_{k+l|k} R u_{k+l|k}, \end{aligned} \tag{7}$$

while respecting the constraints imposed by each agent dynamics (3), and the physical limitations

$$\xi^i \in \overline{P}(\xi^j), \ (i,j) \in \mathbb{N}_{[1,N_a]} \times \mathbb{N}_{[1,N_a]}, i \neq j. \tag{8}$$

Here $Q = Q^T \succeq 0$, $R \succ 0$ are the weighting matrices and $P = P^T \succeq 0$ 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 Mayne et al. (2000).

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 the constraints (8) are reformulated 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 Bemporad and Morari (1999).

3. 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 a non-convex region. In order to reformulate the non-convex constraints in a convex 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 half-spaces defining the constraints in (10) has 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 a rectangular shape for the region is considered 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:

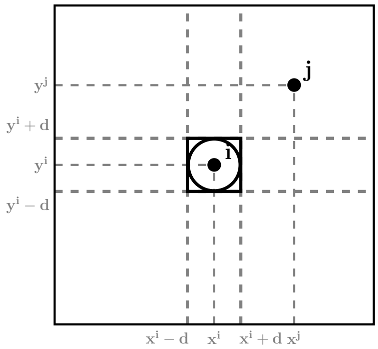

Fig. 2. Approximation of state constraints: the square approximates the regions with linear constraints

$$
\begin{aligned}
&x^j \geq x^i + d \,\text{or}\;\; x^j \leq x^i - d \quad \text{or} \\
&y^j \geq y^i + d \,\text{or}\;\; y^j \leq y^i - d.
\end{aligned} \tag{12}
$$

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:

$$
\begin{aligned}
&x^i - x^j \leq -d + M\delta^1, -x^i + x^j \leq -d + M\delta^2, \\
&y^i - y^j \leq -d + M\delta^3, -y^i + y^j \leq -d + M\delta^4, \\
&\delta^1 + \delta^2 + \delta^3 + \delta^4 \leq 3.
\end{aligned} \tag{13}
$$

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.

A more compact representation 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:

$$
\underbrace{\left[\begin{array}{cccc|cccc} 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \end{array}\right]}_{C^p_{ij}} \left[\begin{array}{c} \xi^i_k \\ \hline \xi^j_k \end{array}\right] +
$$

$$
+ \underbrace{\begin{bmatrix} M & M \\ -M & M \\ M & -M \\ -M & -M \end{bmatrix}}_{D^p_{ij}} \begin{bmatrix} \delta^1 \\ \delta^2 \end{bmatrix} \leq \underbrace{\begin{bmatrix} -d \\ -d + M \\ -d + M \\ -d + 2M \end{bmatrix}}_{\gamma^p_{ij}} \tag{15}
$$

These constraints have to be used in the classical MPC framework with the values of $(C^p_{ij}, D^p_{ij}, \gamma^p_{ij})$ 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:

$$
\begin{aligned}
&v^i_x \leq -v_m, \text{or} -v^i_x \leq -v_m, \quad \text{or} \\
&v^i_y \leq -v_m, \text{or} -v^i_y \leq -v_m,
\end{aligned} \tag{16}
$$

where the constant $v_m > 0$.

Similarly, the non-convex velocity constraints (16) can be rewritten using binary variables δ^c, $c = 1, 2$, which correspond in general terms to *fixed-obstacle avoidance constraints*:

$$
\begin{aligned}
v^i_x &\leq -v_m + M(\delta^1 + \delta^2), \\
-v^i_x &\leq -v_m + M(1 - \delta^1 + \delta^2), \\
v^i_y &\leq -v_m + M(1 + \delta^1 - \delta^2), \\
-v^i_y &\leq -v_m + M(2 - \delta^1 - \delta^2),
\end{aligned} \tag{17}
$$

For each agent velocity constraints can be added in the classical MPC framework with the values of $(C^v_{ij}, D^v_{ij}, \gamma^v_{ij})$, defined similar with(15).

Globally, the non-convex state constraints (12), (16) are transposed in a compact form:

$$
\begin{bmatrix} C^p_{ij} \\ C^v_{ij} \end{bmatrix} \xi^{ij} + \begin{bmatrix} D^p_{ij} & 0 \\ 0 & D^v_{ij} \end{bmatrix} \begin{bmatrix} \delta^p \\ \delta^v \end{bmatrix} \leq \begin{bmatrix} \gamma^p_{ij} \\ \gamma^v_{ij} \end{bmatrix}, \tag{18}
$$

with δ^p and δ^v the auxiliary binary variables introduced for position and velocity constraints reformulation and $\xi^{ij} = [{\xi^i}^T \; {\xi^j}^T]^T$, the global state of any two different agents $(i, j) \in \mathbb{N}_{[1,N_a]} \times \mathbb{N}_{[1,N_a]}, i \neq j$.

4. OPTIMIZATION-BASED CONTROL & PATH FOLLOWING

This section presents the centralized MPC problem, where an optimization is performed to compute the control laws for each agent. Based on the global 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:
$\tilde{\xi} = [x^1 \; y^1 \; v^1_x \; v^1_y | \cdots | x^{N_a} \; y^{N_a} \; v^{N_a}_x \; v^{N_a}_y]^T$,
$\tilde{u} = [u^1_x \; u^1_y | \cdots | u^{N_a}_x \; u^{N_a}_y]^T$,
and the matrices which describe the global 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) and (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 2. 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

$$\begin{aligned} &\hat{\xi}_{k+l+1|k} = A_g \hat{\xi}_{k+l|k} + B_g \hat{u}_{k+l|k}, \quad l = 0 : N-1 \\ &\begin{bmatrix} C^p \\ C^v \end{bmatrix} \hat{\xi}_{k+l|k} + \begin{bmatrix} D^p & 0 \\ 0 & D^v \end{bmatrix} \begin{bmatrix} \delta^p_{k+l|k} \\ \delta^v_{k+i|k} \end{bmatrix} \le \begin{bmatrix} \gamma^p \\ \gamma^v \end{bmatrix} \end{aligned} \tag{24}$$

where $A_g, B_g, C^p, C^v, D^p, D^v, \gamma^p, \gamma^v$ 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 given reference trajectory.

5. 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 window 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 $d = 1$, $v_{min} = 8$, $M > 0$. Fig.5 presents the simulation results for the initial states are: $\xi^1 = [15 \ -1 \ 1 \ -5]^T$, $\xi^2 = [-7 \ 7 \ 5 \ 10]$, $\xi^3 = [8 \ 8 \ -5 \ 2]$. The agents are self-organizing in a triangle formation and the trajectory of its center of mass is plotted in Fig.6 in magenta. Fig.5.c illustrates good tracking performances of the center of mass (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

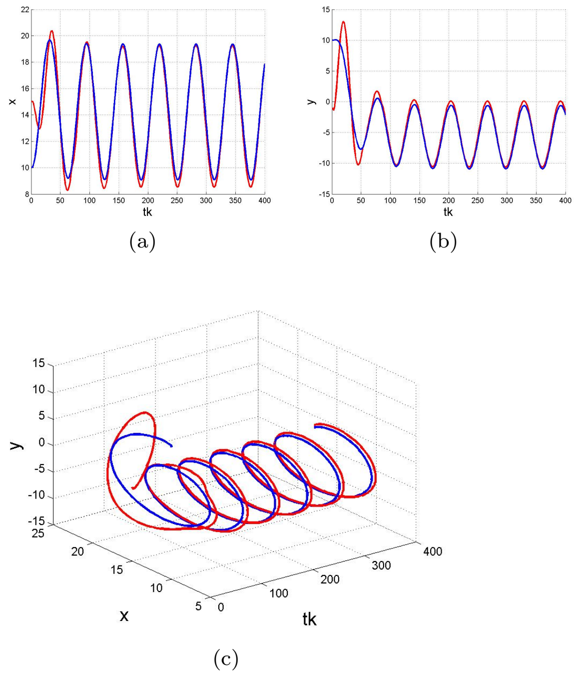

Fig. 3. The reference trajectory and the time evolution of one agent, along the path: (a) X-axis, (b) Y-axis, (c) X,Y-axes

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 formation proves to be sensitive to the alignment and this indicates from simulations that the triangle formation is more stable. This deserves a detailed analysis when disturbances and noises affect the agents dynamic and represents one of the current research topics.

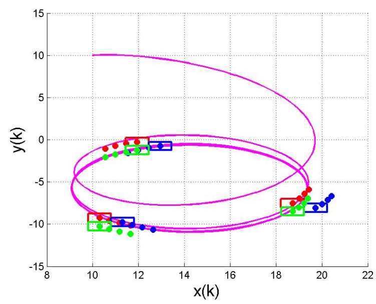

Fig. 4. Behavior of three agents in a triangle formation, with position constraints, at different time instances

6. 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 paper. In the path following of multi-agents systems problems may appear when dealing with collision avoidance constraints, both within the group or with any obstacles. In this context, the collision avoidance constraints describe a non-convex region, therefore a set of auxiliary binary variables are introduced in order to translate the

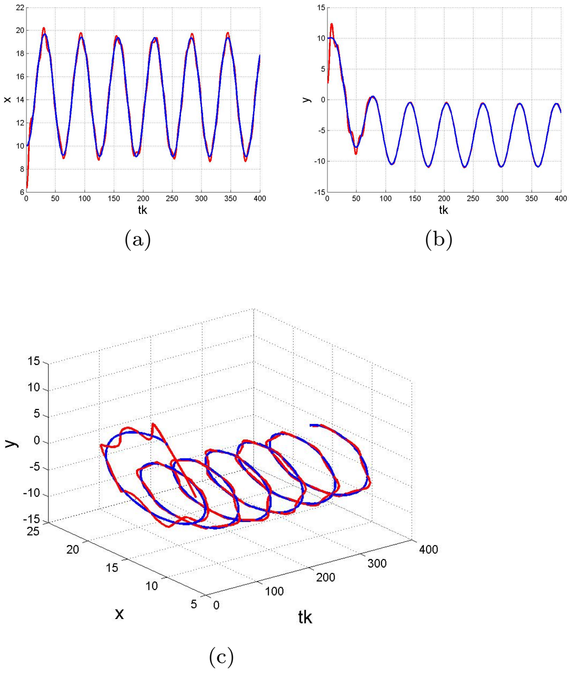

Fig. 5. The reference trajectory and the time evolution of the center of mass of the triangle formation of 3 agents, along the path: (a) X-axis, (b) Y-axis, (c) X,Y-axes

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.

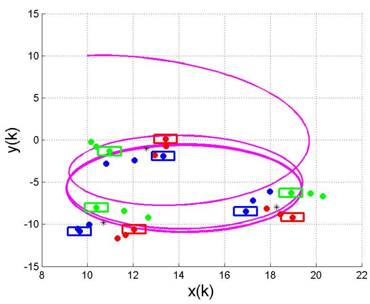

Fig. 6. Behavior of three agents in a triangle formation, with position and velocity constraints, at different time instances.

7. ACKNOWLEDGEMENTS

The research of Ionela Prodan is supported by the EADS Foundation.

REFERENCES

Bemporad A., Morari M. (1999). Control of systems integrating logic, dynamics, and constraints. Automatica **35**, 407–428

Blondel V., Hendrickx J., Olshevsky A., Tsitsiklis J. (2005). Convergence in multiagent coordination, consensus, and flocking. In: 44th IEEE Conference on Decision and Control and European Control Conference, pp. 2996–3000. Seville, Spain

Camacho E.F., Bordons C. (2004). Model predictive control. Springer Verlag

De Doná J., Suryawan F., Seron M., Lévine J. (2009). A Flatness-Based Iterative Method for Reference Trajectory Generation in Constrained NMPC. Lecture notes in control and information sciences **384**, 325–333

Dunbar W., Murray R. (2002). Model predictive control of coordinated multi-vehicle formations. In: Proceedings of the 41th IEEE Conference on Decision and Control, vol. 4, pp. 4631–4636. Las Vegas, Nevada, USA

Fang Z., Song W., Zhang J., Wu H. (2010). Experiment and modeling of exit-selecting behaviors during a building evacuation. Statistical Mechanics and its Applications **389**(4), 815–824

Helbing D., Farkas I., Vicsek T. (2000). Simulating dynamical features of escape panic. Nature **407**(6803), 487–490

Maciejowski J. (2002). Predictive control: with constraints. Pearson education

Manikonda V., Arambel P., Gopinathan M., Mehra R., Hadaegh F., Inc S., Woburn M. (1999). A model predictive control-based approach for spacecraft formation keeping and attitude control. In: Proceedings of the American Control Conference, vol. 6. San Diego, California, USA

Mayne D., Rawlings J., Rao C., Scokaert P.O. (2000). Constrained model predictive control: Stability and optimality. Automatica **36**, 789–814

Mesbahi M., Egerstedt M. (2010). Graph-theoretic Methods in Multi-agent Networks. Princeton University Press

Olfati-Saber R., Murray R. (2002). Distributed cooperative control of multiple vehicle formations using structural potential functions. In: Proceedings of the 15th IFAC World Congress, pp. 346–352. Barcelona, Spain

Richards A., How J. (2005). Model predictive control of vehicle maneuvers with guaranteed completion time and robust feasibility. In: Proceedings of the 24th American Control Conference, vol. 5, pp. 4034–4040. Portland, Oregon, USA

Rossiter J. (2003). Model-based predictive control: a practical approach. CRC

Van Nieuwstadt M., Murray R. (1998). Real-time trajectory generation for differentially flat systems. International Journal of Robust and Nonlinear Control **8**(11), 995–1020