About formation reconfiguration for Multi-Agent dynamical Systems

Minh Tri Nguyen, Cristina Stoica Maniu, Sorin Olaru, Alexandra Grancharova

ABOUT FORMATION RECONFIGURATION FOR MULTI-AGENT DYNAMICAL SYSTEMS

M. T. Nguyen[1], C. Stoica Maniu[1], S. Olaru[1], A. Grancharova[2]

[1] 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

[2] Institute of System Engineering and Robotics, Bulgarian Academy of Sciences, Acad G. Bonchev str., Bl.2, P.O.Box 79, Sofia 1113, Bulgaria, e-mail: alexandra.grancharova@abv.bg

Abstract: The present paper addresses the task assignment problem for an homogeneous Multi-Agent system, face to high mission safety requirements. Recently by using set-theoretic methods, this problem has been formulated in terms of an optimization problem allowing to keep the agents in a tight formation in real-time via task reallocation and classical feedback mechanism. In this paper we propose to solve this problem in view of real-time control by including fault detection and isolation capabilities.

Key words: Multi-agent dynamical systems, Fault Detection and Isolation, set-membership theory, tight formation control.

I. INTRODUCTION

Nowadays, Multi-Agent system (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 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 MAS. The FDI scheme is supposed not to affect the stability of the formation control and to preserve the collision avoidance guarantees.

Recently results have been reported on the application of set-theoretic and optimization tools for MAS control, notable [1]. Furthermore, these tools were also used to design FDI schemes based on the separation between different functioning modes [2]. The faults treated in this brand are sensor fault [3], [4] and actuator fault [5], [6] for single systems. Our idea is to employ them to design a Fault Tolerant Control (FTC) for Multi-Agent systems. The starting point is to consider 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 system is completed by a reconfiguration step to design 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 to apply set-theoretic tools to solve the task assignment in real time, when considering actuator faults. The remainder of the paper is organized as follows. Section II presents useful preliminary results and the problem statement. Section III presents a new FDI scheme with respect to the real time formation preservation. Section IV proposes an example to illustrate the performance of this new FDI scheme. Finally, some concluding remarks and perspectives are mentioned in Section V.

II. PRELIMINARY RESULTS AND PROBLEM STATEMENT

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.

<u>*Notations*</u>.

$\mathcal{A} \oplus \mathcal{B}$ denotes the *Minkovski sum* of two sets $\mathcal{A}$ and $\mathcal{B}$.

$\mathbb{N}_{[1,N]} \triangleq \{0,1,2,\ldots,N\}$ 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 $\mathcal{N}_R = \mathbb{N}_{[1,N]} \setminus \mathcal{N}_E$.

The length of the prediction horizon for the Model Predictive Control (MPC) law is denoted as N_p.

$\tilde{x}_i(k)$ denotes the *one-step predicted state* of the i^{th} agent.

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

<u>*Ultimate bound invariant set*</u>

Theorem 1 [1] Consider the 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

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

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

<u>*Dynamics equation for the network of agents*</u>. Consider a MAS composed of *N* agents. Each agent is characterized by a discrete-time dynamics equation:

$$x_{d,i}(k+1) = A_i x_{d,i}(k) + B_i u_{d,i}(k) + w_i(k), \text{ with } i \in \mathbb{N}_{[1,N]} \tag{2}$$

where $x_{d,i}(k) \in \mathbb{R}^n$ is the i^{th} 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 [1], with $A_i \in \mathbb{R}^{n\times n}$ and $B_i \in \mathbb{R}^{n\times m}$.

The nominal dynamics corresponding to (2) is [1]:

$$x_i(k+1) = A_i x_i(k) + B_i u_i(k), \text{ with } i \in \mathbb{N}_{[1,N]} \tag{3}$$

where $x_i(k) \in \mathbb{R}^n$ and $u_i(k) \in \mathbb{R}^m$. The control input in (2) is defined as $u_{d,i}(k) = u_i(k) + K_i\left[x_{d,i}(k) - x_i(k)\right]$. Denoting by $e_i(k) = x_{d,i}(k) - x_i(k)$ the tracking error of the i^{th} agent, we obtain the following equation:

$$e_i(k+1) = (A_i + B_i K_i) e_i(k) + w_i(k), \text{ with } i \in \mathbb{N}_{[1,N]} \tag{4}$$

The stabilizability assumption of the pairs (A_i, B_i) conducts to the existence of $K_i \in \mathbb{R}^{m\times m}$ which stabilizes (4), so we can construct a *robustly positive invariant* (RPI) set $\mathcal{S}_{e_i}$ (see [1]) which satisfies:

$$e_i(k) \in \mathcal{S}_{e_i}, \forall k \geq k_0 \text{ and } \forall e_i(k_0) \in \mathcal{S}_{e_i} \tag{5}$$

This RPI set $\mathcal{S}_{e_i}$ is considered as the safety region around each agent. Furthermore, although the real state $x_{d,i}(k)$ is unknown due to $w_i(k)$, its trajectory is always bounded by the tube (see Fig. 1):

$$\mathcal{S}_i(x_i(k)) \triangleq x_i(k) \oplus \mathcal{S}_{e_i}(k) \tag{6}$$

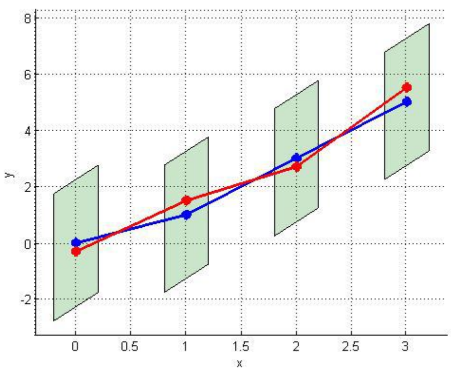

Fig. 1. Nominal trajectory (blue). Real trajectory (red). RPI set of tracking error (green).

The global Multi-Agent system can be defined as:

$$x(k+1) = A_g x(k) + B_g u(k) \tag{7}$$

where $x(k) = \left[x_1^{\mathrm{T}}(k) \quad x_2^{\mathrm{T}}(k) \quad \ldots \quad x_N^{\mathrm{T}}(k)\right]^{\mathrm{T}} \in \mathbb{R}^{Nn}$ and $u(k) = \left[u_1^{\mathrm{T}}(k) \quad u_2^{\mathrm{T}}(k) \quad \ldots \quad u_N^{\mathrm{T}}(k)\right]^{\mathrm{T}} \in \mathbb{R}^{Nm}$ 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 $B_g = diag(B_1, B_2, \ldots, B_N) \in \mathbb{R}^{Nn\times Nm}$. For the global homogeneous system we use A to denote A_i and B to denote B_i .

<u>*Minimal configuration*</u>. This section resumes how to obtain a minimal reconfiguration for a MAS defined as (7), via solving the following optimization problem:

$$\min_{\bar{u}_i} \sum_{i=1}^{N} \left\|\bar{x}_i\right\|$$
$$\text{st:}\begin{cases} \bar{x}_i - \bar{x}_j \notin -\mathcal{S}_i \oplus \mathcal{S}_j, \forall i,j \in \mathbb{N}_{[1,N]}, i \neq j \\ \bar{x}_i = A\bar{x}_i + B\bar{u}_i \end{cases} \tag{8}$$

$\bar{x}_i$ indicates the error between the state of the i^{th} agent and the origin. $\mathcal{S}_i$ represents the *safety region* of one agent.

The objective is to determinate the closest position of all the agents around the *common reference* (i.e. the reference of the formation center) while avoiding the collision (see Fig. 2). Moreover, these target positions have to be equilibrium/stationary points relative to the dynamics (2).

Due to the non-convexity of these constraints (i.e. an agent is allowed to stay in the complement of the (convex) safety region), this problem can be solved by using *Mixed Integer Programming* (MIP). Its solution is the set of target positions for the group of agent. Once the formation is determined, it will be preserved along the common reference x_{ref} , as depicted in Fig. 2.

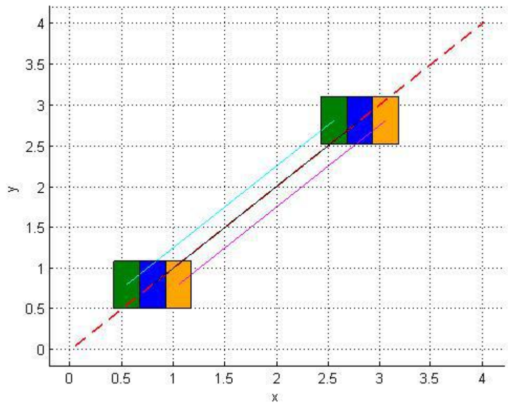

Fig. 2. Trajectories of each agent around the common reference (red dash-line).

Hence the target trajectory of the i^{th} agent is denoted by:

$$\begin{aligned} \breve{x}_i(k) &= x_{ref}(k) + \bar{x}_i \\ \breve{u}_i(k) &= u_{ref}(k) + \bar{u}_i \end{aligned} \tag{9}$$

It is associated with the dynamics equation:

$$\breve{x}_i(k+1) = A\breve{x}_i(k) + B\breve{u}_i(k) \tag{10}$$

The main purpose remains to design a closed-loop control scheme so that the MAS's states track the following common reference:

$$x_{ref}(k+1) = Ax_{ref}(k) + Bu_{ref}(k) \tag{11}$$

<u>*Tracking reference*</u>. 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:

$$u^*(k) = \arg \min_{U(k+N_p|k)} \sum_{s=1}^{N_p} \left\| x(k+s) - \breve{x}(k+s) \right\|_Q + \sum_{s=0}^{N_p-1} \left\| u(k+s) - \breve{u}(k+s) \right\|_R \qquad (12)$$

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

__Problem statement__. In [1], the minimal configuration for MAS is defined in the off-line stage. However, this predefined configuration does not adapt to the change of the number of agents, typically when an agent leaves definitely its team[1], due to a serious fault or when the operator decides to take it out of the team.

III. REAL TIME FAULT TOLERANT SCHEME FOR MAS

A new FDI scheme is proposed in this section. It is used to detect and isolate the faulty agents from the formation, then reconfigure the formation based on the healthy remaining agents. In the sequel, for brevity, $x_{d,i}(k)$ denotes the real state measurable of the i^{th} agent.

__Quarantined Faulty Agent Detection__. In order to determine the functioning mode (Healthy or Faulty) of an agent, a set of N residuals will be used, one for each agent. Each residual is defined as:

$$r_i(k) = x_{d,i}(k) - \tilde{x}_i(k), \text{with } i \in \mathbb{N}_{[1,N]} \qquad (13)$$

with $\tilde{x}_i(k)$ denoting the one-step predictable state of the i^{th} agent. The value of $\tilde{x}_i(k)$ is obtained by using the nominal dynamics (3) and the last state $x_i(k-1)$ i.e.:

$$\tilde{x}_i(k) = Ax_i(k-1) + B_i u_i^*(k-1), \text{with } i \in \mathbb{N}_{[1,N]} \qquad (14)$$

with $u_i^*(k-1)$ is the i^{th} element of the optimal solution of (12) at time instant $k-1$.

If there is no fault then $r_i(k) \in \mathcal{S}_i$ with respect to (5). Hence the safety region $\mathcal{S}_i$ is also the set $\mathcal{R}_i^H$ which characterizes the Healthy functioning of the i^{th} agent:

$$\mathcal{R}_i^H = \mathcal{S}_i, \text{with } i \in \mathbb{N}_{[1,N]} \qquad (15)$$

Once $r_i(k) \notin \mathcal{R}_i^H$, certainly the i^{th} agent is faulty but the fault nature is not yet identified. We can just confirm that this agent is quarantined faulty.

__Faulty Agent Certificate__. To detect the fault mentioned in the problem statement description, we define a threshold set as:

$$\tilde{\mathcal{S}}\left(x_{ref}(k)\right) = ConvexHull\left(\bigcup \mathcal{S}_i\left(\tilde{x}_i(k)\right)\right), \forall i \in \mathcal{N}_R \qquad (16)$$

At each iteration k, if $x_{d,i}(k) \notin \tilde{\mathcal{S}}\left(x_{ref}(k)\right)$ the i^{th} agent is certified faulty.

__Reconfiguration Mechanism__. This section proposes a Reconfiguration Mechanism which is activated when an agent is certificated faulty. At each iteration *k*, we check the quarantined faulty condition and also the faulty certificate for all agents (see Fig. 3).

If $x_{d,i}(k) \notin \tilde{\mathcal{S}}\left(x_{ref}(k)\right)$ (it covers $r_i(k) \notin \mathcal{R}_i^H$), the i^{th} 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 (8) for the $\mathcal{N}_R$ subset of agents.

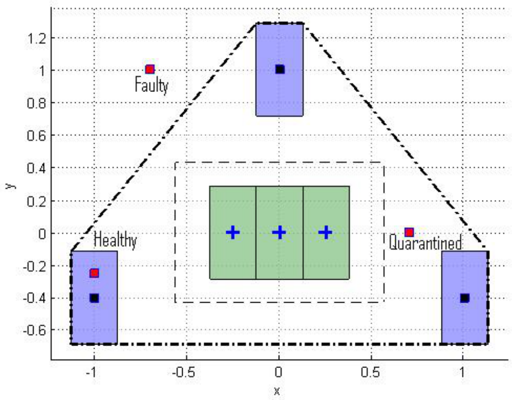

Fig. 3. Real time Fault detection and Isolation. Current minimal configuration (blue plus sign). Predicted threshold set $\tilde{\mathcal{S}}\left(x_{ref}(k)\right)$ (dash-dot line). One-step predicted state $\tilde{x}_i(k)$ (black square). Real measured state $x_{d,i}(k)$ (red square).

__Eliminated Agent Isolation__. Once the minimal configuration is determined, a set named confidence formation set $\mathcal{S}$ is defined as:

$$\mathcal{S} = ConvexHull\left(\bigcup \mathcal{S}(\overline{x}_i)\right), \forall i \in \mathcal{N}_R \qquad (17)$$

This set allows to isolate the remaining agents from the collision with the eliminated agents. Moreover, it is also used to calculate the target position for the eliminated agents outside of $\mathcal{S}$ (see Fig. 4):

$$\min_{\overline{u}_i} \sum_{i \in \mathcal{N}_E} \left\| \overline{x}_i \right\|$$

$$\text{st:} \begin{cases} \overline{x}_i - \overline{x}_j \notin -\mathcal{S}_i \oplus \mathcal{S}_j, \forall i, j \in \mathcal{N}_E, i \neq j \\ \overline{x}_i \notin -\mathcal{S}_i \oplus \mathcal{S}, \forall i \in \mathcal{N}_E \\ \overline{x}_i = A\overline{x}_i + B\overline{u}_i \end{cases} \qquad (18)$$

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 the formation, which can perturb the safety of the formation, typically by collision, because the index of the returned one is not yet taken into account.

__Safety reintegration__. Apart from the faulty agent detection and isolation, we propose here another scheme for the safety reintegration of the eliminated agents. The main purpose is to detect which agent want to return to the formation and conserve the safety of the global system during the reintegration process.

---

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

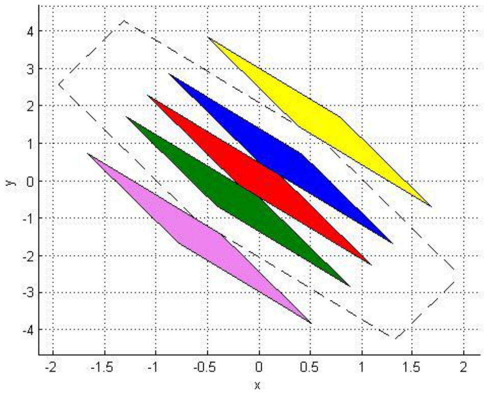

Fig. 4. Task assignment reallocation. Confidence set $\mathcal{S}$ (dash-line)

Firstly, the returned agent must be certified Healthy ($r_i(k) \in \mathcal{R}_i^H$ is checked). After that, its index is added to $\mathcal{N}_R$ if and only if $x_{d,i}(k) \in (-\mathcal{S}_i) \oplus \tilde{\mathcal{S}}(x_{ref}(k))$, with $i \in \mathcal{N}_E$.

When these two conditions are checked, a new configuration will be generated for the new subset $\mathcal{N}_R$.

It is possible to detect simultaneously multiple returned agents, but the reintegration mechanism allows just one unique agent to enter into the new formation each time. A priority reintegration scheme can be further considered.

IV. ILLUSTRATIVE EXAMPLE

In this section, a numerical example is presented. It shows the results obtained by applying the new FDI scheme on a formation of $N = 3$ agents. Each agent is described by its nominal dynamics equation (2), with:

$$A_i = \begin{bmatrix} 0.45 & 0.20 \\ -0.54 & 1.14 \end{bmatrix},\ B_i = \begin{bmatrix} 0.08 \\ 0.85 \end{bmatrix} \text{ and } i = 1,2,3.$$

For the MPC controller, the weighting matrices are $Q = 100 I_n$, $R = 0.01 I_m$. The length of the prediction horizon is $N_p = 5$. The disturbance which affects the agents is bounded by $\bar{w} = [0.1 \quad 0.2]^{\mathrm{T}}$. 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 $[0.2 \quad 0.3]^{\mathrm{T}}$.

Let us consider a fault occurring on the actuator of the 2^{nd} agent (as shown in Fig. 5). This fault is detected when the condition $r_2(k) \notin \mathcal{R}_2^H$ is checked, but the agent's status is officially Faulty since $x_{d,2}(k) \notin \tilde{\mathcal{S}}(x_{ref}(k))$ is checked (instant k_F). So its index is eliminated from the $\mathcal{N}_R$ subset. A new minimal configuration is recalculated for the two remaining agents. As presented in the previous section, the position of this eliminated agent is still calculated providing that it can not hit the healthy remaining agents inside of the confidence region $\mathcal{S}$.

After that, this eliminated agent tries to reintegrate the current formation. At the instant k_I, two conditions $r_2(k) \in \mathcal{R}_2^H$ and $x_{d,2}(k) \in (-\mathcal{S}_2) \oplus \tilde{\mathcal{S}}(x_{ref}(k))$ are checked. Hence the status of the returned agent is determined Healthy and its index will be added in $\mathcal{N}_R$. Finally, the formation is reconfigured again for the new $\mathcal{N}_R$ subset of healthy agents at the next instant k_R.

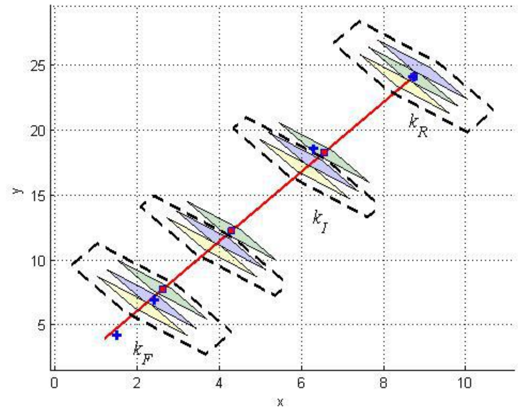

Fig. 5. Common reference of MAS (red). Confidence formation set $\mathcal{S}$ (black dash line). One-step predicted state $\tilde{x}_i(k)$ (red square). Real measured state $x_{d,i}(k)$ (blue plus sign).

V. CONCLUSION AND FUTURE DIRECTIONS

This paper presents a new Fault Detection and Isolation scheme to treat the simplest case of fault strategies: definitive elimination of faulty agents from a Multi-Agent formation. The main idea is firstly to detect and isolate the faulty agents, secondly to allow the online reconfiguration for the formation in a faulty situation. To improve the reconfiguration mechanism, future work will focus on the identification of the fault nature. Moreover, due to the high computational effort required for solving the centralized optimization problem, a decentralized approach (which implies solving the optimization problem at the agent's level) will be further considered.

ACKNOWLEDGMENT

The authors acknowledge the support of PHC RILA Project “Robust Distributed Predictive Control of Complex Systems”.

REFERENCES

1. 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.
2. F. Stoican, “Fault tolerant control based on set-theoretic methods”, Ph.D. thesis, Supelec, 2011.
3. 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.
4. 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.
5. M. M. Seron and J. A. D. 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.
6. G. Franze, F. Tedesco, and D. Famularo, “Actuator fault tolerant control: a set-theoretic approach”, *IEEE Conference on Decision and Control*, 2012.