Fault-Tolerant Formation Control of Passive Multi-Agent Systems Using Energy Tanks

Andrea Cristofaro, Gaetano Giunta, and Paolo Robuffo Giordano, *Senior Member, IEEE*

***Abstract*—This letter considers the design of a fault-tolerant formation control law for multi-agent systems using energy balancing methods. Decentralized algorithms for fault diagnosis and graph topology update are proposed, and a formation reconfiguration procedure is developed based on energy tanks, which are able to guarantee that new connections among the agents can be established in a passive way. A simulation study supports and corroborates the theoretical findings.**

***Index Terms*—Agents-based systems, fault tolerant systems, decentralized control.**

I. Introduction

SYSTEMS composed by multiple autonomous agents are useful in several applications where complex tasks have to be performed. Examples range from target tracking [1], to search and rescue operations [2] and to load transportation [3]. Many control problems involving networks of agents strongly hinge on graph theory and are typically related to driving all systems composing the network toward a *consensus* behavior [4] or to the more general problem of formation control, whose goal is to make the overall set of agents attain a desired geometrical arrangement in space [5].

The port-Hamiltonian framework is well recognized as a powerful setup for modeling and handling networked systems based on the useful interpretation of interconnections as energy exchanges. Energy-balancing methods allow indeed to infer stability properties of a multi-agent system from the inherent stability, and in particular the passivity, of each subsystem provided that proper interconnections are enforced. In this regard *energy tanks*, which were first introduced in [6], enable for a temporary storage of the energy that would be wasted due to the system dissipation which can then be re-used later on for temporarily implementing possible non-passive actions in a safe way. Example of possible applications are null-space projection [7], impedance control [8] and connectivity maintenance [9]. In particular, as highlighted in the latter, when agents navigate through a complex environment might need to split due to the presence of obstacles and other constraints, or may want to join to strengthen the group connectivity. However, when the inter-agent interactions are modeled as "spring-like" links, the creation of new links may not be a passive action in general and, under certain operative conditions, it might need the extraction of sufficient amount of energy from the tanks in order to be implemented with stability guarantees [10]. In this letter we aim at extending these ideas to the problem of passive fault-tolerant formation control. In the unfortunate event of a fault affecting one of the agents in the network, the overall system needs to react and to perform a formation reconfiguration, see for example [11]–[16] and the references therein.

Compared to classical systems, one has to face additional challenges when dealing with fault-tolerant control of multi-agent systems, mainly related to the need of devising decentralized schemes. In fact, actions must be undertaken by the agents using only the information that is locally available depending on the network topology. In light of this, tasks such as fault detection/isolation and topology update might become quite complex when there is a lack of global information due to redundancy issues [17]. This motivates the current study, whose main contribution is two-fold:

- Achieve decentralized fault diagnosis using a bank of observers whose estimated states are post-processed by a consensus-like protocol with the aim of reaching an agreement about the vehicle claimed to be faulty;
- Design an operative procedure to reconfigure the formation control law in a passive way based on energy tanks regardless of the particular new chosen formation.

Furthermore, as an additional task, we propose an effective algorithm for the decentralized update of the network topology under the assumption that the distribution of edges in the underlying graph has a certain pattern. The remainder of this letter is structured as follows. The basic setup is described in Section II, while the main contributions are given in Section III. Numerical simulations are reported in Section IV to illustrate the proposed results and, finally, some conclusions are drawn in Section V.

II. Problem Setup

We consider a group of $N \in \mathbb{N}$ agents, each of them being modeled as a free-floating mass $q \in \mathbb{R}^n$ with second-order

Manuscript received January 12, 2022; revised March 23, 2022; accepted April 15, 2022. Date of publication April 21, 2022; date of current version May 2, 2022. Recommended by Senior Editor C. Seatzu. *(Corresponding author: Andrea Cristofaro.)*

Andrea Cristofaro and Gaetano Giunta are with the Department of Computer, Control and Management Engineering, Sapienza University of Rome, 00185 Rome, Italy (e-mail: andrea.cristofaro@uniroma1.it).

Paolo Robuffo Giordano is with CNRS, Univ Rennes, Inria, IRISA, Campus de Beaulieu, 35042 Rennes, France.

Digital Object Identifier 10.1109/LCSYS.2022.3169308

linear dynamics

$$M_i\ddot{q}_i = -B_i\dot{q}_i + u_i \qquad i = 1, 2, \dots, N \tag{1}$$

where $M_i, B_i \in \mathbb{R}^{n\times n}$ indicate, respectively, the inertia matrix and the dissipation matrix. Although such dynamics is quite simple and easy to handle, it is commonly used also to model nonlinear systems which are controlled in cascade with hierarchical schemes [18], differential flatness or dynamic feedback linearization [19].

Considering the generalized momentum $p_i = M_i\dot{q}_i$, equation (1) rewrites as

$$\dot{p}_i = -B_iM_i^{-1}p_i + u_i = -B_i\frac{\partial \mathcal{K}_i}{\partial p_i} + u_i \tag{2}$$

with associated kinetic energy $\mathcal{K}_i(p_i) = \frac{1}{2}p_i^\top M_i^{-1}p_i$ and output $v_i = M_i^{-1}p_i$. The agents are interconnected according to an underlying communication/sensing graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $|\mathcal{V}| = N$, which is assumed to be connected and, for the sake of simplicity, undirected. Nevertheless, the extension to digraphs can be done by minor adjustements.

The input u_i is split in two components, the coupling force u_i^a and the external force u_i^{ext}. Typically, the agents are interconnected by means of the coupling force u_i^a and, in addition, leader agents are commanded by the external input u_i^{ext} used for assigning tasks to the whole multi-agent system, such as set-point regulation or trajectory tracking.

The geometry of the formation is regulated by means of generalized virtual springs ξ_{ij}, associated to a potential energy $V(\xi_{ij})$. Such potential $V(\cdot)$ is assumed to be a continuously differentiable function with the following features:

- lower-bounded
- has an absolute minimum at d_0
- $\lim_{|z|\to 0} V(z) = +\infty$
- $V(z) = \bar{V}$ for $|z| \geq d_{\max}$

The idea is that the energy is minimized when the virtual spring is at rest, namely for $\xi_{ij} = d_0$, this corresponding to the desired relative position between agent i and agent j. Furthermore, the parameter $d_{\max} > 0$ allows imposing a maximum connection range among the agents and an artificial repulsive force arises when two agents become too close in order to avoid possible collisions. Accordingly, the virtual spring dynamics is defined by

$$\begin{aligned}\dot{\xi}_{ij} &= w_{ij}\\ F_{ij}^a &= \frac{\partial V(\xi_{ij})}{\partial \xi_{ij}}\end{aligned}$$

with power-preserving interconnection

$$\begin{bmatrix} u_i^a \\ u_j^a \\ w_{ij}\end{bmatrix} = \begin{bmatrix} 0 & 0 & -\sigma_{ij} \\ 0 & 0 & \sigma_{ij} \\ \sigma_{ij} & -\sigma_{ij} & 0\end{bmatrix}\begin{bmatrix} M_i^{-1}p_i \\ M_j^{-1}p_j \\ F_{ij}^a\end{bmatrix} \tag{3}$$

where $\sigma_{ij} \in \{0, 1\}$ is a switching parameter used to enable/disable the neighboring condition among agent i and agent j, according to proximity of agents and/or availability of communication links.

III. Fault-Tolerant Formation Control

A. A Decentralized Fault Isolation Scheme

A fault affecting one of the agent may propagate its effects on the whole formation, posing serious threats to overall stability and safety. It is then desirable to implement efficient and, possibly decentralized, fault detection and isolation schemes. To this end, let us begin with the synthesis of a bank of decentralized observers based on relative velocity measurements only.[1] Let us set

$$\eta_{ij} := v_i - v_j = M_i^{-1}p_i - M_j^{-1}p_j$$

which corresponds to the available relative measurement among the agents i and j, used also in the control law (3). We define the observer for agent i relative to agent j

$$\begin{aligned}\dot{\hat{v}}_{i,j}^{(i)} &= -M_i^{-1}B_i\hat{v}_{i,j}^{(i)} + M_i^{-1}u_i + K_i(\eta_{ij} - \hat{\eta}_{ij})\\ \dot{\hat{v}}_{j,i}^{(i)} &= -M_j^{-1}B_j\hat{v}_{j,i}^{(i)} + M_j^{-1}u_j - K_j(\eta_{ij} - \hat{\eta}_{ij})\\ \hat{\eta}_{ij} &= \hat{v}_{i,j}^{(i)} - \hat{v}_{j,i}^{(i)}\end{aligned} \tag{4}$$

where K_i, K_j are suitable gain matrices.

Proposition 1: Assume that matrices $P = P^T \succ 0$, K_i, K_j and a scalar $\mu > 0$ can be found such that the matrix inequality $P\Omega + \Omega^TP \prec -\mu I_{2n\times 2n}$ is verified, where

$$\Omega = \begin{bmatrix} -M_i^{-1}B_i - K_i & K_i \\ K_j & -M_j^{-1}B_j - K_j\end{bmatrix}$$

The observer (4) guarantees the asymptotic convergence of the estimation $\hat{\eta}_{ij}$ to the actual relative measurement, that is

$$\lim_{t\to+\infty} \|\eta_{ij} - \hat{\eta}_{ij}\| = 0$$

Proof: We first observe that the LMI is always feasible since A_{ij} is stable and the pair (A_{ij}, C) is detectable by construction, where

$$A_{ij} = \begin{bmatrix} -M_i^{-1}B_i & 0 \\ 0 & -M_j^{-1}B_j\end{bmatrix}, \quad C = \begin{bmatrix} I_n & -I_n\end{bmatrix}$$

In particular, at least for positive gains K_1, K_2 small enough, $A_{ij} - [K_1\ K_2]^TC$ is still a Hurwitz matrix. With a slight abuse of notation, let us set $\epsilon_i = v_i - \hat{v}_{i,j}^{(i)}$, $\epsilon_j = v_j - \hat{v}_{j,i}^{(i)}$ and $\epsilon = [\epsilon_i^T\ \epsilon_j^T]^T$. Define the Lyapunov function candidate $V(\epsilon) = \epsilon^TP\epsilon$; differentiating, and observing that by construction the identity $\dot{\epsilon} = \Omega\epsilon$ holds, one gets

$$\dot{V}(\epsilon) = \epsilon^T(P\Omega + \Omega^TP)\epsilon \leq -\mu\|\epsilon\|^2$$

which, in turn, implies that both $\|\epsilon_i\|$ and $\|\epsilon_j\|$ are vanishing. Then, by a simple application of the triangle inequality, we can infer that

$$\begin{aligned}\|\eta_{ij} - \hat{\eta}_{ij}\| &= \|v_i - v_j - \hat{v}_{i,j}^{(i)} + \hat{v}_{j,i}^{(i)}\|\\ &\leq \|v_i - \hat{v}_{i,j}^{(i)}\| + \|v_j - \hat{v}_{j,i}^{(i)}\| = \|\epsilon_i\| + \|\epsilon_j\|\end{aligned}$$

is vanishing too. ∎

[1] Relative position measurements might also be considered, at the price of increasing the complexity of the scheme by using second order observers based on the model (1).

Let us now apply the previous construction to the diagnosis of an actuator fault occurring on agent i, that is

$$\dot{p}_i = -B_i M_i^{-1} p_i + u_i + \varphi_i$$

for some unknown and persistent signal φ_i. As a consequence, the error $\eta_{ij} - \hat{\eta}_{ij}$ (along with its symmetric $\eta_{ji} - \hat{\eta}_{ji}$) is no longer expected to be vanishing. In particular, defining a threshold $\varrho > 0$, we can adopt the following detection rule

$$\begin{cases} \|\eta_{ij} - \hat{\eta}_{ij}\| \le \varrho \Rightarrow \text{Healthy operational conditions} \\ \|\eta_{ij} - \hat{\eta}_{ij}\| > \varrho \Rightarrow \text{Fault on agent } i \text{ or agent } j \end{cases}$$

As usual, the threshold ϱ accounts for model uncertainties, disturbances and observer transient. Associated to the previous rule, we can define a logic variable $\lambda_{ij} \in \{0, 1\}$ as

$$\lambda_{ij} = \begin{cases} 0 & \text{if } \sigma_{ij} = 0 \\ 0 & \text{if } \{\sigma_{ij} = 1\} \wedge \{\|\eta_{ij} - \hat{\eta}_{ij}\| \le \varrho\} \\ 1 & \text{if } \{\sigma_{ij} = 1\} \wedge \{\|\eta_{ij} - \hat{\eta}_{ij}\| > \varrho\} \end{cases}$$

In light of this, each agent compiles a vector $\zeta_i \in \mathbb{R}^N$ encoding the local information on fault awareness, that is

$$\zeta_i = \begin{bmatrix} \lambda_{i1} & \lambda_{i2} & \cdots & \lambda_{i(i-1)} & 0 & \lambda_{i(i+1)} & \cdots & \lambda_{iN} \end{bmatrix}^T \quad (5)$$

Such local information is generally not sufficient to formulate an accurate fault diagnosis, but the cumulative information is instead enough. Therefore, the aim is to enable agents to share their local information in order to achieve an agreement. To achieve this goal, we propose a consensus protocol with a reset policy to inject the most recent information, and define a fault isolation cycle accordingly. Let L be the Laplacian matrix of the interconnection graph and define $\mathcal{L}_N = L \otimes I_N$. Consider the vector $z = (\pi, \delta, \tau) \in \mathbb{R}^{N^2} \times \mathbb{Z}^N \times \mathbb{R}^+$, where $\pi = [\pi_1 \cdots \pi_N]$ is a cumulative vector of fault information, δ is a vector of labels and τ is a the clock variable. Fix a reset time $\tau_\star > 0$ and define the flow and jump sets as the closed sets

$$\begin{aligned} \mathcal{C} &= \{z = (\pi, \delta, \tau) \in \mathbb{R}^{N^2} \times \mathbb{Z}^N \times \mathbb{R}^+ : \tau \in [0, \tau_\star]\} \\ \mathcal{D} &= \{z = (\pi, \delta, \tau) \in \mathbb{R}^{N^2} \times \mathbb{Z}^N \times \mathbb{R}^+ : \tau = \tau_\star\} \end{aligned} \quad (6)$$

Select initial conditions for z as

$$\begin{aligned} &\pi(0) = [\pi_i(0)^T \ \cdots \ \pi_N(0)^T]^T = [\zeta_1^T \ \cdots \ \zeta_N^T]^T \\ &\delta(0) = 0, \ \ \tau(0) = 0 \end{aligned}$$

with ζ_i given by (5) and, using the formalism of [20] and similarly to what is done in [21, Appendix A] for the decentralized estimate of the baricenter, define the consensus-like hybrid system

$$\begin{cases} \dot{\pi} = -\mathcal{L}_N \pi & \\ \dot{\delta} = 0 & z \in \mathcal{C} \\ \dot{\tau} = 1 & \\ \pi_i^+ = \zeta_i, \quad i = 1, \ldots, N & \\ \delta_i^+ = \overline{\arg\max}\, \pi_i(\tau_\star^-), \quad i = 1, \ldots, N & z \in \mathcal{D} \\ \tau^+ = 0 & \end{cases} \quad (7)$$

where the function[2] $\overline{\arg\max}$ applied on a vector $\omega = [\omega_1 \cdots \omega_r]^T \in \mathbb{R}^r$ is defined as:

$$\overline{\arg\max}\, \omega = \begin{cases} k, & \text{if } \exists k \in \{1, \ldots, r\} : \omega_k > \omega_j \ \forall j \neq k \\ 0, & \text{otherwise} \end{cases}$$

[2]The notation $\pi_i(\tau_\star^-)$ indicates the value of π_i right before the jump.

Theorem 1: Consider the nominal multi-agent system (2)-(3) and suppose that agent j_0 undergoes a fault at some time t_f. If the reset time $\tau_\star$ in (6) is fixed large enough, the hybrid system (7) guarantees that

$$\delta_1(t_{id}) = \delta_2(t_{id}) = \cdots = \delta_N(t_{id}) = j_0 \in \{0, 1, 2, \ldots, N\}$$

with $t_{id} := t_f + 2\tau_\star$, thus providing a full agreement about the fault occurrence and location.

Proof: The reset time $\tau_\star$ needs to be tuned large enough to ensure the convergence of π to its steady-state during the flow, but not too large in order to prevent delays in the diagnosis process. To this end, a simple and reliable heuristics is to select $\tau_\star$ larger than 5 times the time-constant associated to the linear map $\mathcal{L}_N$, which is essentially dictated by the second smallest eigenvalue of the Laplacian matrix L and gives an upper bound for the transient. After each cycle of length $\tau_\star$, the i-th agent retrieves the label δ_i: whenever this is different from zero, it corresponds to the index of the vehicle claimed to be faulty. Thanks to the connectivity of the graph, the consensus properties of the generalized Laplacian $\mathcal{L}_N$ and the choice of $\tau_\star$, after convergence we have $\pi_i = \pi_j$ for any i, j and, as a consequence, also $\delta_i = \delta_j$. In fact, the opinion about the label of the faulty agent becomes eventually consistent among the whole team of agents. This shows that fault isolation is successfully achieved and concludes the proof. ■

As final remark it is interesting to note that, in the case of multiple faults, the protocol (7) naturally delivers the label of the faulty vehicle with the highest degree of connectivity, whenever such dominant agent exists.

B. Decentralized Topology Update

Once the faulty vehicle j_0 has been identified, such information is available over the whole network, with $\delta_i = j_0$ for any $i = 1, \ldots, N$. To perform a reconfiguration the graph topology needs to be updated, with neighbouring agents disconnecting from the agent j_0 by setting

$$\sigma_{ij_0} = \sigma_{j_0 i} = 0 \ \forall i = 1, \ldots, N \quad (8)$$

In other words, the faulty vehicle is withdrawn from the formation. Accordingly, due to the interconnection (3), the virtual spring state w_{ij_0} ceases to be updated and stops influencing the dynamics of the agents. At the same time, in order to guarantee a good graph connectivity, new connections are established between the former neighbours of agent j_0, or among a subset of them. To this end, considering the set of neighbours of the faulty agent j_0 before the reconfiguration

$$\mathcal{N}_{j_0}^- = \{i \in \{1, \ldots, N\} : (i, j_0) \in \mathcal{E}\},$$

switches in the communication topology are triggered with

$$\sigma_{ik} = \sigma_{ki} = 1 \text{ for some } i, k \in \mathcal{N}_{j_0}^-. \quad (9)$$

Deciding the links that are worth or necessary to establish can be a tricky task. On the one hand we aim at keeping the overall graph connected but, on the other hand, we wish to keep the number of interactions between the agents as low as possible for limiting the complexity of the control algorithms. Several studies cover the related topic of graph merging, especially in

Algorithm 1 Decentralized Update of Cyclic Graph Topology

Input i, j_0
If $j_0 \in \mathcal{N}_i^-$
  % Retrieve $i' \in \mathcal{N}_{j_0}^-$, $i' \neq i$:
  If $j_0 > i$
  $i' = (j_0 \mod N) + 1$
  Else
    $i' = N - ((N - j_0 + 1) \mod N)$
  End
  % Create new links:
  Set $\sigma_{ii'} = 1$
  % Delete links with the faulty agent:
  Set $\sigma_{ij_0} = 0$
End
If $i = j_0$
  % Disconnect the faulty agent:
  Set $\sigma_{j_0 k} = 0 \ \forall k \in \mathcal{N}_{j_0}^-$
End

terms of rigidity maintenance [22]–[25]. It must be noted that having a clear picture of the lack of connectivity and of the specific links that are needed for it to be restored requires in general global information which cannot be directly obtained in a decentralized way. However, chances for decentralized strategies open up when the distribution of the edges in the graph follows a given pattern. We will address in detail the case of *cyclic graphs*, which typically arise in mobile robotics applications such as patrolling and target encircling. We provide an algorithm to be executed by each agent for the update of links, based on a simple procedure to retrieve the neighbours of the faulty agent. Due to the useful cyclic structure, the generic agent k has only two neighbours, labeled with $k+1$ and $k-1$ (mod N) consistently with their proximity; in particular, $\sigma_{ij} = 0$ for any i, j with $|i - j| > 1$. Thanks to such nice feature, each agent only needs access to the size N of the network, its own label and the label of the faulty agent.

Remark 1: It is worth noticing that, up to elementary adjustments in the selection of indices, the proposed algorithm can be used repeatedly for handling successive faults occurring in the system. Furthermore, reconfiguration of topology patterns other than the cyclic one can also be tackled by simple schemes. For example, the above algorithm can be easily adapted to the case of cyclic graphs with an additional node placed at the center and connected to all other nodes. In the case of complete graphs instead, as all possible edges are already active, it is only required to disconnect the faulty agent from the network without the need of creating new links.

C. Passive Reconfiguration

The split operation (8) does not require any additional energy to be performed and, as such, preserves the passivity of the system. However, unlike for the case of split, it must be observed that a formation join (9) may not be a passive operation in general as it requires the update of the edge state w_{ij} consistently with the relative positions of agents, and such reset may need some energy for being implemented. This happens typically when the virtual elastic potential energy verifies

$$V(\xi_{ij}(t_{\circlearrowleft}^-)) < V(q_i(t_{\circlearrowleft}^-) - q_j(t_{\circlearrowleft}^-)), \tag{10}$$

where $t_{\circlearrowleft}$ is the time instant at which the formation reconfiguration is triggered. The condition (10) is referred to as *energy obstacle*. To overcome this issue and allow for passive joins, one can resort to the implementation of energy tanks [10] which store for later use the energy naturally dissipated by the agents, i.e., the quantity D_i defined by

$$D_i = p_i^T M_i^{-T} B_i M_i^{-1} p_i.$$

The tanks are a powerful control tool, as the stored energy can be released *on demand* to implement actions in a passive way. Denoting by x_i the state of the tanks, whose initial conditions can be selected arbitrarily, their dynamics is governed by the equation $\dot{x}_i = \nu_i \frac{D_i}{x_i} - \sum_{j \in \mathcal{N}_i} h_{ij} \frac{\partial V(\xi_{ij})}{\partial \xi_{ij}}$ with $h_{ij} = \alpha_{ij} \min\{1, \delta_i\} x_i \frac{\partial V(\xi_{ij})}{\partial \xi_{ij}}^T$, where $\nu_i, \alpha_{ij} \geq 0$ are modulation coefficients, and with associated energy

$$T(x_i) = \frac{1}{2} x_i^2.$$

Accordingly, the virtual spring equations are modified to account for the energy exchange with the tanks as follows:

$$\dot{\xi}_{ij} = w_{ij} - h_{ij} x_i + h_{ji} x_j.$$

It follows from the definition that the energy can be extracted from the tank only if $\delta_i \neq 0$, that is only after fault isolation. The coefficients ν_i switch between zero and a fixed positive value $\bar{\nu} < 1$, depending on whether the tank energy T_i has reached a suitable upper bound $\bar{T}$, which is introduced to ensure boundedness of the overall system. Furthermore, a simple and useful way to define the coefficients α_{ij} is to look at the corresponding energy obstacles and set

$$\alpha_{ij} = \begin{cases} 0 & \text{if } V(\xi_{ij}) \geq V(q_i - q_j) \\ \bar{\alpha} & \text{if } V(\xi_{ij}) < V(q_i - q_j) \end{cases}$$

for some constant $\bar{\alpha} > 0$ dictating the energy extraction rate. In light of this enhanced architecture, the total energy $\mathcal{H}$, which is lower bounded by construction, is given by

$$\mathcal{H} = \sum_{i=1}^{N} (\mathcal{K}_i(p_i) + T(x_i)) + \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} V(\xi_{ij})$$

and the overall system can be naturally recast in a port-Hamiltonian setup. In particular, the interconnection of agent dynamics, tanks and virtual springs writes as

$$\begin{bmatrix} \dot{p} \\ \dot{x} \\ \dot{\xi} \end{bmatrix} = \left(\begin{bmatrix} 0 & 0 & \mathcal{I} \\ 0 & 0 & -\Gamma \\ -\mathcal{I}^T & \Gamma^T & 0 \end{bmatrix} + \begin{bmatrix} -B & 0 & 0 \\ PB & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \right) \begin{bmatrix} \frac{\partial \mathcal{H}}{\partial p} \\ \frac{\partial \mathcal{H}}{\partial x} \\ \frac{\partial \mathcal{H}}{\partial \xi} \end{bmatrix} + GF^e$$
$$v = G^T \begin{bmatrix} \frac{\partial \mathcal{H}}{\partial p} & \frac{\partial \mathcal{H}}{\partial x} & \frac{\partial \mathcal{H}}{\partial \xi} \end{bmatrix}^T \tag{11}$$

with $\mathcal{I} = \mathcal{I}_\mathcal{G} \otimes I_n$ where $\mathcal{I}_\mathcal{G}$ is the incidence matrix of the graph, $G = [(I_N \otimes I_n)^T \ 0^T \ 0^T]^T$, $B = \mathrm{diag}(B_i)$, Γ collecting the weights h_{ij}, $P = \mathrm{diag}(\frac{1}{x_i} p_i^T M_i^{-T})$ and F^e including the external inputs u_i^{ext}.

It is worth stressing that the second matrix in the right-hand side of (11) is not negative semidefinite due to the term PB, which, however, does not disrupt the overall energy balancing. To check this, one can observe that, thanks to choice $\nu_i \leq 1$, the quantity $\frac{\partial^T \mathcal{H}}{\partial x} PB \frac{\partial \mathcal{H}}{\partial p} = \sum_{i=1}^{N} \nu_i D_i$ can not be larger than the energy dissipated by the agents. This leads to the following statement.

Proposition 2: The interconnected system (11) is passive with respect to the input/output port (F^e, v).

Proof: The passivity of the system follows directly from [10, Proposition 3] as a special case. In fact, differentiating the total energy along the system solutions one gets the inequality $\dot{\mathcal{H}} = \frac{\partial^T \mathcal{H}}{\partial p}\dot{p} + \frac{\partial^T \mathcal{H}}{\partial x}\dot{x} + \frac{\partial^T \mathcal{H}}{\partial \xi}\dot{\xi} \leq v^T F^e$, which guarantees that the total energy can only increase through exchanges at the input/output port. ■

From an operational perspective, one can implement the `PassiveJoin` procedure illustrated in [10] in combination with any desired formation update algorithm, for example the one described in Section III-B for cyclic graphs. In particular, assuming that the agents i_1 and i_2 need to join at time $t_{\circlearrowleft}$ in the presence of an energy obstacle

$$\Delta V_{i_1 i_2} = V(q_i(t_{\circlearrowleft}^-) - q_j(t_{\circlearrowleft}^-)) - V(\xi_{ij}(t_{\circlearrowleft}^-)) > 0,$$

and provided that the amount of energy stored in the tanks $x_{i_1}(t_{\circlearrowleft}),\ x_{i_2}(t_{\circlearrowleft})$ is large enough, the formation reset is implemented in a passive way by performing the following simple steps:

- update the graph topology: $\sigma_{i_1 i_2} = \sigma_{i_2 i_1} = 1$
- update the edge state consistently with the current position of agents:

$$\xi_{i_1 i_2}(t_{\circlearrowleft}^+) = q_{i_1}(t_{\circlearrowleft}) - q_{i_2}(t_{\circlearrowleft})$$

- extract the needed energy from the tanks:

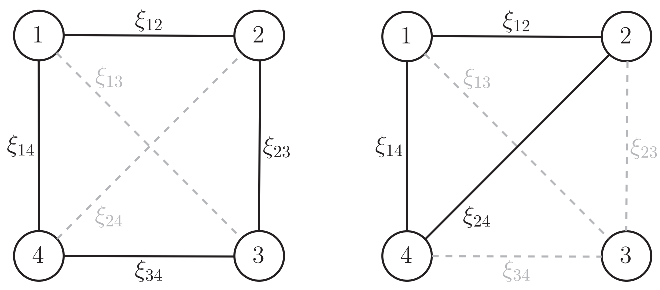
$$x_{i_1}(t_{\circlearrowleft}^+) = -\Delta V_{i_1 i_2} \frac{\sqrt{T(x_{i_1}(t_{\circlearrowleft}))}}{\sqrt{T(x_{i_1}(t_{\circlearrowleft}))} + \sqrt{T(x_{i_2}(t_{\circlearrowleft}))}}$$

$$x_{i_2}(t_{\circlearrowleft}^+) = -\Delta V_{i_1 i_2} \frac{\sqrt{T(x_{i_2}(t_{\circlearrowleft}))}}{\sqrt{T(x_{i_1}(t_{\circlearrowleft}))} + \sqrt{T(x_{i_2}(t_{\circlearrowleft}))}}$$

Whenever the energy obstacle is present but the local tank energy is not sufficient to implement the proposed action, one can postpone the join and, in the meanwhile, either run a decentralized consensus algorithm over all tanks to balance the distribution of the stored energy or increase the damping coefficients with the aim of accelerating the energy storing process (see [10, eq. (10) and Remark 6] for further details).

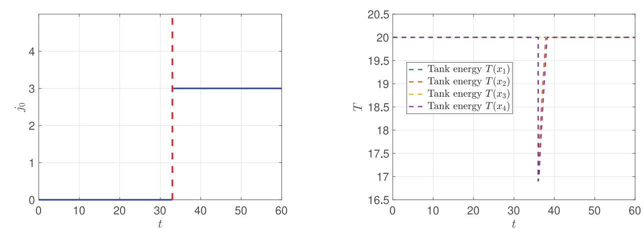
IV. Simulations

To illustrate the application of the proposed architecture, let us consider a multi-agent system composed by $N = 4$ agents connected through the cyclic graph sketched in Fig. 1(a). For the sake of simplicity the agents are assumed to belong to the space $\mathbb{R}^3$ with uniform and isotropic mass and dissipation coefficients $M = mI_{3\times 3}$ and $B = bI_{3\times 3}$, with $m = 0.16$Kg and $b = 2$Kg/s. The desired nominal formation is a square shape with a side length of $d = 5$m. This is enforced by a suitable shape of the potential functions $V(\xi_{ij})$ depending on relative positions and a consistent initialization of the edge states ξ_{ij}. Such potential functions have been selected, locally around at the minimum point corresponding to the desired relative position, as simple elastic energies with spring coefficient $k_V = 10\text{s}^{-1}$. Moreover, agent 1 is given the role of leader, whose task is to steer the whole formation along a prescribed elicoidal trajectory parametrized by



(a) (b)

Fig. 1. Graph topology: nominal formation (a), reconfigured formation (b). Dashed edges indicate inactive links.



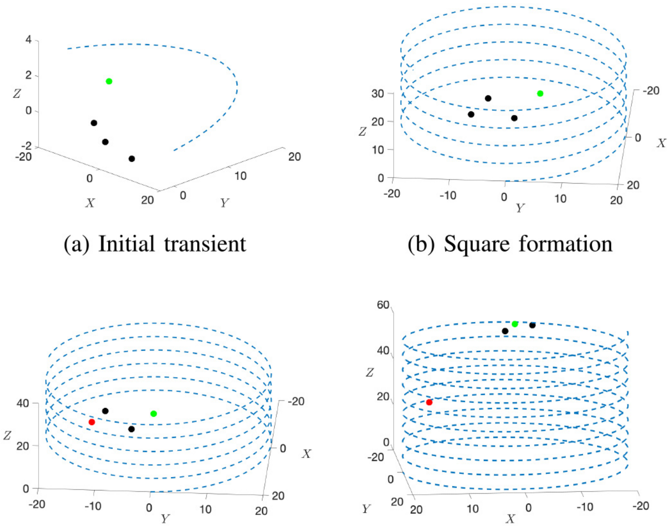
(a) Faulty agent identification (b) Tank energies

Fig. 2. Fault identification and reconfiguration process.

$$q_d(t) = (20\cos t, 20\sin t, t).$$

The trajectory tracking goal is taken care by implementing a simple PD+feedforward controller for the leader, in addition to the formation control law. A fault affecting agent 3 is injected at $t = 30$s, this causing the 70% loss of the corresponding control action. Using the diagnosis scheme based on the hybrid observers (4)-(7), the fault is detected and correctly identified at $t_{\text{id}} = 33$s, as shown in Fig. 2(a). The faulty agent being identified, the network topology is updated using the decentralized algorithm and the new graph is depicted in Fig. 1(b). Let us stress that the join procedure needed for the reconfiguration, i.e., the activation of link ξ_{24}, involves an energy obstacle $\Delta V_{24} > 0$. This is taken care of by the energy tanks x_2 and x_4, as clearly visible in the abrupt energy decreasing at $t = 36$s in Fig. 2(b). Let us also stress that the energy burden involves two out of the four energy tanks, in particular only those corresponding to the former neighbours of the faulty agent. To illustrate the efficiency of the proposed fault-tolerant formation control method, snapshots of the system evolution are provided in Fig. 3 corresponding to different phases: initial transient (a), square formation (b), reconfiguration transient (c), triangular reconfigured formation (d). The leader in highlighted in green color, while the faulty agent is marked in red color.

After reconfiguration, such agent is completely disconnected from the network and its position just remains frozen at the state it had at the moment of the topology update. This is due to the fact that, since gravity is not considered in the simulation, when follower agents are not connected they receive no inputs. The performances of the fault-tolerant formation



(a) Initial transient (b) Square formation

(c) Reconfiguration transient (d) Triangle formation

Fig. 3. Snapshots of multi-agent system evolution.

control law can be well appreciated in Figure 3(d), where a triangle formation is achieved while, simultaneously, the leader keeps following the desired reference trajectory.

V. Conclusion

The problem of fault-tolerant formation control of multi-agent systems has been tackled using passivity methods inherited from [10], where energy-tanks were used for guaranteeing passivity of the group behavior in cluttered environments where possible split/join can occur, whereas here we exploit their potential to guarantee passive reconfigurations, whatever the new formation to be achieved is. A decentralized fault diagnosis scheme is designed first, including a consensus-like hybrid algorithm to identify the faulty agent, and then a decentralized procedure to reconfigure the formation and update the graph topology is given. Embedding the multi-agent system in a port-Hamiltonian setting, the powerful tool of energy tanks is used to guarantee the overall system passivity and stability. In particular, when additional energy is needed for establishing new connections among the former neighbours of the faulty agent, this is readily extracted from the corresponding tanks by means of proper interconnections with the edge states. For cyclic graphs, we also deliver a simple and constructive algorithm for the topology update. More general graph structures may be considered in future extensions, including time-varying topologies depending on operational conditions or environmental constraints. Future works may also cover the case of nonlinear systems as well as fault-tolerant formation control with obstacle avoidance and/or rigidity maintenance.

References

[1] L. E. Parker and B. A. Emmons, "Cooperative multi-robot observation of multiple moving targets," in *Proc. Int. Conf. Robot. Automat.*, vol. 3, 1997, pp. 2082–2089.

[2] O. Simonin and O. Grunder, "A cooperative multi-robot architecture for moving a paralyzed robot," *Mechatronics*, vol. 19, no. 4, pp. 463–470, 2009.

[3] K. Klausen, C. Meissen, T. I. Fossen, M. Arcak, and T. A. Johansen, "Cooperative control for multirotors transporting an unknown suspended load under environmental disturbances," *IEEE Trans. Control Syst. Technol.*, vol. 28, no. 2, pp. 653–660, Mar. 2020.

[4] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton, NJ, USA: Princeton Univ. Press, 2010.

[5] J. R. Lawton, R. W. Beard, and B. J. Young, "A decentralized approach to formation maneuvers," *IEEE Trans. Robot. Autom.*, vol. 19, no. 6, pp. 933–941, Dec. 2003.

[6] V. Duindam and S. Stramigioli, "Port-based asymptotic curve tracking for mechanical systems," *Eur. J. Control*, vol. 10, no. 5, pp. 411–420, 2004.

[7] A. Dietrich, C. Ott, and S. Stramigioli, "Passivation of projection-based null space compliance control via energy tanks," *IEEE Robot. Autom. Lett.*, vol. 1, no. 1, pp. 184–191, Jan. 2016.

[8] F. Ferraguti, C. Secchi, and C. Fantuzzi, "A tank-based approach to impedance control with variable stiffness," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013, pp. 4948–4953.

[9] P. R. Giordano, A. Franchi, C. Secchi, and H. H. Bülthoff, "A passivity-based decentralized strategy for generalized connectivity maintenance," *Int. J. Robot. Res.*, vol. 32, no. 3, pp. 299–323, 2013.

[10] A. Franchi, C. Secchi, H. I. Son, H. H. Bulthoff, and P. R. Giordano, "Bilateral teleoperation of groups of mobile robots with time-varying topology," *IEEE Trans. Robot.*, vol. 28, no. 5, pp. 1019–1033, Oct. 2012.

[11] H. Yang, B. Jiang, and Y. Zhang, "Fault-tolerant shortest connection topology design for formation control," *Int. J. Control Autom. Syst.*, vol. 12, no. 1, pp. 29–36, 2014.

[12] X. Yu, Z. Liu, and Y. Zhang, "Fault-tolerant formation control of multiple UAVs in the presence of actuator faults," *Int. J. Robust Nonlinear Control*, vol. 26, no. 12, pp. 2668–2685, 2016.

[13] C. Deng and G.-H. Yang, "Distributed adaptive fault-tolerant control approach to cooperative output regulation for linear multi-agent systems," *Automatica*, vol. 103, pp. 62–68, May 2019.

[14] L. Saiella, A. Cristofaro, M. Ferro, and M. Vendittelli, "Fault-tolerant formation control of a team of quadrotors with a suspended payload," in *Proc. Int. Conf. Unmanned Aircraft Syst. (ICUAS)*, 2021, pp. 1–9.

[15] Y. Ren, K. Zhang, B. Jiang, W. Cheng, and Y. Ding, "Distributed fault-tolerant time-varying formation control of heterogeneous multi-agent systems," *Int. J. Robust Nonlinear Control*, vol. 32, no. 5, pp. 2864–2882, 2022.

[16] C. Deng, X.-Z. Jin, W.-W. Che, and H. Wang, "Learning-based distributed resilient fault-tolerant control method for heterogeneous mass under unknown leader dynamic," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Apr. 16, 2021, doi: 10.1109/TNNLS.2021.3070869.

[17] A. Cristofaro and T. A. Johansen, "Fault tolerant control allocation using unknown input observers," *Automatica*, vol. 50, no. 7, pp. 1891–1897, 2014.

[18] S. Bouabdallah and R. Siegwart, "Full control of a quadrotor," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2007, pp. 153–158.

[19] G. Oriolo, A. De Luca, and M. Vendittelli, "WMR control via dynamic feedback linearization: design, implementation, and experimental validation," *IEEE Trans. Control Syst. Technol.*, vol. 10, no. 6, pp. 835–852, Nov. 2002.

[20] R. Goebel, R. G. Sanfelice, and A. R. Teel, *Hybrid Dynamical Systems*. Princeton, NJ, USA: Princeton Univ. Press, 2012.

[21] M. Aggravi, C. Pacchierotti, and P. R. Giordano, "Connectivity-maintenance teleoperation of a UAV fleet with wearable haptic feedback," *IEEE Trans. Autom. Sci. Eng.*, vol. 18, no. 3, pp. 1243–1262, Jul. 2021.

[22] T. Eren, B. Anderson, W. Whiteley, A. Morse, and P. Belhumeur, "Merging globally rigid formations of mobile autonomous agents," in *Proc. Int. Joint Conf. Autonom. Agents Multiagent Syst. (AAMAS)*, 2004, pp. 1260–1261.

[23] C. Yu, B. Fidan, and B. D. Anderson, "Principles to control autonomous formation merging," in *Proc. Amer. Control Conf.*, 2006, p. 7.

[24] J. Fang and A. S. Morse, "Merging globally rigid graphs and sensor network localization," in *Proc. IEEE Conf. Decis. Control*, 2009, pp. 1074–1079.

[25] T. Han, Z. Lin, and M. Fu, "Three-dimensional formation merging control under directed and switching topologies," *Automatica*, vol. 58, pp. 99–105, Aug. 2015.