Fault Tolerant Predictive Control for Multi-Agent Dynamical Systems: Formation Reconfiguration using Set-Theoretic Approach

M.T. Nguyen, C. Stoica Maniu, S. Olaru, A. Grancharova

Abstract— This paper deals with the task assignment problem, which is a major issue in dynamical formation control. The formation is defined in terms of a group of homogeneous dynamical agents. The position of each agent in the formation is predetermined by pre-imposing the distance between each pair of agents. Recently by using set-theoretic methods, the task assignment has been formulated in terms of an optimization problem allowing to keep the agents in a tight formation in real-time. In this paper we propose a new algorithm for the task assignment formulation in view of real-time control by including fault detection and isolation capabilities. The proposed methods will be illustrated by means of a numerical example.

Index Terms— Multi-Agent dynamical systems, Fault Detection and Isolation, set-membership theory, tight formation control.

I. INTRODUCTION

A dynamical Multi-Agent system (MAS) is composed of multiple intelligent agents interacting within an environment subject to constraints. Nowadays, MAS receives considerable attention due to the need to control a group of relatively independent sub-systems for the purpose of achieving a common goal.

Beside the performance quality, the mission safety becomes an intensive research field for MAS application. This criterion requires a supplementary fault diagnosis layer to detect and isolate the plant, sub-system or sensor faults. Recently a redefinition of Fault Detection and Isolation (FDI) goals becomes a highly required priority for Multi-Agent system. The FDI scheme is supposed not to affect the stability of the formation control and to preserve the collision avoidance guarantees. In the existing literature, many researches have been conducted on this topic. Precisely in [1], [2], [3] the authors have developed a set of FDI filters to maintain the functioning of MAS under the actuator fault in presence of large environmental disturbance. Others works [4], [5] have used model-based Fault Detection to generate residual signals for MAS.

Recently results have been reported on the application of set-theoretic and optimization tools for MAS control, notable [6]. Furthermore, these tools were also used to design FDI scheme based on the separation between different functioning modes [7]. The faults treated in this brand are

M.T. Nguyen, C. Stoica Maniu and S. Olaru are with SUPELEC Systems Sciences (E3S) – Automatic Control Department, 3 rue Joliot Curie, F-91192, Gif-sur-Yvette cedex, France (e-mail: {minhtri.nguyen; cristina.stoica;sorin.olaru}@supelec.fr).

A. Grancharova is with the Institute of System Engineering and Robotics, Bulgarian Academy of Sciences, Acad. G. Bonchev str., Bl.2, P.O.Box 79, Sofia 1113, Bulgaria (phone: 3592-873-26-14; fax: 3592-870-33-61;e-mail: alexandra.grancharova@abv.bg).

sensor fault [8], [9] and actuator fault [10], [11] for single systems. Our idea is to employ them to design a Fault Tolerant Control (FTC) for Multi-Agent system. We begin by considering the simplest case of fault where the priority is to preserve the formation. Precisely, the FDI scheme based on set-theoretic methods will be able to detect if an agent is faulty and if this fault falls in a serious category, to eliminate it from the team. The proposed FDI technique for Multi-Agent systems (which is the novelty of this paper) is completed by a reconfiguration step to calculate a new optimal configuration for the remaining agents. The role of the control input is to steer and keep the MAS into this new formation, while avoiding the collision between the agents.

The aim of the present paper is two fold. First, it reviews and analyzes the algorithm given in [6], [12] for Multi-Agent system with respect to faults and reconfiguration objectives. Second, set-theoretic tools are used to solve the task assignment in real time, when considering actuator faults for the MAS system.

The remainder of the paper is organized as follows. Section II presents useful mathematical notations and basic definitions. In this section the necessary elements from previous work will be briefly recapitulated. Section III presents two problem statements: the feasibility of solving optimization problems with non-convex constraints, and the real time formation preservation, whereas we will just focus on the latter. A new task assignment algorithm will be proposed in the Section IV to solve this problem. Section V proposes an example to illustrate the performance of the new FDI algorithm for MAS. Finally, some concluding remarks and perspectives are mentioned in Section VI.

II. MATHEMATICAL NOTATIONS, DEFINITIONS AND PROPERTIES

In order to use set-theoretic concepts, we introduce next a series of useful concepts linking the dynamical systems to static geometrical sets in the state-space.

A. Notations

The *Minkovski sum* of two sets $\mathcal{A}$ and $\mathcal{B}$ is defined by $\mathcal{A} \oplus \mathcal{B} = \{a + b : a \in \mathcal{A}, b \in \mathcal{B}\}$.

$|\cdot|$ denotes the element-wise absolute value. $\|\cdot\|_Q$ denotes the value of the Euclidean norm, with $Q = Q^\top \succ 0$ a weighting symmetric positive definite matrix. $I_n \in \mathbb{R}^{n \times n}$ denotes the unitary matrix of dimension n.

$\mathbb{N} \triangleq \{0, 1, 2...\}$ is the set of all natural numbers.

$\mathbb{N}_{[1,N]} \triangleq \{0, 1, 2..., N\}$, with $N \in \mathbb{N}$, contains the index of each agent in the MAS.

We use $\mathcal{N}_E \in \mathbb{N}_{[1,N]}$ to denote the set containing the indices of the eliminated agents due to the fault occurrence. Hence the set of the remaining agents' indices is $\mathcal{N}_R = \mathbb{N}_{[1,N]} \backslash \mathcal{N}_E$.

$U(k + N_p|k) = [u^\top(k|k)\, u^\top(k+1|k) \ldots u^\top(k+N_p|k)]^\top$ denotes the set of control inputs given by the Model Predictive Control (MPC) law.

In the paper two notions of receding horizons will be used:
- The prediction horizon denoted by N_p;
- The fault monitoring horizon denoted by N_m.

Consider a bounded polyhedral set $\mathcal{S} \subset \mathbb{R}^n$, its closure is denoted by $cl(\mathcal{S})$ and $C(\mathcal{S}) = cl(\mathcal{S}) \setminus \mathcal{S}$ is the complement of $\mathcal{S}$. Both $\mathcal{S}$ and $C(\mathcal{S})$ are bounded in order to allow the implementation.

$\check{x}_i(k)$ denotes the *one-step predicted state* of the ith agent.

$\bar{x}_i$ denotes the target position of the ith agent in the case that the common reference of the entire MAS reduces to the origin. Its role will be detailed in the next subsection.

$\breve{x}_i(k)$ denotes the trajectory reference of the ith agent, once one configuration for the entire system is determined.

B. Ultimate bound invariant set

Definition 1: [13] Consider an autonomous linear discrete time-invariant system $x_{k+1} = Ax_k + w_k$, with matrix A assumed to be a Schur matrix. A set $\mathcal{S}$ is called *robustly positive invariant* (RPI) for this system, if $Ax_k + w_k \in \mathcal{S}$ for all $x_k \in \mathcal{S}, w_k \in \mathcal{W}$, which is equivalent to:

$$A\mathcal{S} \oplus \mathcal{W} \subseteq \mathcal{S} \qquad (1)$$

Theorem 1: [6] Consider system $x_{k+1} = Ax_k + w_k$, with matrix A assumed to be a Schur matrix and a non-negative vector w_k such that $|w_k| \leq \bar{w}$, $\forall w_k \in \mathcal{W} \subset \mathbb{R}^n$. Let $A = VJV^{-1}$ be the Jordan decomposition of A. Then the set

$$\Omega_{UB} = \{x \in \mathbb{R}^n : |V^{-1}x| \leq (I - |J|)^{-1}|V^{-1}\bar{w}|\} \quad (2)$$

is robustly invariant (RI) with respect to the system's dynamics.

C. Dynamics equation for the network of agents

Consider a MAS composed of N agents. Each agent is characterized by a discrete-time dynamics equation:

$$x_{d,i}(k+1) = A_i x_{d,i}(k) + B_i u_{d,i}(k) + w_i(k),\ i \in \mathbb{N}_{[1,N]} \quad (3)$$

where $x_{d,i}(k) \in \mathbb{R}^n$ is the ith agent's state and $u_{d,i}(k) \in \mathbb{R}^m$ is the corresponding input vector. $w_i(k) \in \mathcal{W}$ denotes the disturbances, with $\mathcal{W} \subset \mathbb{R}^n$ a bounded set which contains the origin. The pairs (A_i, B_i) are assumed to be stabilizable [6], with $A_i \in \mathbb{R}^{n \times n}$ and $B_i \in \mathbb{R}^{n \times m}$.

The nominal dynamics corresponding to (3) is defined according to the framework proposed in [6]:

$$x_i(k+1) = A_i x_i(k) + B_i u_i(k),\ i \in \mathbb{N}_{[1,N]} \quad (4)$$

where $x_i(k) \in \mathbb{R}^n$ and $u_i(k) \in \mathbb{R}^m$. The control input in (3) is defined as $u_{d,i}(k) = u_i(k) + K_i[x_{d,i}(k) - x_i(k)]$. By denoting $e_i(k) = x_{d,i}(k) - x_i(k)$ as the tracking error of the ith agent we obtain the following equation:

$$e_i(k+1) = (A_i + B_i K_i)e_i(k) + w_i(k),\ i \in \mathbb{N}_{[1,N]} \quad (5)$$

The stabilizability assumption of the pairs (A_i, B_i) ensures the existence of $K_i \in \mathbb{R}^{m \times m}$ which stabilizes the tracking error dynamics (5). Then applying *Theorem 1* for the tracking error equation, we can construct a RPI set $\mathcal{S}_{e_i}$ which satisfies:

$$e_i(k) \in \mathcal{S}_{e_i},\ \forall k \geq k_0 \text{ and } \forall e_i(k_0) \in \mathcal{S}_{e_i} \qquad (6)$$

This means that the real state (3) always resides in a tube around its nominal trajectory. Furthermore, this RPI set $\mathcal{S}_{e_i}$ can be considered as the safety region around each agent in MAS. It is important to mention that the real state $x_{d,i}(k)$ is unknown due to $w_i(k)$, but by the construction of (6), its trajectory is bounded by the tube (see Fig. 1):

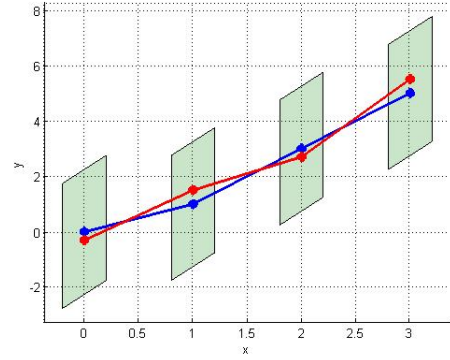$$\mathcal{S}(x_i(k)) \triangleq x_i(k) \oplus \mathcal{S}_{e_i} \qquad (7)$$

Fig. 1: Nominal trajectory (blue). Real trajectory (red). RPI set of tracking error (green).

The global Multi-Agent system can be defined as:

$$x(k+1) = A_g x(k) + B_g u(k) \qquad (8)$$

where $x(k) = [x_1^\top(k)\ x_2^\top(k) \ldots x_N^\top(k)]^\top \in \mathbb{R}^{Nn}$ and $u(k) = [u_1^\top(k)\ u_2^\top(k) \ldots u_N^\top(k)]^\top \in \mathbb{R}^{Nm}$ denote respectively the collective state and input vector of the global system. Similarly $A_g = diag(A_1, A_2, \ldots, A_N) \in \mathbb{R}^{Nn \times Nn}$ and $B_g = diag(B_1, B_2, \ldots, B_N) \in \mathbb{R}^{Nn \times Nm}$ collect all the input matrices corresponding to each agent. For a global homogeneous system we use A, B to denote A_i, B_i, resp.

D. Minimal configuration

This section resumes the approach proposed in [6] to obtain a minimal reconfiguration for a MAS defined as (8). The objective is to determinate the position of all the agents around the common reference providing that they are as close as possible to this reference while avoiding the collision (see Fig. 2). To achieve such an objective, it requires avoiding the intersection of the safety regions of any pair of agents. Moreover, these target positions have to be equilibrium/stationary points relative to the dynamics (4), otherwise the agent's behavior cannot ensure the satisfaction of dynamical constraints once the formation is achieved.

The previous idea is formulated in terms of an optimization problem as follows:

$$\min_{\bar{u}_i} \sum_{i=1}^{N} \|\bar{x}_i\|$$

$$\text{st: } \begin{cases} \bar{x}_i - \bar{x}_j \notin -\mathcal{S}_i \oplus \mathcal{S}_j, \forall i, j \in \mathbb{N}_{[1,N]}, i \neq j \\ \bar{x}_i = A\bar{x}_i + B\bar{u}_i \end{cases} \quad (9)$$

where $\bar{x}_i$ indicates the error between the state of the ith agent and the origin (which represents the common reference). $\mathcal{S}(\bar{x}_i)$ represents the *located safety region* of the ith agent. According to (7), $\mathcal{S}(\bar{x}_i) = \bar{x}_i \oplus \mathcal{S}_i$, in which $\mathcal{S}_i$ represents the *safety region* of this agent.

Due to the non-convexity of these constraints, this problem can be solved by using *Mixed Integer Programming* (MIP). This requires finding $\bar{x}_i - \bar{x}_j$ in the complement set $C(-\mathcal{S}_i \oplus \mathcal{S}_j)$. This set is described by MIP implementation as the feasible region of the optimization problem. Its solution given by MIP is the set of target positions for the group of agent. Once the formation is determined, it will be preserved along the *common reference*[1] x_{ref}, as depicted in Figure 2. Hence the target trajectory of the ith agent is denoted by:
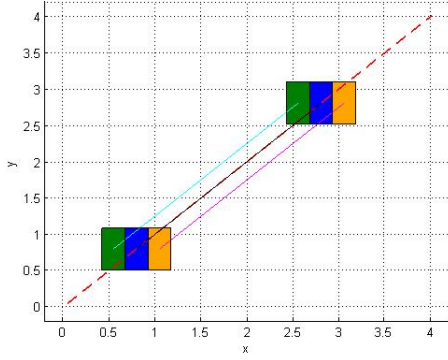

Fig. 2: Trajectories of each agent around the common reference (red dash-line)

$$\begin{aligned} \breve{x}_i(k) &= x_{ref}(k) + \bar{x}_i \\ \breve{u}_i(k) &= u_{ref}(k) + \bar{u}_i \end{aligned} \quad (10)$$

This trajectory is associated with the dynamics equation:

$$\breve{x}_i(k+1) = A\breve{x}_i(k) + B\breve{u}_i(k) \quad (11)$$

This can be verified by observing that the pairs $[\bar{x}_i, \bar{u}_i]$ obtained by solving (9) are always static. In others words, they are used to represent the distance between the common reference $[x_{ref}(k), u_{ref}(k)]$ and the reference of each agent $[\breve{x}_i(k), \breve{u}_i(k)]$. Hence we deduce $\breve{x}_i(k+1) = x_{ref}(k+1) + \bar{x}_i = Ax_{ref}(k) + Bu_{ref}(k) + \bar{x}_i = A\breve{x}_i(k) + B\breve{u}_i(k) - A\bar{x}_i - B\bar{u}_i + \bar{x}_i = A\breve{x}_i(k) + B\breve{u}_i(k)$ because of the condition $\bar{x}_i = A\bar{x}_i + B\bar{u}_i$.

The main purpose of the formation control remains the design a closed-loop control scheme so that the MAS's states

[1]The common reference is the reference of the formation center.

track the following common reference:

$$x_{ref}(k+1) = Ax_{ref}(k) + Bu_{ref}(k) \quad (12)$$

E. Tracking reference

In this section, the proposed control objective is two fold. Firstly, it is supposed to estimate the agents' real state via the corresponding nominal state. In fact, as presented in the previous section, the real state is always included in a set, called safety region, centered at the nominal state. Theses sets are used to find the minimal configuration for MAS.

Secondly, once the minimal configuration is determined, we can use a centralized Model Predictive Control technique to steer the agents to their target positions. This control is based on the knowledge of the nominal dynamics as illustrated by the following expression:

$$u^*(k) = \arg \min_{U(k+N_p|k)} \sum_{s=1}^{N_p} \|x(k+s) - \breve{x}(k+s)\|_Q$$

$$+ \sum_{s=0}^{N_p - 1} \|u(k+s) - \breve{u}(k+s)\|_R$$

subject to:

$$\begin{cases} x(k+s+1) = A_g x(k+s) + B_g u(k+s), \quad s \in \mathbb{N}_{[0,N_p]} \\ \breve{x}(k+s+1) = A_g \breve{x}(k+s) + B_g \breve{u}(k+s) \\ x_i(k+s) - x_j(k+s) \notin -\mathcal{S}_i \oplus \mathcal{S}_j, \quad s \in \mathbb{N}_{[1,N_p]} \end{cases}$$

$$(13)$$

We use a simple MPC formulation in (13) eluding the terminal cost and terminal constraints ingredients. This implies that the length of the prediction horizon will play an important role for the recursive feasibility and constraints satisfaction. Alternative formulations enforcing the stability can be employed but their design is out of the scope of the present paper which concentrates on the monitoring and fault detection of the formation which is supposed to run on a properly design tracking control mechanism.

III. PROBLEM STATEMENT

As mentioned in the previous sections, MIP techniques are appropriate to solve the optimization problem with non-convex constraints. In the reality, there are some cases where the MIP problem is not feasible due to the restrained computation of the complement set, specially when the formation of MAS is affected by severe fault and it is not taken into account by the control scheme. We aim to discuss next the adaptability of this solution to a reconfiguration of the subsystems.

A. Real time calculation

Let us consider a MAS composed of $N = 2$ agents. The constraint imposed to avoid the collision is written as $x_1 - x_2 \notin -\mathcal{S}_1 \oplus \mathcal{S}_2$. In others words, the 1st agent's position has to respect $x_1 \notin \{x_2\} \oplus (-\mathcal{S}_1) \oplus \mathcal{S}_2$ but it is constrained to belong to $C(\{x_2\} \oplus (-\mathcal{S}_1) \oplus \mathcal{S}_2)$. The situation becomes more complicated when the number of agents increases, because the position of one agent is included in the intersection of the complement of the safety region of the other agents. Practically, due to bounded representation of the sets, in the
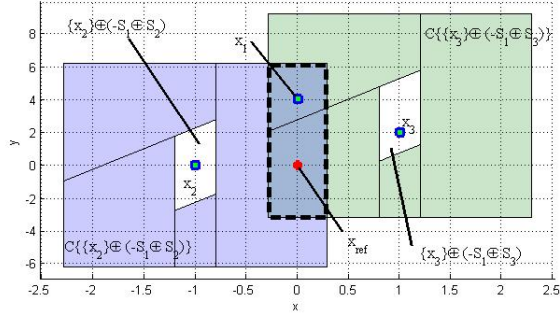
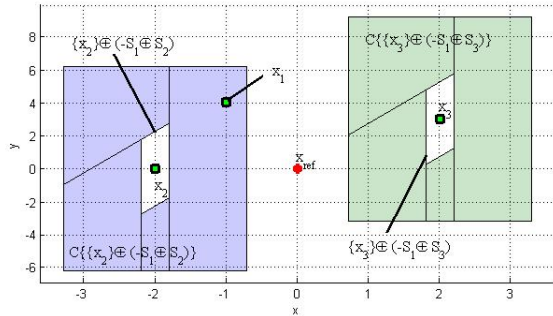Fig. 3: Feasible region for the optimal position of the 1st agent in a formation of $N = 3$ agents

Fig. 4: MIP infeasible computation

context of a centralized formation control approach, when an agent leaves the formation due to a fault or an abandon of the formation objective in the operation, the intersection of its safety region with the others is empty. Immediately the calculation of target position of the rest of formation will fail as long as the position of all agents are calculated in the same time, and each position depends strictly on the position of all remaining agents in the formation. For example, consider the case of $N = 3$ agents, the constraint imposed on the 1st agent is:

$$\begin{cases} x_1 \notin \{x_2\} \oplus (-S_1) \oplus S_2 \\ x_1 \notin \{x_3\} \oplus (-S_1) \oplus S_3 \end{cases}$$

or precisely:

$$x_1 \in C(\{x_2\} \oplus (-S_1) \oplus S_2) \cap C(\{x_3\} \oplus (-S_1) \oplus S_3)$$

as depicted in Figure 3. Otherwise, Figure 4 illustrates the case where the set of constraints renders the optimization infeasible.

B. Task assignment reconfiguration

In [6], the authors proposed a method to compute the minimal configuration for MAS in the off-line stage. In fact, this configuration can be affected by the change of the number of agents, so it should be calculated online. This case is meaningful when an agent leaves definitely its team[2], due to a serious fault or when the operator decides to take it

[2]Practically it may become even adversary with respect to the team but such behavior is not considered here.

out of the team. However in both cases the formation does not change and then the agents keep following the fixed formation which is not minimal formation yet.

IV. ONLINE FAULT TOLERANT SCHEME FOR MAS

As presented, previous work in [6] succeeded to find a minimal configuration for MAS in the off-line stage. Hence our proposition is to recalculate online this configuration by updating the current positions of the agents. By this way we can detect which agent is faulty and then reconfigure the formation by isolating the eliminated agent and subsequently using the previously computed control input to steer the agents to their positions in the new formation. For brevity, in the sequel, $x_{d,i}(k)$ denotes the real state of the ith agent. Moreover, $x_{d,i}(k)$ is assumed to be measurable. Two cases of faults are treated in this section:

- Quarantined Fault
- Eliminated Fault

For the first one, we need a certain time to certify the faulty status of the agent. For the second one, the fault certificate needs to be fulfilled as soon as possible.

A. Quarantined Fault case

1) Fault Detection and Isolation: In order to check when the reconfiguration is activated, or to determine the functioning mode (Healthy or Faulty) of an agent, a set of N residuals will be used. Each agent will be associated with one residual. Each residual is defined as:

$$r_i(k) = x_{d,i}(k) - \check{x}_i(k), \text{ with } i \in \mathbb{N}_{[1,N]} \tag{14}$$

The value of $\check{x}_i(k)$ is obtained by using the nominal dynamics (4) and the last state $x_i(k-1)$ i.e.:

$$\check{x}_i(k) = Ax_i(k-1) + B_i u_i^*(k-1), \text{ with } i \in \mathbb{N}_{[1,N]} \tag{15}$$

with $u_i^*(k-1)$ is the ith element of the optimal solution of (13) at time instant $k-1$.

Subsequently, we use $\mathcal{R}_i^H$ to define the Healthy functioning of the ith agent. In fact, if there is no fault, $r_i(k) \in \mathcal{R}_i^H$ where $\mathcal{R}_i^H$ is the safety region $\mathcal{S}_i$, i.e.:

$$\mathcal{R}_i^H = \mathcal{S}_i, \text{ with } i \in \mathbb{N}_{[1,N]} \tag{16}$$

We take a set named $\mathcal{R}_i^F$ to characterize the Faulty functioning and $\mathcal{R}_i^{H \to F}$ to characterize the Healthy-to-Faulty transition functioning. These sets must respect the following detectability set-separation condition:

$$\begin{cases} \mathcal{R}_i^H \cap \mathcal{R}_i^{H \to F} = \oslash \\ \mathcal{R}_i^H \cap \mathcal{R}_i^F = \oslash \end{cases}, \text{ with } i \in \mathbb{N}_{[1,N]} \tag{17}$$

If this separation does not hold, then the FDI mechanism will not be able to ensure one-step detection but can engage a monitoring procedure [7].

In fact, in this paper we consider just the critical case of fault, like leaving the formation due to serious fault. The fault identification is not considered in this paper. This means that once the fault occurs, the residual r_i jumps out of $\mathcal{R}_i^H$

and gets in $\mathcal{R}_i^F$ immediately. The condition of FDI (17) is simplified to:

$$\mathcal{R}_i^H \cap \mathcal{R}_i^F = \oslash, \text{ with } i \in \mathbb{N}_{[1,N]} \qquad (18)$$

A candidate set which satisfies the condition (18) is $\mathcal{R}_i^F = C(\mathcal{R}_i^H)$.

2) Reconfiguration Mechanism: This section proposes a Reconfiguration Mechanism which is activated when an agent is suspected faulty. We define here a set called *confidence formation set $\mathcal{S}$*:

$$\mathcal{S} = ConvexHull\left(\bigcup \mathcal{S}(\bar{x}_i)\right), \forall i \in \mathcal{N}_R \qquad (19)$$

with $\mathcal{N}_R = \mathbb{N}_{[1,N]} \backslash \mathcal{N}_E$ denoting the set of the remaining agents' indices.

Moreover, the MAS is equipped with a set of N timers, one for each agent. At each iteration k, we check if $r_i(k)$ is included in $\mathcal{R}_i^H$ or not. If $r_i(k) \notin \mathcal{R}_i^H$, the corresponding timer will be activated, corresponding to a "quarantined status" for the respective agent. When the timer counts N_m iterations, if $r_i(k + N_m) \notin \mathcal{R}_i^H$, the ith agent will be eliminated from the team and the formation is reconfigured at the next iteration for the remaining agents. This new formation is obtained by resolving (9) for the $\mathcal{N}_R$ subset of agents. Hence the confidence set is also reconfigured.

Moreover, we propose to calculate a supplementary stage of optimization, to determine the target position of the eliminated agent outside of $\mathcal{S}$:

$$\min_{\bar{u}_i} \sum_{i \in \mathcal{N}_E} \|\bar{x}_i\|$$
$$\text{st: } \begin{cases} \bar{x}_i - \bar{x}_j \notin -\mathcal{S}_i \oplus \mathcal{S}_j, \forall i,j \in \mathcal{N}_E, i \neq j \\ \bar{x}_i \notin -\mathcal{S}_i \oplus \mathcal{S}, \forall i \in \mathcal{N}_E \\ \bar{x}_i = A\bar{x}_i + B\bar{u}_i \end{cases} \qquad (20)$$

The remaining healthy agents will be steered to the new formation while the trajectory $\breve{x}_i$ of the eliminated one is hold outside of $\mathcal{S}$. This is meaningful for the formation safety when this agent tries to reintegrate into the formation. Precisely, by imposing this temporary position, we can avoid the collision between the returned agent and the agents inside of $\mathcal{S}$. When this agent reaches this temporary position, its index will be added in $\mathcal{N}_R$ and then the formation will be reconfigured.

B. Eliminated Fault case

The previous FDI scheme is used to detect the quarantined fault on each agent. In others words, a quarantined fault needs a certain time to certify the faulty status of agent. In the reality, there are some cases where the fault detection needs to be activated as soon as possible. The threshold set for these case is parameterized the same way as (7), i.e:

$$\breve{\mathcal{S}}(x_{ref}(k)) = ConvexHull\left(\bigcup \mathcal{S}(\breve{x}_i(k))\right), \forall i \in \mathcal{N}_R \qquad (21)$$

At each iteration k, we check if $x_{d,i}(k)$ is included in this set or not. If $x_{d,i}(k) \notin \breve{\mathcal{S}}(x_{ref}(k))$ the ith agent will be eliminated immediately from the team and the formation is reconfigured at the next iteration in the same way as the quarantined fault case.

C. Algorithm of FDI

All the above ideas are resumed by the *Algorithm 1* for the task assignment of the $\mathcal{N}_R$ healthy subset of agents. This algorithm is executed at each sampling time.

The functioning mode of each agent is represented by:

$$status_i = \begin{cases} 1 \text{ if Healthy} \\ 0 \text{ if Faulty} \end{cases}$$

The status (activated/deactivated) of each timer is described by the corresponding element in the vector $timer = [timer_1 \ timer_2 \dots timer_N]^\top$, with $timer_i = \begin{cases} 1 \text{ if activated} \\ 0 \text{ if deactivated} \end{cases}$

V. ILLUSTRATIVE EXAMPLE

In this section, a numerical example is presented. It shows the results obtained by applying the *Algorithm 1* on a formation of $N = 3$ agents. Each agent is described by its nominal dynamics equation (3), with $A_i = \begin{bmatrix} 0.45 & 0.20 \\ -0.54 & 1.14 \end{bmatrix}$, $B_i = \begin{bmatrix} 0.08 & 0 \\ 0 & 0.85 \end{bmatrix}$ and $i = 1, 2, 3$.

For the MPC controller, the weighting matrices are $Q = 100I_n$, $R = 0.01I_m$. The length of the prediction horizon is $N_p = 5$. The monitoring horizon is $N_m = 6$. The disturbance which affects the agents is bounded by $\bar{w} = [0.1 \ 0.2]^\top$.

These three agents have to enter in a minimal configuration and hence, the entire formation have to track a predefined reference. The safety sets are constructed by following the method presented in the Section II. We use the pole placement technique to find the closed-loop gains K_i for each agent. The chosen poles are $[0.2; 0.3]$.

At the moment of occurrence of a fault on the 2nd agent's actuators, which is regarded as a change in the matrix B_2, clearly the 2nd agent starts leaving the team. As illustrated in Figure 5, the formation is reconfigured immediately once the updated status of the 2nd agent is determined Faulty. Firstly the reference position of the 2nd agent is isolated from the confidence set $\mathcal{S}$. After that, the reference governor recalculates another optimal configuration for the remaining two agents. The confidence set is also reconfigured base on this new configuration. Finally, the trajectories are generated by taking into account this new confidence set.

VI. CONCLUSION

In this paper we treated the simplest case of fault strategies: definitive elimination of agent from a Multi-Agent formation. We proposed a new algorithm to find the optimal formation which allows the online reconfiguration for the formation in a faulty situation. Otherwise, it allows to construct in real time the threshold set for fault detection or the safety region of agents. The off-line calculation of the residual sets allows to reduce the computation resources. Future work will focus on the feasibility of the Mixed Integer Programming computation in real time. In addition, it is necessary to consider other cases in which we can identify and locate the fault and after that we can try to recover

Algorithm 1: Task Assignment Reconfiguration with Fault Detection capabilities

Data: current position of all agents, residual set $\mathcal{R}_i^H$, $\mathcal{R}_i^F$

Result: Minimal reconfigured formation

1 - construct $\check{\mathcal{S}}(x_{ref})$ for $\mathcal{N}_R$;
2 **for** $i \in \mathcal{N}_R$ **do**
3   **if** $x_{d,i}(k) \notin \check{\mathcal{S}}(x_{ref}(k))$ **then**
4        $status_i := 0$;
5        $\mathcal{N}_E := \mathcal{N}_E \cup \{i\}$;
6   **else**
7   - calculate residual $r_i(k) = x_{d,i}(k) - \check{x}_i(k)$;
8   **if** $timer_i = 1$ **then**
9   **if** $r_i(k) \notin \mathcal{R}_i^H$ **then**
10   **if** $t = N_m$ **then**
11               $status_i := 0$;
12               $\mathcal{N}_E := \mathcal{N}_E \cup \{i\}$;
13   **else**
14               $t := t + 1$;
15   **end**
16   **else**
17             $timer_i := 0$;
18             $status_i := 1$;
19             $t := 0$;
20   **end**
21   **else**
22   **if** $r_i(k) \notin \mathcal{R}_i^H$ **then**
23             $timer_i := 1$;
24             $status_i := 1$;
25             $t := 1$;
26   **else**
27             $timer_i := 0$;
28             $status_i := 1$;
29             $t := 0$;
30   **end**
31   **end**
32   **end**
33 **end**
34 - reconfigure the confidence set $\mathcal{S}$ for $\mathcal{N}_R := \mathbb{N}_{[1,N]} \backslash \mathcal{N}_E$;
35 - solve (9) for $\mathcal{N}_R$;
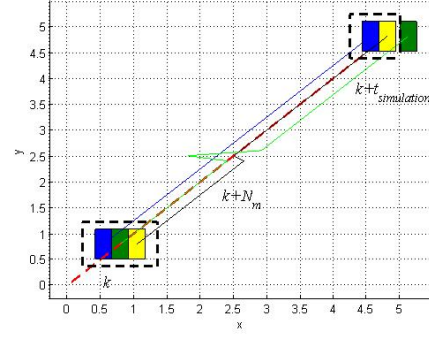36 - solve (20) for $\mathcal{N}_E$;

Fig. 5: Trajectories of three agents (blue, green, yellow). Common reference of MAS (red dash line). Confidence formation set $\mathcal{S}$ (black dash line)

REFERENCES

[1] N. Meskin and K. Khorasani, "Actuator fault detection and isolation for a network of unmanned vehicles," *IEEE Trans. Automat. Contr.*, vol. 54(4), pp. 835–840, 2009.

[2] N. Meskin and K. Khorasani, "A geometric approach to fault detection and isolation of continuous-time markovian jump linear systems," *IEEE Trans. Automat. Contr.*, vol. 55(6), pp. 1343–1357, 2010.

[3] N. Meskin and K. Khorasani, "A hybrid fault detection and isolation strategy for a network of unmanned vehicles in presence of large environmental disturbances," *IEEE Trans. Contr. Syst. Technol.*, vol. 18(6), pp. 1422–1429, 2010.

[4] S. Stanković, N. Ilíc, Z. Djurovíc, M. Stankovíc, and K. H. Johansson, "Consensus based overlapping decentralized fault detection and isolation," in *Conference on Control and Fault Tolerant Systems*, Nice, France, 2010.

[5] G. Antonelli, F. Arrichiello, F. Caccavale, and A. Marino, "A decentralized controller-observer scheme for multi-agent weighted centroid tracking," *IEEE Trans. Automat. Contr.*, vol. 58(5), pp. 1310–1316, 2013.

[6] I. Prodan, S. Olaru, C. Stoica, and S. Niculescu, "Predictive control for tight group formation of multi-agent systems," in *Proc. of IFAC World Congress*, Milan, Italy, 2011.

[7] F. Stoican, "Fault tolerant control based on set-theoretic methods," Ph.D. thesis, Supelec, 2011.

[8] F. Stoican, S. Olaru, and G. Bitsoris, "A fault detection scheme based on controlled invariant sets for multisensor systems," in *Conference on Control and Fault Tolerant Systems*, Nice, France, 2010.

[9] F. Stoican, S. Olaru, M. M. Seron, and J. D. Dona, "A fault tolerant control scheme based on sensor-actuation channel switching and dwell time," in 49th *IEEE Conference on Decision and Control*, Atlanta, USA, 2010.

[10] M. M. Seron and J. A. De Dona, "Fault tolerant control using virtual actuators and invariant-set based fault detection and identification," in 48th *IEEE Conference on Decision and Control and* 28th *Chinese Control Conference*, Shanghai, China,, 2009.

[11] G. Franze, F. Tedesco, and D. Famularo, "Actuator fault tolerant control: a set-theoretic approach," in *IEEE Conference on Decision and Control*, Hawaii, USA, 2012.

[12] I. Prodan, "Commande sous contraintes de systemes dynamiques multi-agents," Ph.D. thesis, Supelec, 2012.

[13] F. Blanchini and S. Miani, *Set-theoretic methods in control*. Birkhauser, 2007.

the entire system to maintain the tracking reference. Face to the computation ability, a decentralized approach, where the optimization problem is solved at the agent's level will be further considered.

VII. ACKNOWLEDGMENT