

Distributed Adaptive Fault-Tolerant Leader-Following Formation Control of Nonlinear Uncertain Second-Order Multi-Agent Systems

Mohsen Khalili[†], Xiaodong Zhang[†], Yongcan Cao[‡], Marios M. Polycarpou*,
and Thomas Parisini[§]

[†] Department of Electrical Engineering,
Wright State University, Dayton, OH 45435, USA

[‡] Department of Electrical and Computer Engineering,
University of Texas, San Antonio, TX 78249, USA

* KIOS Research Center, Department of Electrical and Computer Engineering,
University of Cyprus, Nicosia 1678, Cyprus

[§] Dept. of Electrical and Electronic Engineering, Imperial College London, UK,
and Dept. of Engineering and Architecture, University of Trieste, Italy

Submitted to <i>International Journal of Robust and Nonlinear Control</i> April 2017, Revised: November 2017 Revised: March 2018

Abstract

This paper presents a distributed integrated fault diagnosis and accommodation scheme for leader-following formation control of a class of nonlinear uncertain second-order multi-agent systems. The fault model under consideration includes both process and actuator faults, which may evolve abruptly or incipiently. The time-varying leader communicates with a small subset of follower agents, and each follower agent communicates to its directly connected neighbors through a bidirectional network with possibly asymmetric weights. A local fault diagnosis and accommodation component is designed for each agent in the distributed system, which consists of a fault detection and isolation module and a reconfigurable controller module comprised of a baseline controller and two adaptive fault-tolerant controllers, activated after fault detection and after fault isolation, respectively. By using appropriately designed Lyapunov functions, the closed-loop stability and asymptotic convergence properties of the leader-follower formation are rigorously established under different modes of the fault-tolerant control system.

Keywords: Fault-Tolerant Control, Formation Control, Adaptive Control, Multi-Agent Systems, Nonlinear Uncertain Systems.

1 Introduction

1.1 State of the Art

Many modern engineering systems are modeled as multi-agent systems (MASs), comprised of various distributed and interconnected autonomous agents/subsystems. Examples of such systems include cooperative unmanned aerial vehicles, air traffic control systems, and connected autonomous ground vehicles, etc. In recent years, cooperative control using distributed formation control algorithms has received significant attention (see, e.g., [1–3] and references therein). Two types of control problem have been considered, i.e., the cooperative regulator problem (also known as leaderless consensus) and the cooperative tracking problem. For the regulator problem, all the agents/nodes are driven to the consensus equilibrium that is dependent on initial conditions of the agents. For the tracking problem, all follower agents are synchronized to track a leader agent acting as a command generator, despite the leader’s command is only received by a small portion of followers. Adaptive cooperative control methods for achieving leader-following tracking in the presence of various uncertainties have also been proposed by assuming the absence of faults [4–6]. Since such systems need to operate reliably at all times, despite the possible occurrence of faulty behaviors in some agents, the development of fault-tolerant control (FTC) schemes is a crucial requirement in achieving dependable and safe operations.

Considerable research efforts have focused on the development of FTC methods for actuator faults in multi-agent systems. Fault-tolerant formation control of unmanned aerial vehicles was investigated in [7], where the reference of the leader is adjusted to yield feasible references in response to actuator faults in followers. Fault-tolerant cooperative tracking algorithms have been developed for accommodating actuator faults in multi-agent systems with linear and Lipschitz nonlinear dynamics [8–11]. Actuator faults in agents with more general nonlinear dynamics are considered in [12, 13], where adaptive approximator (e.g., neural networks and fuzzy systems) are used to approximate the agents’ uncertain nonlinear dynamics. A distributed adaptive scheme is developed in [14] to compensate for the effects of time-varying actuator faults in multi-agent systems with delayed state perturbations. A robust adaptive cooperative tracking control scheme is developed in [15] to deal with time-delays and dead-zone nonlinearities. *The results in [8–15] are all based on the critical assumption that the Laplacian matrix of the communication graph is symmetric.* However, the distributed FTC problem for leader-follower multi-agent systems naturally require the consideration of an asymmetric Laplacian matrix, which is significantly more challenging. Limited results are available on leader-following FTC design for agents connected by graphs with asymmetric Laplacian matrix. Under directed communication graphs, interesting FTC algorithms for actuator faults to achieve leader-following tracking [16] and containment control [17] have been presented. It is assumed that the derivatives of the leaders’ outputs are available to their neighboring followers in [17]. *However, in all the aforementioned results, the FTC problem for process faults was not considered, which is crucial to the safe operations of MASs. Moreover, cooperative uniformly ultimately bounded (CUUB) results were obtained for MASs under an asymmetric Laplacian matrix [16, 17].*

1.2 Objectives and Contributions

The objective of this paper is to develop a fault-tolerant leader-follower formation control scheme for a class of second-order nonlinear uncertain MASs, which are interconnected via a bidirectional communication topology with possibly *asymmetric* weights and may be subject to both process and actuator faults. Compared with the existing results, the contributions of the paper are summarized as follows.

- A fault-tolerant formation control scheme is derived for MASs under a bidirectional communication topology with possibly *asymmetric* weights. It is worth noting that the asymmetric weights of the

graph under consideration don't assume the critical detail-balanced condition considered in the literature [18, 19], which significantly increases the complexity of stability analysis. For instance, the methods for stability analysis presented in [8–15], which utilize the symmetric property of the Laplacian matrix to solve the leader-follower formation problem for undirected symmetric graphs, are no longer applicable.

- Unlike the CUUB results for asymmetric graphs shown in the literature (e.g., the sliding mode FTC method [16] and the containment control scheme [17]), the adaptive FTC method proposed in this paper guarantees that the formation tracking error asymptotically converges to zero. Note that in the leader-following topology considered in this paper, the time-varying leader only communicates to a small subset of follower agents, and each follower agent exchanges measurement information only with its neighbors through an asymmetric interconnection topology. These constraints make it more difficult to accomplish the *asymptotic* convergence property of leader-following formation error in the presence of faults and modeling uncertainty.
- The agent fault model under consideration includes both process and actuator faults, while process faults were not considered in all the aforementioned papers. Moreover, both incipient faults (i.e., slowly developing faults) and abrupt faults are considered in this paper.
- The proposed FTC architecture can potentially achieve improved tracking performance by exploiting online fault detection and isolation information, since the objective of FTC is to compensate for the effect of such faults. In the papers described above, only the FTC scheme in [16] utilizes the integrated fault diagnosis and accommodation architecture, where only CUUB results were established and the problem of FTC design after fault isolation was not addressed.

An integrated fault diagnosis and accommodation component is designed for each agent by utilizing measurements information exchanged between neighboring agents. Each local FTC component consists of two main modules: 1) a decentralized fault diagnosis module consisting of a fault detection estimator and a bank of fault isolation estimators; 2) a distributed controller (fault accommodation) module consisting of a baseline controller and two adaptive fault-tolerant controllers employed after fault detection and after fault isolation, respectively. After fault detection, a distributed adaptive-approximation-based controller is designed for achieving fault-tolerance of the agents before fault isolation because of the ability of the “adaptive approximator” to learn the unknown fault functions [20]. Another adaptive fault-tolerant controller with simplified structure is used to improve the control performance if the faults are isolated. Under certain assumptions, by using suitably chosen Lyapunov functions, the closed-loop stability and asymptotic leader-follower formation tracking properties of the adaptive fault-tolerant controllers are rigorously established. This paper significantly extends the results in the previous papers by authors [21, 22]. Specifically, a class of *first-order* MASs was considered in [21], and a second-order model with *symmetric* graphs was considered in [22], while this paper considers second-order agents under graphs with *asymmetric* weights. Moreover, the fault model considered in this paper includes both abrupt and *incipient* faults, while those previous papers only considered abrupt faults.

1.3 Organization

The rest of the paper is organized as follows. The problem formulation for fault-tolerant leader-follower formation control of multi-agent systems is described in Section 2. The design and analysis of the fault-tolerant control scheme employed between fault detection and isolation is presented in Section 3. The FTC design after fault isolation is described in Section 4. In Section 5, a simulation example

is used to illustrate the effectiveness of the FTC method. Finally, Section 6 provides some concluding remarks.

2 Problem Formulation

2.1 Graph Theory Notations

To facilitate the problem formulation, some basic standard notations for graphs are recalled for the reader's convenience. A directed graph \mathcal{G} is a pair $(\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{v_1, \dots, v_m\}$ is a set of nodes, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is a set of edges, and m is the number of nodes. An edge is an ordered pair of distinct nodes (v_j, v_i) , meaning that the i th node can receive information from the j th node. For an edge (v_j, v_i) , node v_j is called the parent node, node v_i is the child node, and v_j is a neighbor of v_i . A sequence of distinct edges in the directed graph \mathcal{G} creates a directed path between two distinct nodes. A graph contains a directed spanning tree if there exists a node called the root, which has no parent node, such that the node has directed paths to all other nodes in the graph.

The set of neighbors of node v_i is denoted by $N_i = \{j : (v_j, v_i) \in \mathcal{E}\}$. The weighted adjacency matrix $C = [c_{ij}] \in \mathbb{R}^{m \times m}$ associated with the directed graph \mathcal{G} is defined by $c_{ii} = 0$, $c_{ij} > 0$ if $(v_j, v_i) \in \mathcal{E}$, and $c_{ij} = 0$ otherwise. An interaction graph \mathcal{G} is said to be fixed, if each node has a fixed neighbor set and c_{ij} is fixed. For undirected graphs, $c_{ij} = c_{ji}$ and for balanced graphs $\sum_{j=1}^m c_{ij} = \sum_{j=1}^m c_{ji}$. The Laplacian matrix $L = [l_{ij}] \in \mathbb{R}^{m \times m}$ is defined as $l_{ii} = \sum_{j \in N_i} c_{ij}$ and $l_{ij} = -c_{ij}$, $i \neq j$. Both C and L are symmetric only for undirected graphs. The sum of the elements on each row of the Laplacian matrix L is zero, therefore 0 is an eigenvalue of L . The graph \mathcal{G} has a spanning tree if and only if the Laplacian matrix of the graph \mathcal{G} has a simple zero eigenvalue. More detailed description of graph theory can be found in [2].

2.2 Distributed Multi-Agent System Model

Consider a set of M interconnected agents where the second-order dynamics of the i th agent, for $i = 1, \dots, M$, is described by:

$$\begin{aligned} \dot{p}_i &= v_i \\ \dot{v}_i &= \phi_i(x_i) + u_i + \eta_i(x_i, t) + \beta_i(t - T_i)f_i(x_i) + \theta_i(t)u_i, \end{aligned} \quad (1)$$

where $x_i = \begin{bmatrix} p_i \\ v_i \end{bmatrix} \in \mathbb{R}^2$ and $u_i \in \mathbb{R}$ are the state vector and the input of the i th agent, respectively. Additionally, $\phi_i : \mathbb{R}^2 \mapsto \mathbb{R}$, $\eta_i : \mathbb{R}^2 \times \mathbb{R}^+ \mapsto \mathbb{R}$, $f_i : \mathbb{R}^2 \mapsto \mathbb{R}$ are smooth vector fields.

The model given by

$$\dot{x}_i = \begin{bmatrix} 0 \\ \phi_i(x_i) \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_i \quad (2)$$

represents the *known nominal* dynamics of the i th agent with ϕ_i being the known nonlinearity. The "healthy" (in the absence of faults) system is described by

$$\dot{x}_i = \begin{bmatrix} 0 \\ \phi_i(x_i) \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} (u_i + \eta_i(x_i, t)). \quad (3)$$

The difference between the nominal model (2) and the actual (healthy) system dynamics (3) is due to $\eta_i(x_i, t)$, characterizing the modeling uncertainty in the state dynamics of the healthy agent, including disturbances and modeling errors.

The term $\beta_i(t - T_i)f_i(x_i)$ in (1) denotes the unknown change in the dynamics of i th agent due to the occurrence of a process fault, where $\beta_i(t - T_i)$ represents the time profile of the process fault which occurs at some unknown time T_i . Additionally, for isolation purposes, we assume that there are $r_i - 1$ types

of partially known fault functions associated with the i th agent; Specifically, each process fault function f_i^w , $w = 1, \dots, r_i - 1$, is described by

$$f_i^w(x_i) \triangleq (\theta_i^w)^T g_i^w(x_i), \quad (4)$$

where θ_i^w , for $i = 1, \dots, M$, is an unknown parameter vector assumed to belong to a known compact set Θ_i^w (i.e., $\theta_i^w \in \Theta_i^w \subseteq \mathbb{R}^{b_i^w}$), and $g_i^w : \mathbb{R}^2 \mapsto \mathbb{R}^{b_i^w}$ is a known smooth vector field. As described in [23], the process fault model described by (4) characterizes a general class of nonlinear process faults where the vector field g_i^w represents the functional structure of the w th process fault, and the unknown parameter vector θ_i^w characterizes the fault magnitude. For instance, for each leakage fault in the well-known three-tank benchmark system [23], the fault functional structure (i.e., g_i^w) can be represented as a nonlinear function of the liquid levels, and the unknown leakage size is the fault magnitude θ_i^w . In this paper, the time profile function $\beta_i(\cdot)$ is modeled by a time-varying function that is zero before fault occurrence (i.e., $t < T_i$), and satisfies $0 < \beta_i \leq 1$ for $t \geq T_i$. For instance, the time profile of abrupt faults can be modeled as a step function, and the time profile of incipient faults can be modeled as a drift-type fault or an exponential term with an unknown fault-evolution rate. Therefore, both incipient and abrupt faults are considered in this paper.

Furthermore, the term $\theta_i(t)u_i$ in (1) represents the changes in the dynamics of i th agent due to the occurrence of a time-varying actuator fault. Specifically, $\theta_i(t) \in [\theta_i^*, 0]$ is the unknown and time-varying fault parameter characterizing a partial loss of effectiveness fault in actuator u_i , for $i = 1, \dots, M$, where $\theta_i^* \in (-1, 0)$ is an *unknown* lower constant bound. The case of $\theta_i = 0$ corresponds to a healthy actuator, whereas a *negative* value of $\theta_i(t)$ implies that the actuator is partially faulty. Note that it is required that $\theta_i^* > -1$ to ensure the controllability of the distributed agents.

Based on the process faults described by (4) and actuator fault model described above, the fault class associated with agent i under consideration is given by

$$\mathcal{F}_i \triangleq \{f_i^1(x_i), \dots, f_i^{(r_i-1)}(x_i), \theta_i u_i\}. \quad (5)$$

The objective of this paper is to develop a distributed fault-tolerant leader-following formation control scheme for the class of distributed multi-agent systems described by (1). In this paper, we denote \mathcal{G} as the fixed graph of the overall MAS. The leader only communicates to a subset of followers, and each follower only communicates to its neighboring agents. Without loss of generality, let the leader be identified as agent number 0 with state $x_0(t) = [p_0(t), v_0(t)]^T$, where p_0 and v_0 denote the position and velocity of the leader respectively; therefore $\dot{p}_0 = v_0$. Then, the stability and convergence analysis is based on the following assumptions:

Assumption 1. The modeling uncertainty, represented by $\eta_i(x_i, t)$ in (1), has a known upper bound, i.e., $\forall x_i \in \mathbb{R}^2$,

$$|\eta_i(x_i, t)| \leq \bar{\eta}_i(x_i, t), \quad (6)$$

where the bounding function $\bar{\eta}_i$ is known and uniformly bounded with respect to (x_i, t) .

Assumption 2. The derivative of the leader's second state (i.e., \dot{v}_0) is bounded by an unknown constant κ , i.e.,

$$|\dot{v}_0| \leq \kappa. \quad (7)$$

The modeling uncertainty considered Assumption 1 is assumed to be *unstructured* but bounded by certain known functions. The knowledge on the bounding function $\bar{\eta}_i$ is mainly needed by the fault diagnosis procedure to distinguish between the effects of faults and modeling uncertainty. In the fault-diagnosis literature, efforts to enhance the robustness of FDI schemes with respect to modeling uncertainty

can be made either at the residual generation stage by using decoupling techniques or at the decision making stage by using adaptive thresholds. In the first approach, the modeling uncertainty is often assumed to be *structured*, which allows the use of linear and nonlinear state transformations to decouple faults from modeling uncertainty. If the modeling uncertainty is *unstructured*, decoupling faults from modeling uncertainty is not possible. Then, a known bound on the modeling uncertainty is needed to derive adaptive thresholds for distinguishing between the effects of faults and modeling uncertainty [23,24]. Note that the bounding function $\bar{\eta}_i$ can possibly be obtained by making use of certain limited knowledge on the modeling uncertainty under the worst-case scenario (see, for instance, an aircraft engine fault diagnosis application considered in [25]).

Assumption 2 is needed to achieve formation control for a time-varying leader. Note that it only requires an *unknown* bound on the derivative of the leader’s state. An adaptive bounding control design will be developed to estimate the unknown bound κ .

2.3 Distributed Fault-Tolerant Control Architecture

The research objective in this paper is to develop distributed robust FTC algorithms such that each agent’s state converges to a desired formation with the time-varying leader even in the presence of faults and modeling uncertainty. Specifically, the proposed distributed FTC algorithm is designed to guarantee the velocity of each follower converges to that of the leader, and the positions between neighboring agents i and j converge to a specified desired distance, i.e., $p_i(t) - p_j(t) \rightarrow \bar{p}_{ij}$ and $v_i(t) \rightarrow v_0(t)$, where \bar{p}_{ij} is the desired constant relative position between the agents i and j , for $i = 1, \dots, M, j \in N_i$.

An example of a distributed FTC architecture with one leader and five follower agents is shown in Figure 1. It is noted that the leader (i.e., node 0) communicates only with a subset of followers (only agent 2 in this example), while each follower communicates to its directly connected neighbors through a bidirectional network with possibly asymmetric weights. It is assumed that the leader has a directed path to all followers. As described in [26], synchronization to the leader node cannot be achieved if there are some follower nodes that are either isolated or do not receive information from any other nodes (as the leader). As shown in Figure 1, each local control module consists of a fault detection and isolation (FDI) component and a FTC component. The decentralized FDI component is used to detect the occurrence of a fault and determine the particular fault type. Based on the local diagnostic information, each agent reconfigures its fault-tolerant controller, which is designed by using local measurements and measurements received from neighboring agents.

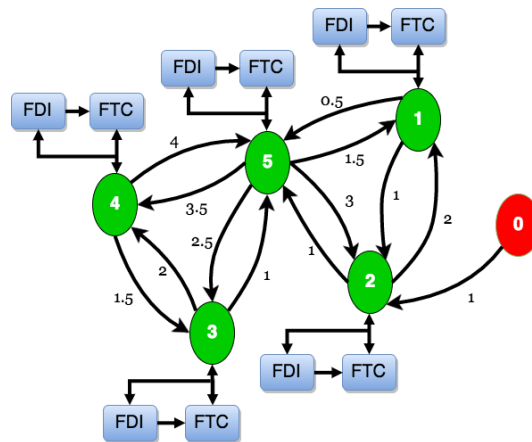


Figure 1: An example of distributed FTC architecture

Based on the diagnostic information, the distributed FTC scheme is designed according to the following qualitative objectives:

1. In a fault free mode of operation, a baseline controller guarantees that the state of i th agent $x_i(t)$ should track the predefined formation with a time-varying leader $x_0(t)$, even in the possible presence of plant modeling uncertainty.
2. If a fault is detected by the FDI scheme, the baseline controller is reconfigured to compensate for the effect of the fault whose fault type is yet unknown. The fault-tolerant controller is designed in such a way that some control performance can be recovered by exploiting the information that a fault has occurred, guaranteeing the boundedness of system signals and some leader-following formation performance, even in the presence of the fault.
3. If the fault type is isolated by the isolation scheme, then the controller is reconfigured again. The second fault-tolerant controller is designed using the information on the type of fault that has occurred so as to further improve the control performance.

Because the baseline controller is a special case of the adaptive fault-tolerant controllers, this paper focuses on the design of the two adaptive FTC schemes utilizing the fault detection and isolation information, respectively.

Remark 1: Compared with passive FTC schemes, the proposed FTC architecture allows to maximize the use of a simpler nominal controller designed for achieving optimal control performance of the “healthy” plant (i.e., in the absence of faults), instead of trading performance for robustness to certain faults. Additionally, the design and analysis of adaptive FTC schemes often require certain known knowledge of the fault functional structure (i.e., \bar{g}_i^s), which may not be sufficient to accommodate new or unexpected faults that do not belong to the pre-defined fault class \mathcal{F}_i given by (5). In contrast, the proposed FTC architecture takes into account this important practical issue. Specifically, in this case, the new or unexpected fault that has occurred will be detected (if the fault detectability condition is satisfied [?]), but not isolable, because it does not belong to the pre-defined fault class \mathcal{F}_i . Then, the neural-network-based adaptive fault-tolerant controller (22) activated after fault detection maintains the stability of the system, and the adaptive fault-tolerant controller (57) or (63), designed based on the functional structure of isolated faults, does not become activated. Therefore, it is essential to use the neural-network-based adaptive fault-tolerant controller in the proposed FTC architecture.

Remark 2: Various interesting FTC schemes have been developed for centralized systems [27–30]. A centralized adaptive FTC scheme for a class of nonlinear uncertain systems was presented in the previous paper by authors [27], where the FTC design is based on a global mathematical model and is required to have real-time access to all sensor measurements in the overall system. Due to limitations of computational resources and communication overhead, this centralized method is not applicable to leader-following tracking control of large-scale multi-agent systems. Therefore, the distributed FTC problem considered in this paper addresses some key new challenges compared to the centralized problem in [27]. Specifically, while each local FTC component is designed using only measurements of the local agent and its neighbors, the distributed leader-following FTC scheme guarantees each agent tracks the leader’s state, which is accessible only to a limited number of agents, even in the presence of faults and modeling uncertainty. An appropriate Lyapunov function and distributed controller gains must be skillfully chosen for deriving the asymptotic fault-tolerant leader-following consensus property of the multi-agent system under a bidirectional communication topology with general asymmetric weights.

3 Fault Accommodation: after Fault Detection

3.1 Decentralized Fault Detection

Under normal operating conditions, each local fault detection estimator (FDE) monitors the corresponding agent to detect the occurrence of any faults. The decentralized fault detection method can be easily designed using the results in [27]. Based on the agent model described by (1), the FDE for each agent is chosen as:

$$\dot{\hat{x}}_i = \Lambda_i^0(x_i - \hat{x}_i) + \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} (\phi_i(x_i) + u_i), \quad (8)$$

where $\hat{x}_i \in \mathbb{R}^2$ denotes the estimated local state, $\Lambda_i^0 = \begin{bmatrix} \lambda_{p_i}^0 & 0 \\ 0 & \lambda_{v_i}^0 \end{bmatrix} \in \mathbb{R}^{2 \times 2}$ is a positive definite estimator gain matrix.

For each local FDE, let $\varepsilon_i \triangleq x_i - \hat{x}_i = [\varepsilon_{p_i}, \varepsilon_{v_i}]^T$ denote the state estimation error of the i th agent. Then, before fault occurrence (i.e., for $0 \leq t < T_i$), a bounding function on each component of the state estimation error ε_i can be derived. Specifically, it can be shown that $|\varepsilon_{v_i}| \leq \nu_i(t)$, where

$$\nu_i(t) \triangleq \int_0^t e^{-\lambda_{v_i}^0(t-\tau)} \bar{\eta}_i(x_i, \tau) d\tau + \bar{x}_i e^{-\lambda_{v_i}^0 t}, \quad (9)$$

and \bar{x}_i is a bound on the initial state estimation error (i.e., $|\varepsilon_{v_i}(0)| \leq \bar{x}_i$). Even if the initial state estimation error is conservative, this will not affect the threshold at steady state since it is multiplied by a decaying term, as shown in the adaptive threshold (9). Note that the integral term in the above thresholds can be easily implemented as the output of a first-order linear filter $H(s) = 1/(s + \lambda_{v_i}^0)$ with the input given by $\bar{\eta}_i(x_i, t)$.

Thus, the decision on the occurrence of a fault (detection) in the i th agent is made when the absolute value of the estimation error (i.e., ε_{v_i}) generated by the local FDE exceeds its corresponding threshold (i.e., $|\varepsilon_{v_i}(t)| > \nu_i(t)$), where $\nu_i(t)$ is given by (9). The fault detection time T_d is defined as the first time instant such that $|\varepsilon_{v_i}(T_d)| > \nu_i(T_d)$, for some $T_d \geq T_i$, that is,

$$T_d \triangleq \inf \bigcup_{i=1}^M \{t \geq 0 : |\varepsilon_{v_i}(t)| > \nu_i(t)\}.$$

3.2 Neural-Network-Based Adaptive Fault-Tolerant Controller

After the fault is detected at time $t = T_d$ by agent i , the nominal controller is reconfigured to ensure the system stability and some tracking performances after fault detection. In the following, we describe the design of the fault-tolerant controller using distributed adaptive tracking techniques.

Before the fault is isolated, no information about the fault type is available. Adaptive approximators such as neural-network models can be used to estimate the unknown process fault function $f_i(x_i)$ described by (4). The term ‘‘adaptive approximator’’ [20] is used to represent nonlinear multivariable approximation models with adjustable parameters, such as neural networks, fuzzy logic networks, polynomials, spline functions, etc. Specifically, we consider linearly parametrized network (e.g., radial-basis-function networks with fixed centers and variances) described as follows:

$$\hat{f}_i(x_i, \hat{\vartheta}_i) = \hat{\vartheta}_i^T \varphi_i(x_i), \quad (10)$$

where $\varphi_i(\cdot)$ represents the fixed basis functions, and $\hat{\vartheta}_i$ is the adjustable weights of the nonlinear approximator.

Based on (1) and (10), for $t \geq T_d$, the system dynamics described by (1) can be rewritten as

$$\dot{x}_i = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} [\phi_i(x_i) + (1 + \theta_i(t))u_i + \eta_i(x_i, t) + \hat{f}_i(x_i, \vartheta_i) + \beta_i \delta_i(x_i) + (\beta_i - 1)\hat{f}_i(x_i, \vartheta_i)], \quad (11)$$

where $\delta_i \triangleq f_i(x_i) - \hat{f}_i(x_i, \vartheta_i)$ is the network approximation error for the i th agent, and ϑ_i is the *unknown* optimal weight vector given by

$$\vartheta_i \triangleq \arg \inf_{\hat{\vartheta}_i \in \Theta_i} \left\{ \sup_{x_i \in \mathcal{X}_i} |f_i(x_i) - \hat{f}_i(x_i, \hat{\vartheta}_i)| \right\},$$

where $\mathcal{X}_i \subseteq \mathbb{R}^2$ denotes the set to which the variable x_i belongs for all possible modes of behavior of the controlled system. For each network, we make the following assumption on the network residual approximation error:

Assumption 3. For each $i = 1, \dots, M$, the residual approximation error satisfies

$$|\delta_i(x_i)| \leq \alpha_i \bar{\delta}_i(x_i), \quad \forall x_i \in \mathbb{R}^2 \quad (12)$$

where $\bar{\delta}_i$ is a known positive bounding function, and α_i is an unknown constant.

Remark 3: In [5, 6, 12, 16, 17, 31], a constant bound was assumed for the residual approximation error, while a more general bound in the form of an unknown constant multiplied by a function of agent state is considered in this paper, therefore increasing the complexity of stability analysis.

We let $\alpha_{m_i}^0$ represent an unknown constant defined as

$$\alpha_{m_i}^0 \triangleq \sup_{t \geq T_d} \max \left\{ |\beta_i(t - T_i) \alpha_i|, |[\beta_i(t - T_i) - 1] \vartheta_i| \right\}, \quad (13)$$

where T_i is the process fault occurrence time. Note that the fault time profile $\beta_i(t - T_i)$ satisfies $0 \leq \beta_i \leq 1$. Then, the finite constant $\alpha_{m_i}^0$ defined by (13) always exists.

The following Lemmas are needed for the design and analysis of the distributed fault-tolerant formation control algorithm:

Lemma 1. Suppose $\Psi \in \mathbb{R}^{(M+1) \times (M+1)}$ is the Laplacian matrix of intercommunication graph as if the communication between the leader and followers is bidirectional. The matrix

$$\Omega \triangleq \chi \Psi + \Psi^T \chi \quad (14)$$

is positive semidefinite with $\mathbf{1}_{M+1}$ as its right eigenvector, where $\chi = \text{diag}\{\chi_0, \chi_1, \chi_2, \dots, \chi_M\}$ is a diagonal matrix consists of the elements of the left eigenvector of Ψ associated with the eigenvalue zero, i.e., $\Psi^T \bar{\chi} = 0$, $\bar{\chi} = [\chi_0, \chi_1, \chi_2, \dots, \chi_M]^T$, and $\mathbf{1}_{M+1}$ is a $(M + 1) \times 1$ column vector of ones.

Proof: Consider the communication between the leader and followers to be bidirectional (i.e., by adding edges with positive gains \bar{k}_i connecting agent i to the leader, if agent i receives information from the leader). Specifically, \bar{k}_i is a design constant satisfying $\bar{k}_i > 0$ if the leader directly communicates to follower agent i in the graph \mathcal{G} , and $\bar{k}_i = 0$ otherwise. Therefore, because the leader has a path to all the followers, the augmented graph topology of all the agents including the bidirectional leader becomes strongly connected. Based on Lemma 6 and Lemma 9 in [32], the symmetric matrix Ω defined

in (14) is positive semidefinite and has a simple zero eigenvalue with $\mathbf{1}_{M+1}$ as its right eigenvector. \square

Remark 4: It is worth noting that the Laplacian matrix Ψ for a bidirectional leader is only considered for the purpose of controller performance analysis, while the underlying communication topology has a directed leader since the leader is only sending the data and does not receive any data from other agents.

Lemma 2. *Suppose the Laplacian matrix Ψ and the diagonal matrix χ , defined in Lemma 1, have the following decomposition:*

$$\Psi = \begin{bmatrix} \Psi_0 & \Psi_{12} \\ \Psi_{21} & \hat{\Psi} \end{bmatrix}, \quad \chi = \begin{bmatrix} \chi_0 & \mathbf{0} \\ \mathbf{0}_M & \hat{\chi} \end{bmatrix}, \quad (15)$$

where $\Psi_0 \in \mathfrak{R}$, $\Psi_{12} \in \mathfrak{R}^{1 \times M}$, $\Psi_{21} \in \mathfrak{R}^{M \times 1}$, $\hat{\Psi} \in \mathfrak{R}^{M \times M}$, $\chi_0 \in \mathfrak{R}$, $\hat{\chi} \in \mathfrak{R}^{M \times M}$, and $\mathbf{0}_M$ is a $M \times 1$ column vector of zeros. The matrix

$$\bar{\Psi} \triangleq \hat{\chi} \hat{\Psi} + \hat{\Psi}^T \hat{\chi} \quad (16)$$

is positive definite.

Proof: From (14) and (15), we have

$$\Omega = \begin{bmatrix} 2\chi_0\Psi_0 & \chi_0\Psi_{12} + \Psi_{21}^T\hat{\chi} \\ \hat{\chi}\Psi_{21} + \Psi_{12}^T\chi_0 & \hat{\chi}\hat{\Psi} + \hat{\Psi}^T\hat{\chi} \end{bmatrix}. \quad (17)$$

Based on Lemma 9 in [32], Ω can be considered as a Laplacian matrix of an augmented undirected graph $\bar{\mathcal{G}}$, which has the same node set as the graph corresponding to Ψ , but with weights $\chi_i k_{ij} + \chi_j k_{ji}$ connecting agent i and agent j , for $j \in N_i$ and $j \neq 0$, as well as weights $\chi_i k_{i0} + \chi_0 k_i$ connecting agent i and the leader (i.e., agent 0), where k_{ij} and k_{ji} are positive constants denoting the weights on the intercommunication graph \mathcal{G} . Consider $\hat{\Omega} \in \mathfrak{R}^{(M+1) \times (M+1)}$ to be a matrix whose first row is a $1 \times (M+1)$ row vector of zeros and has the same rows as Ω for the $(i+1)$ th row, $i = 1, \dots, M$. Then, we can decompose the matrix $\hat{\Omega}$ as follows:

$$\hat{\Omega} = \begin{bmatrix} \mathbf{0}_{1 \times 1} & \mathbf{0}_{1 \times M} \\ \hat{\chi}\Psi_{21} + \Psi_{12}^T\chi_0 & \hat{\chi}\hat{\Psi} + \hat{\Psi}^T\hat{\chi} \end{bmatrix}. \quad (18)$$

From the specific structures of $\hat{\Omega}$ and Ω given in (18) and (17), we can see that the topology graph corresponding to $\hat{\Omega}$ for the $M+1$ agents (with the leader being agent 0) has the same nodes as $\bar{\mathcal{G}}$ and a spanning tree with the leader as its root. Thus, the matrix $\hat{\Omega}$ has M positive eigenvalues and has a simple zero eigenvalue with $\mathbf{1}_{M+1}$ as its right eigenvector [2]. Now, by using the specific structure of $\hat{\Omega}$ given in (18), we can conclude $\hat{\chi}\hat{\Psi} + \hat{\Psi}^T\hat{\chi}$ is a positive definite matrix. \square

Lemma 3. *Consider the positive definite square matrix $\bar{\Psi} \in \mathfrak{R}^{M \times M}$ defined in (16). Define*

$$A = \begin{bmatrix} \mathbf{0}_{M \times M} & I_M \\ -\ell\bar{\Psi} & -\gamma\bar{\Psi} \end{bmatrix}, \quad P = \begin{bmatrix} \rho\bar{\Psi} & \epsilon\bar{\Psi} \\ \epsilon\bar{\Psi} & \rho\bar{\Psi} \end{bmatrix}, \quad (19)$$

where I_M is the identity matrix of order M , and ρ , ϵ , γ , and ℓ are positive constants satisfying $\rho > \epsilon$. Then, the matrix $Q = PA + A^T P$ is negative definite, if the following conditions are satisfied:

$$\gamma\epsilon = \ell\rho, \quad \frac{\epsilon}{\ell\epsilon + \rho\gamma} < \mu_{min}, \quad \frac{\rho^2}{4\ell(\rho^2 - \epsilon^2)} < \mu_{min}, \quad (20)$$

where μ_{min} is the smallest eigenvalue of $\bar{\Psi}$.

Proof: Using (19), the matrix Q can be written as

$$Q = \begin{bmatrix} -2\ell\bar{\Psi}^2 & \rho\bar{\Psi} - (\epsilon\gamma + \ell\rho)\bar{\Psi}^2 \\ \rho\bar{\Psi} - (\epsilon\gamma + \ell\rho)\bar{\Psi}^2 & 2\epsilon\bar{\Psi} - 2\gamma\rho\bar{\Psi}^2 \end{bmatrix}. \quad (21)$$

The eigenvalues of Q can be found using the following characteristic equation:

$$|sI - Q| = \begin{vmatrix} sI + 2\ell\bar{\Psi}^2 & -\rho\bar{\Psi} + (\epsilon\gamma + \ell\rho)\bar{\Psi}^2 \\ -\rho\bar{\Psi} + (\epsilon\gamma + \ell\rho)\bar{\Psi}^2 & sI - 2\epsilon\bar{\Psi} + 2\gamma\rho\bar{\Psi}^2 \end{vmatrix} = 0.$$

Note that $\begin{vmatrix} \hat{A} & \hat{B} \\ \hat{C} & \hat{D} \end{vmatrix} = |\hat{A}\hat{D} - \hat{C}\hat{B}|$ if \hat{A} and \hat{C} commute. Also, using the eigenvalue properties of functions of a square matrix [33], it can be shown that $|s^2I + h_1(\bar{\Psi})s + h_2(\bar{\Psi})| = |sI - \bar{h}_1(\bar{\Psi})| \cdot |sI - \bar{h}_2(\bar{\Psi})| = \prod_{i=1}^M [s - \bar{h}_1(\mu_i)] \cdot \prod_{i=1}^M [s - \bar{h}_2(\mu_i)] = \prod_{i=1}^M (s^2 + h_1(\mu_i)s + h_2(\mu_i))$, where $h_1(\cdot)$, $h_2(\cdot)$, $\bar{h}_1(\cdot)$ and $\bar{h}_2(\cdot)$ are polynomials of any order so that $\bar{h}_1(\cdot) + \bar{h}_2(\cdot) = -h_1(\cdot)$ and $\bar{h}_1(\cdot)\bar{h}_2(\cdot) = h_2(\cdot)$, and μ_i is the i th eigenvalue of $\bar{\Psi}$. Thus, we have

$$\begin{aligned} |sI - Q| &= |s^2I + 2(-\epsilon\bar{\Psi} + (\ell\epsilon + \rho\gamma)\bar{\Psi}^2)s + 4\ell\epsilon\bar{\Psi}^2(\rho\gamma\bar{\Psi}^2 - \epsilon\bar{\Psi}) - (\rho\bar{\Psi} - (\gamma\epsilon + \ell\rho)\bar{\Psi}^2)^2| \\ &= \prod_{i=1}^M [s^2 + 2(-\epsilon\mu_i + (\ell\epsilon + \rho\gamma)\mu_i^2)s + 4\ell\epsilon\mu_i^2(\rho\gamma\mu_i^2 - \epsilon\mu_i) - (\rho\mu_i - (\gamma\epsilon + \ell\rho)\mu_i^2)^2]. \end{aligned}$$

To ensure that the eigenvalues of Q are in the left-half complex plane, the coefficient of s and the constant need to be positive, i.e.,

$$\begin{cases} \mu_i(-\epsilon + (\ell\epsilon + \rho\gamma)\mu_i) > 0 \\ \mu_i^2(4\ell\epsilon(\rho\gamma\mu_i^2 - \epsilon\mu_i) - (\rho - (\gamma\epsilon + \ell\rho)\mu_i)^2) > 0. \end{cases}$$

With some algebraic manipulations and note that $\mu_i > 0$, the following conditions guarantee that the eigenvalues of Q lie in the left-half complex plane:

$$\begin{cases} -\epsilon + (\ell\epsilon + \rho\gamma)\mu_i > 0 \\ -(\gamma\epsilon - \ell\rho)^2\mu_i^2 + (-4\ell\epsilon^2 + 2\rho(\gamma\epsilon + \ell\rho))\mu_i - \rho^2 > 0. \end{cases}$$

The above inequalities are guaranteed by the conditions given in (20). Thus, the proof is completed. \square

Remark 5: The constant μ_{min} in (20) is the smallest non-zero eigenvalue of the augmented graph $\bar{\mathcal{G}}$, which is called algebraic connectivity. Note that μ_{min} is dependent on the size and connectivity of the considered network and determines the convergence rate of cooperative algorithms [2]. In general, for networks that have smaller values of μ_{min} , in order to satisfy the conditions given in (23), larger values for some of the control design gains (e.g., ℓ , γ , ρ , and ϵ) may be needed. Comparing to the formation control algorithm given in [34], Lemma 3 includes more control parameters (e.g., ρ and ℓ) and allows more flexibility in controller design.

Based on the system model (11), the neural network model (10), and Assumption 3, an adaptive neural controller can be designed using distributed neural-network-based adaptive approximation and adaptive bounding control techniques. Specifically, we consider the following distributed controller algorithm, for $i = 1, \dots, M$,

$$u_i = \frac{1}{1 + u_{\theta_i}} \bar{u}_i \quad (22)$$

$$\bar{u}_i \triangleq -\phi_i(x_i) - \sum_{j \in N_i} z_{ij} (\ell \bar{p}_{ij} + \gamma \bar{v}_{ij}) - \psi_i^0 - \hat{f}_i(x_i, \hat{\vartheta}_i(t)) - (\bar{\eta}_i + \hat{\kappa}_i) \text{sgn}(\Xi_i) \quad (23)$$

$$u_{\theta_i} \triangleq \begin{cases} 0, & \text{for } \Xi_i \bar{u}_i \geq 0 \\ \hat{\theta}_i, & \text{for } \Xi_i \bar{u}_i < 0 \end{cases} \quad (24)$$

$$\dot{\hat{\vartheta}}_i = \Gamma_i \sum_{j \in N_i} z_{ij} (\epsilon \bar{p}_{ij} + \rho \bar{v}_{ij}) \varphi_i(x_i) \quad (25)$$

$$\psi_i^0 = \hat{\alpha}_i^0 \bar{\Delta}_i(x_i) \text{sgn} \left(\sum_{j \in N_i} z_{ij} (\epsilon \bar{p}_{ij} + \rho \bar{v}_{ij}) \right) \quad (26)$$

$$\dot{\hat{\alpha}}_i^0 = \Upsilon_i \left| \sum_{j \in N_i} z_{ij} (\epsilon \bar{p}_{ij} + \rho \bar{v}_{ij}) \right| \bar{\Delta}_i(x_i) \quad (27)$$

$$\dot{\hat{\kappa}}_i = \bar{\Upsilon}_i \left| \sum_{j \in N_i} z_{ij} (\epsilon \bar{p}_{ij} + \rho \bar{v}_{ij}) \right| \quad (28)$$

$$\dot{\hat{\theta}}_i = \begin{cases} 0, & \text{for } \Xi_i \bar{u}_i \geq 0 \\ \mathcal{P} \left[\bar{\Gamma}_i \sum_{j \in N_i} z_{ij} (\epsilon \bar{p}_{ij} + \rho \bar{v}_{ij}) u_i \right], & \text{for } \Xi_i \bar{u}_i < 0 \end{cases}, \quad (29)$$

where $\bar{p}_{ij} \triangleq p_i - p_j - \bar{p}_{ij}$ and $\bar{v}_{ij} \triangleq v_i - v_j$, \bar{p}_{ij} are the constant desired relative positions between the agents i and j , z_{ij} are constant design gains to be defined later in (31), ℓ , γ , ρ , and ϵ are positive constants defined in Lemma 3, $\text{sgn}(\cdot)$ is the sign function defined to take zero value at zero, $\Xi_i \triangleq \sum_{j \in N_i} z_{ij} (\epsilon \bar{p}_{ij} + \rho \bar{v}_{ij})$, $\bar{\Delta}_i \triangleq \bar{\delta}_i + |\varphi_i(x_i)|$, $\hat{\theta}_i$ is an estimation of the lower bound on actuator fault parameter $\theta_i(t)$ (i.e., θ_i^*), $\hat{\vartheta}_i$ is an estimation of the neural network parameter vector ϑ_i , $\varphi_i \triangleq \text{col}(\bar{\varphi}_j : j = 1, \dots, \bar{c})$ is the collective vector of fixed basis functions, $\hat{\alpha}_i^0$ is an estimation of the unknown bounding constant $\alpha_{m_i}^0$ described in (13), $\hat{\kappa}_i$ is an estimation of the unknown positive constant bound κ on $|\dot{v}_0|$ (see Assumption 2), Γ_i is a symmetric positive definite learning rate matrix, and $\bar{\Gamma}_i$, Υ_i and $\bar{\Upsilon}_i$ are positive learning rate constants. Additionally, in order to prevent the denominator in the feedback control law (22) coming too close to zero, let ϵ_θ be a small positive number such that $1 + \hat{\theta}_i > \epsilon_\theta$ denotes a safe distance for the denominator to be bounded away from zero. Therefore, it is required that $\hat{\theta}_i > -1 + \epsilon_\theta$, which is ensured by the the projection operator \mathcal{P} in (29).

Remark 6: In the control law (23), the terms $-\sum_{j \in N_i} z_{ij} (\ell \bar{p}_{ij} + \gamma \bar{v}_{ij}) - \phi_i(x_i)$ guarantee the convergence of formation error (i.e., $p_i - p_j$ approaches \bar{p}_{ij} , and v_i approaches v_0), for the ideal case of constant leader velocity (i.e., $\dot{v}_0 = 0$) and absence of faults and modeling uncertainty. The last term in (23) (i.e., $-(\bar{\eta}_i + \hat{\kappa}_i) \text{sgn}(\Xi_i)$) and the adaptive law (28) are designed to guarantee the robustness of leader-follower formation with respect to modeling uncertainty and a time-varying leader with an unknown bound on the derivative of the leader's second state (see Assumption 2). The adaptive term $\frac{1}{1+u_{\theta_i}}$ in the control law (22) and the adaptive law (29) are used to compensate for the effect of time-varying actuator faults. The term ψ_i^0 in (26) and the adaptive law (27) are designed to deal with the network residual approximation error as well as the uncertainty associated with the incipient fault time profile. Lastly, the term $\hat{f}_i(x_i, \hat{\vartheta}_i(t))$ in (23) is the adaptive neural-network approximator with adaptive law (25) designed to approximate the process fault function and compensate for its effect.

Using some algebraic manipulations, we can rewrite (22) as $u_i = \bar{u}_i - u_{\theta_i} u_i$. Therefore, by using (23)

and substituting u_i into (11), the closed-loop system dynamics are given by

$$\begin{aligned}\dot{p}_i &= v_i \\ \dot{v}_i &= - \sum_{j \in N_i} z_{ij} (\ell \tilde{p}_{ij} + \gamma \tilde{v}_{ij}) + \eta_i - (\bar{\eta}_i + \hat{\kappa}_i) \text{sgn}(\Xi_i) - \psi_i^0 + \hat{f}_i(x_i, \vartheta_i(t)) - \hat{f}_i(x_i, \hat{\vartheta}_i(t)) \\ &\quad + (\theta_i(t) - u_{\theta_i}) u_i + \beta_i \delta_i(x_i) + (\beta_i - 1) \hat{f}_i(x_i, \vartheta_i).\end{aligned}\quad (30)$$

We choose the following distributed controller gains: for $i = 1, \dots, M$, and $j \in N_i$,

$$z_{ij} = \begin{cases} \chi_i k_{ij} + \chi_j k_{ji}, & \text{for } j \neq 0 \\ \chi_i k_{i0} + \chi_0 \bar{k}_i, & \text{for } j = 0 \end{cases} \quad (31)$$

where χ_i is defined in Lemma 1, \bar{k}_i is defined in the proof of Lemma 1, and k_{ij} and k_{ji} are positive constants denoting the weights on the intercommunication graph \mathcal{G} .

Note that the distributed gains z_{ij} given in (31) are the i th row and j th column entries of $\bar{\Psi}$ defined in Lemma 2. Therefore, $\sum_{j \in N_i} z_{ij} (\ell \tilde{p}_{ij} + \gamma \tilde{v}_{ij}) = \ell \bar{\Psi} \tilde{p} + \gamma \bar{\Psi} \tilde{v}$, where \tilde{p} is the column stack vector of $\tilde{p}_i \triangleq p_i - \bar{p}_i$, \bar{p}_i is the unknown constant desired relative position between the leader and agent i , and \tilde{v} is the column stack vector of $\tilde{v}_i \triangleq v_i - v_0$. Using (30) and (19), we represent the collective tracking error dynamics as

$$\dot{\tilde{x}} = A \tilde{x} + \begin{bmatrix} \mathbf{0}_M \\ \zeta - \bar{\zeta} - \mathbf{1}_M \dot{v}_0 + \tilde{f} + \Delta - \psi^0 + \varpi \end{bmatrix}, \quad (32)$$

where A is defined in Lemma 3 with the positive definite matrix $\bar{\Psi}$ given in Lemma 2, $\tilde{x} = [\tilde{p}^T \ \tilde{v}^T]^T \in \mathfrak{R}^{2M}$, and the terms $\zeta \in \mathfrak{R}^M$, $\bar{\zeta} \in \mathfrak{R}^M$, $\tilde{f} \in \mathfrak{R}^M$, $\Delta \in \mathfrak{R}^M$, $\psi^0 \in \mathfrak{R}^M$ and $\varpi \in \mathfrak{R}^M$ are defined as

$$\zeta \triangleq [\eta_1, \dots, \eta_M]^T, \quad (33)$$

$$\bar{\zeta} \triangleq [\bar{\zeta}_1, \dots, \bar{\zeta}_M]^T, \quad (34)$$

$$\tilde{f} \triangleq [\tilde{f}_1, \dots, \tilde{f}_M]^T, \quad (35)$$

$$\Delta \triangleq [\Delta_1, \dots, \Delta_M]^T, \quad (36)$$

$$\psi^0 \triangleq [\psi_1^0, \dots, \psi_M^0]^T, \quad (37)$$

$$\varpi \triangleq [\varpi_1, \dots, \varpi_M]^T, \quad (38)$$

and $\bar{\zeta}_i = (\bar{\eta}_i + \hat{\kappa}_i) \text{sgn}(\Xi_i)$, $i = 1, \dots, M$, $\tilde{f}_i \triangleq \tilde{\vartheta}_i^T \varphi_i$, $\tilde{\vartheta}_i = \vartheta_i - \hat{\vartheta}_i$ is the network parameter estimation error associated with the i th agent, $\Delta_i = \beta_i \delta_i + (\beta_i - 1) \hat{f}_i(x_i, \vartheta_i)$, $\varpi_i \triangleq (\theta_i(t) - u_{\theta_i}) u_i$, and ψ_i^0 is defined in (26).

To derive the adaptive algorithm and to investigate analytically the stability properties of the closed-loop system, we consider the following Lyapunov function candidate:

$$V = \tilde{x}^T P \tilde{x} + \tilde{\vartheta}^T (\Gamma)^{-1} \tilde{\vartheta} + (\tilde{\alpha}^0)^T (\Upsilon)^{-1} \tilde{\alpha}^0 + \tilde{\theta}^T (\bar{\Gamma})^{-1} \tilde{\theta} + \tilde{\kappa}^T (\tilde{\Upsilon})^{-1} \tilde{\kappa}, \quad (39)$$

where P is defined in Lemma 3, $\tilde{\vartheta} = [\tilde{\vartheta}_1^T, \dots, \tilde{\vartheta}_M^T]^T$ is the collective parameter estimation errors for the neural network model, $\tilde{\alpha}^0 = [\tilde{\alpha}_1^0, \dots, \tilde{\alpha}_M^0]^T$ is the collective bounding parameter estimation errors defined as $\tilde{\alpha}_i^0 = \alpha_{m_i}^0 - \hat{\alpha}_i^0$, $\tilde{\theta} = [\tilde{\theta}_1, \dots, \tilde{\theta}_M]^T$ is the collective estimation errors of actuator fault parameter lower bound defined as $\tilde{\theta}_i = \theta_i^* - \hat{\theta}_i$, $\tilde{\kappa} = [\tilde{\kappa}_1, \dots, \tilde{\kappa}_M]^T$ is the collective bounding parameter estimation errors defined as $\tilde{\kappa}_i = \kappa - \hat{\kappa}_i$, and $\Gamma = \text{diag}\{\Gamma_1, \dots, \Gamma_M\}$, $\bar{\Gamma} = \text{diag}\{\bar{\Gamma}_1, \dots, \bar{\Gamma}_M\}$, $\Upsilon = \text{diag}\{\Upsilon_1, \dots, \Upsilon_M\}$ and $\tilde{\Upsilon} = \text{diag}\{\tilde{\Upsilon}_1, \dots, \tilde{\Upsilon}_M\}$ are constant learning rate matrices.

Then, the time derivative of the Lyapunov function (39) along the solution of (32) is given by

$$\begin{aligned} \dot{V} = & \tilde{x}^T Q \tilde{x} + 2\tilde{x}^T P \left[\zeta - \bar{\zeta} - \mathbf{1}_M \dot{v}_0 + \tilde{f} + \Delta - \psi^0 + \varpi \right] + \tilde{\theta}^T (\Gamma)^{-1} \dot{\tilde{\theta}} + (\tilde{\alpha}^0)^T (\Upsilon)^{-1} \dot{\tilde{\alpha}}^0 \\ & + \tilde{\theta}^T (\bar{\Gamma})^{-1} \dot{\tilde{\theta}} + \tilde{\kappa}^T (\bar{\Upsilon})^{-1} \dot{\tilde{\kappa}}, \end{aligned} \quad (40)$$

where Q is defined in Lemma 3, and $\mathbf{0}_M$ is a $M \times 1$ column vector of zeros. Based on (19), (33) – (38), and by using $\tilde{p}_i = \tilde{p}_{i0}$, $\tilde{v}_i = \tilde{v}_{i0}$, $\tilde{p}_{ij} = \tilde{p}_i - \tilde{p}_j$, and the left eigenvector property (i.e., $\sum_{j \in N_i} \chi_i k_{ij} = \sum_{j \in N_i} \chi_j k_{ji}$), we have

$$\tilde{x}^T P \begin{bmatrix} \mathbf{0}_M \\ \zeta \end{bmatrix} = \epsilon \tilde{p}^T \bar{\Psi} \zeta + \rho \tilde{v}^T \bar{\Psi} \zeta = \sum_{i=1}^M \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) \eta_i, \quad (41)$$

$$\tilde{x}^T P \begin{bmatrix} \mathbf{0}_M \\ \bar{\zeta} \end{bmatrix} = \epsilon \tilde{p}^T \bar{\Psi} \bar{\zeta} + \rho \tilde{v}^T \bar{\Psi} \bar{\zeta} = \sum_{i=1}^M \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) (\bar{\eta}_i + \hat{\kappa}_i) \text{sgn}(\Xi_i), \quad (42)$$

$$\tilde{x}^T P \begin{bmatrix} \mathbf{0}_M \\ \mathbf{1}_M \dot{v}_0 \end{bmatrix} = \epsilon \tilde{p}^T \bar{\Psi} \mathbf{1}_M \dot{v}_0 + \rho \tilde{v}^T \bar{\Psi} \mathbf{1}_M \dot{v}_0 = \sum_{i=1}^M \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) \dot{v}_0, \quad (43)$$

$$\tilde{x}^T P \begin{bmatrix} \mathbf{0}_M \\ \tilde{f} \end{bmatrix} = \epsilon \tilde{p}^T \bar{\Psi} \tilde{f} + \rho \tilde{v}^T \bar{\Psi} \tilde{f} = \sum_{i=1}^M \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) \tilde{\vartheta}_i^T \varphi_i, \quad (44)$$

$$\tilde{x}^T P \begin{bmatrix} \mathbf{0}_M \\ \Delta \end{bmatrix} = \epsilon \tilde{p}^T \bar{\Psi} \Delta + \rho \tilde{v}^T \bar{\Psi} \Delta = \sum_{i=1}^M \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) \Delta_i, \quad (45)$$

$$\tilde{x}^T P \begin{bmatrix} \mathbf{0}_M \\ \psi^0 \end{bmatrix} = \epsilon \tilde{p}^T \bar{\Psi} \psi^0 + \rho \tilde{v}^T \bar{\Psi} \psi^0 = \sum_{i=1}^M \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) \psi_i^0, \quad (46)$$

$$\tilde{x}^T P \begin{bmatrix} \mathbf{0}_M \\ \varpi \end{bmatrix} = \epsilon \tilde{p}^T \bar{\Psi} \varpi + \rho \tilde{v}^T \bar{\Psi} \varpi = \sum_{i=1}^M \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) (\theta_i(t) - u_{\theta_i}) u_i. \quad (47)$$

By substituting (41) – (47) into (40), we have

$$\begin{aligned} \dot{V} = & \tilde{x}^T Q \tilde{x} + 2 \sum_{i=1}^M \left[\sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) (\eta_i - \dot{v}_0) - \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) (\bar{\eta}_i + \hat{\kappa}_i) \text{sgn}(\Xi_i) \right. \\ & + \tilde{\theta}_i^T \left(-(\Gamma_i)^{-1} \dot{\tilde{\theta}}_i + \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) \varphi_i \right) - \tilde{\alpha}_i^0 (\Upsilon_i)^{-1} \dot{\tilde{\alpha}}_i^0 + \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) (\Delta_i - \psi_i^0) \\ & \left. + \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) (\theta_i(t) - u_{\theta_i}) u_i - \tilde{\theta}_i (\bar{\Gamma}_i)^{-1} \dot{\tilde{\theta}}_i - \tilde{\kappa}_i^T (\bar{\Upsilon}_i)^{-1} \dot{\tilde{\kappa}}_i \right]. \end{aligned} \quad (48)$$

Based on Assumption 1, we have

$$\eta_i \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) - \bar{\eta}_i \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) \text{sgn}(\Xi_i) \leq 0. \quad (49)$$

By using (24) and selecting the adaptive algorithm for $\hat{\theta}_i$ as (29), we have

$$\Xi_i (\theta_i(t) - u_{\theta_i}) u_i - \tilde{\theta}_i (\bar{\Gamma}_i)^{-1} \dot{\tilde{\theta}}_i = \begin{cases} \theta_i(t) \Xi_i u_i, & \text{for } \Xi_i \bar{u}_i \geq 0 \\ (\theta_i(t) - \theta_i^*) \Xi_i u_i, & \text{for } \Xi_i \bar{u}_i < 0 \end{cases}. \quad (50)$$

Note that $-1 < \theta_i^* \leq \theta_i(t) \leq 0$, and based on (23), u_i and \bar{u}_i have the same sign. Therefore, it follows from (50) that

$$\Xi_i (\theta_i(t) - u_{\theta_i}) u_i - \tilde{\theta}_i (\bar{\Gamma}_i)^{-1} \dot{\tilde{\theta}}_i \leq 0.$$

Note that since the parameter projection modification can only make the Lyapunov function derivative more negative, the stability properties derived for the standard algorithm still hold [20]. Therefore, by applying the above inequalities to (48) and selecting the adaptive algorithm for $\hat{\vartheta}_i$ as (25), we have

$$\begin{aligned} \dot{V} \leq & \tilde{x}^T Q \tilde{x} + 2 \sum_{i=1}^M \left(-\tilde{\alpha}_i^0 (\Upsilon_i)^{-1} \dot{\hat{\alpha}}_i^0 + \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) (\Delta_i - \psi_i^0) \right) + 2 \sum_{i=1}^M \left(-\tilde{\kappa}_i^T (\tilde{\Upsilon}_i)^{-1} \dot{\hat{\kappa}}_i \right. \\ & \left. + \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) [-\dot{v}_0 - \hat{\kappa}_i \text{sgn}(\Xi_i)] \right). \end{aligned} \quad (51)$$

By using (26), Assumption 3, and based on $|\Delta_i| \leq \alpha_{m_i}^0 \bar{\Delta}_i$ we have

$$\begin{aligned} \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) (\Delta_i - \psi_i^0) &= \Xi_i [\Delta_i - \hat{\alpha}_i^0 \bar{\Delta}_i \text{sgn}(\Xi_i)] \\ &\leq |\Xi_i| \hat{\alpha}_i^0 \bar{\Delta}_i. \end{aligned} \quad (52)$$

By using (51), (52) and Assumption 2, we obtain

$$\dot{V} \leq \tilde{x}^T Q \tilde{x} + 2 \sum_{i=1}^M \left[|\Xi_i| \hat{\alpha}_i^0 \bar{\Delta}_i - \tilde{\alpha}_i^0 (\Upsilon_i)^{-1} \dot{\hat{\alpha}}_i^0 + |\Xi_i| \tilde{\kappa}_i - \tilde{\kappa}_i^T (\tilde{\Upsilon}_i)^{-1} \dot{\hat{\kappa}}_i \right].$$

Therefore, by using adaptive laws (27) and (28) for $\hat{\alpha}_i^0$ and $\hat{\kappa}_i$, respectively, we have

$$\dot{V} \leq \tilde{x}^T Q \tilde{x} \leq 0, \quad (53)$$

where Q is given in Lemma 3. Using Lemma 3, we know that \dot{V} is negative semidefinite. Based on the definition of V , we conclude that $\tilde{x}_i = [\tilde{p}_i, \tilde{v}_i]^T$, $\hat{\vartheta}_i$, $\hat{\theta}_i$, $\hat{\kappa}_i$ and $\hat{\alpha}_i^0$ are uniformly bounded. By integrating both sides of (53), it can be easily shown that $\tilde{x}_i \in L_2$. Additionally, x_i is bounded because \tilde{x}_i and the leader's state x_0 are bounded. Therefore, based on (23), (11), and the smoothness of the function ϕ_i , we have $\tilde{u}_i \in L_\infty$ and $\dot{x}_i \in L_\infty$. Since $\tilde{x}_i \in L_\infty \cap L_2$ and $\dot{\tilde{x}}_i \in L_\infty$, based on Barbalat's Lemma [35], we can conclude that the leader-following formation between agents' outputs is reached asymptotically, i.e., $\tilde{x}_i \rightarrow 0$ as $t \rightarrow \infty$.

The aforementioned design and analysis procedure is summarized in the following theorem:

Theorem 1. *Suppose that Assumptions 1–3 hold. Then, if a fault is detected, by using the distributed controller gains given by (31), the adaptive fault-tolerant law (22), the neural network weight parameter adaptive law (25), the actuator fault parameter adaptive law (29), and the bounding parameter adaptive laws (26) – (28) guarantee that all the signals and parameter estimates are uniformly bounded, i.e., \tilde{x}_i , $\hat{\vartheta}_i$, $\hat{\kappa}_i$, $\hat{\theta}_i$ and $\hat{\alpha}_i^0$ are bounded, and the leader-following formation is achieved asymptotically with a time-varying reference state, i.e., $p_i(t) - p_j(t) \rightarrow \bar{p}_{ij}$ and $v_i(t) \rightarrow v_0(t)$ as $t \rightarrow \infty$.*

Remark 7: Under the communication topology considered in this paper, the leader only communicates to a small subset of followers, and each follower only communicates to its directly connected neighbors through a bidirectional network with possibly asymmetric weights. This makes it more difficult to accomplish the asymptotic convergence property of leader-following consensus error in the presence of faults and modeling uncertainty. Therefore, the important Lemmas 1–3 provide a skillfully chosen positive definite matrix P and negative definite matrix Q , which are crucial for designing an appropriate Lyapunov function using the distributed gains (31) for deriving the asymptotic fault-tolerant leader-following consensus property of the multi-agent system under a bidirectional communication topology with general asymmetric weights.

Remark 8: Interesting adaptive-approximation-based cooperative tracking control algorithms have been developed for undirected communication graphs (i.e, with a symmetric Laplacian matrix) [6,12], strongly connected directed graphs [5,16], and directed graphs having a spanning tree [31], while this paper considers bidirectional graphs with asymmetric weights. In [5,6,12,16,31], a constant bound was assumed for the residual approximation error, while the bound is allowed to be a function of agent’s state in this paper (see Assumption 3), therefore increasing the complexity of stability analysis. Additionally, CUUB results were obtained in [5,16]. Specifically, adaptive algorithms were not developed in [5] to deal with actuator faults, neural network residual approximation error, and the bound on the derivative of the leader’s second state (see Assumption 2). A discontinuous adaptive sliding mode control law was developed in [16], where the bound on the formation tracking error is derived as a function of several assumed constant bounds representing the overall effects of the neural network parameter estimation error, optimal weights, residual approximation error, and parameter estimate, as well as the fault signal in each agent. In contrast, the formation error is guaranteed to asymptotically converge to zero by the adaptive FTC algorithm proposed in this paper. Furthermore, in [31] the leader’s state and its derivative are assumed to be bounded by a known constant that is available to all agents, which allow the design of distributed observers for each agent to estimate the state of the leader. Note that this critical assumption is not needed for the distributed FTC algorithm designed in this paper, which only requires an *unknown* bound on the derivative of the leader’s state.

Remark 9: An adaptive cooperative tracking algorithm for higher-order linear multi-agent systems under undirected graph has been presented in [36], where a discontinuous adaptive control law using sign function is used to estimate an unknown constant bound on the derivative of the leader’s state. It is shown that the states of the followers asymptotically converge to the leader using nonsmooth analysis in the sense of Filippov solutions and Barbalat’s Lemma. As described in [36], the sign function is measurable and locally essentially bounded and therefore the Filippov solution exists. A similar reasoning logic is applicable to the discontinuous terms in this paper.

4 Fault Accommodation: after Fault Isolation

4.1 Decentralized Fault Isolation

To facilitate fault isolation, it is assumed there is only a single fault in each agent at any time. However, the system model (1) allows the occurrence of simultaneous faults in multiple agents. Thus, the decentralized fault isolation method can be easily designed using the results in [23]. Let us assume that a fault is detected in the i th agent at some time T_d ; accordingly, at $t = T_d$ the fault isolation algorithm in the local FDI component is activated which consists of a bank of fault isolation estimators (FIEs). Each local FIE is designed based on the functional structure of a particular fault type in the agent [see (5)]. Specifically, the following r_i nonlinear adaptive estimators are designed as isolation estimators for the i th agent: for $s = 1, \dots, r_i$,

$$\dot{\hat{x}}_i^s = -\Lambda_i^s(\hat{x}_i^s - x_i) + \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} (\phi_i(x_i) + u_i + (\hat{\rho}_i^s)^T \bar{g}_i^s(x_i, u_i)), \quad (54)$$

where \bar{g}_i^s , for $s = 1, \dots, r_i$, represents the functional structure of the s th fault (see (5), i.e., $\bar{g}_i^s \triangleq g_i^s$ for process faults and $\bar{g}_i^s \triangleq u_i$ for the actuator fault), $\hat{\rho}_i^s$ is the estimate of the fault parameter vector in the i th agent, and $\Lambda_i^s = \begin{bmatrix} \lambda_{p_i}^s & 0 \\ 0 & \lambda_{v_i}^s \end{bmatrix}$ is a diagonal positive definite matrix.

The adaptation in the isolation estimators is due to the unknown fault parameter vector ρ_i^s . The adaptive law for updating each $\hat{\rho}_i^s$ is derived by using the Lyapunov synthesis approach (see, for instance,

[35]), with a projection operator \mathcal{P} restricting $\hat{\varrho}_i^s$ to the corresponding known set $\bar{\varrho}_i^s$, where $\bar{\varrho}_i^w \triangleq \Theta_i^w$, $w = 1, \dots, r_i - 1$, for process faults and $\bar{\varrho}_i^r \triangleq (-1 + \epsilon_\theta, 0]$ for the actuator fault (see (29)). Specifically, if we let $\varepsilon_i^s(t) = x_i - \hat{x}_i^s = [\varepsilon_{p_i}^s, \varepsilon_{v_i}^s]^T$ be the estimation error generated by the sth FIE associated with the i th agent, then the following adaptive algorithm is chosen:

$$\dot{\hat{\varrho}}_i^s = \mathcal{P}_{\bar{\varrho}_i^s} \{ \gamma_i^s \bar{g}_i^s(x_i, u_i) \varepsilon_{v_i}^s \},$$

where $\gamma_i^s > 0$ is a constant learning rate.

By following the reasoning logic given in [27], a bound on each component of the state estimation error can be obtained as $|\varepsilon_{v_i}^s| \leq \varsigma_i^s(t)$, where

$$\varsigma_i^s \triangleq \int_{T_d}^t e^{-\lambda_{v_i}^s(t-\tau)} (\bar{\eta}_i + (\xi_i^s + \hat{\varrho}_i^s(\tau)) \cdot |\bar{g}_i^s(x_i, u_i)|) d\tau + \bar{x}_i^s e^{-\lambda_{v_i}^s(t-T_d)},$$

and \bar{x}_i^s is a possibly conservative bound on the initial state estimation error (i.e., $|\varepsilon_{v_i}^s(T_d)| \leq \bar{x}_i^s$), ξ_i^s represents the maximum fault parameter vector estimation error (i.e., $|\varrho_i^s - \hat{\varrho}_i^s(t)| \leq \xi_i^s$), for $i = 1, \dots, M$. Note that the form of ξ_i^s depends on the geometric properties of the compact set $\bar{\varrho}_i^s$ [27]. For instance, assume that the parameter set $\bar{\varrho}_i^s$ is the smallest hypersphere containing the set of all possible $\hat{\varrho}_i^s(t)$ with center O_i^s and radius R_i^s ; then we have $\xi_i^s = R_i^s + |\hat{\varrho}_i^s(t) - O_i^s|$.

Thus, based on the generalized observer scheme, the following fault isolation decision scheme is devised: If for each $b \in \{1, \dots, r_i\} \setminus \{s\}$, there exist some finite time $t^b > T_d$, such that $|\varepsilon_{v_i}^b(t^b)| > \varsigma_i^b(t^b)$, then the occurrence of fault s in the i th subsystem is concluded.

4.2 Adaptive FTCs after Fault Isolation

Let us now assume that the isolation procedure described in Section 4.1 provides the information that fault s , for $s = 1, \dots, r_i$, has been isolated at time T_{isol} . Specifically, the fault isolation decision scheme determines the sth fault type in the fault class \mathcal{F}_i described by (5) has occurred. Then, the controller is reconfigured again to further improve control performance based on the diagnostic information of isolated fault type. In the following sections, we detail the FTC designs for the two cases of process faults and actuator faults, respectively.

4.2.1 Adaptive Fault-Tolerant Controller for Process Faults

After the isolation of process fault type s described by (4), i.e., $t \geq T_{isol}$, the dynamics of the system takes on the following form:

$$\dot{x}_i = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} (\phi_i(x_i) + u_i + \eta_i + \beta_i (\theta_i^w)^T g_i^w(x_i)). \quad (55)$$

We let $\alpha_{m_i}^p$ represent an unknown bounding constant defined as

$$\alpha_{m_i}^p \triangleq \sup_{t \geq T_{isol}} \max\{|\beta_i(t - T_i) - 1| \theta_i^w\}. \quad (56)$$

The control objective is to have the state x_i , for $i = 1, \dots, M$, track the time-varying bounded state of the leader and form a predefined formation. The following adaptive fault-tolerant controller is adopted:

$$u_i = -\phi_i(x_i) - \sum_{j \in N_i} z_{ij}(\ell \tilde{p}_{ij} + \gamma \tilde{v}_{ij}) - (\hat{\theta}_i^w)^T g_i^w(x_i) - (\bar{\eta}_i + \hat{\kappa}_i) \text{sgn}(\Xi_i) - \psi_i^p \quad (57)$$

$$\dot{\hat{\theta}}_i^w = \Gamma_i \sum_{j \in N_i} z_{ij}(\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) g_i^w(x_i) \quad (58)$$

$$\psi_i^p = \hat{\alpha}_i^p |g_i^w(x_i)| \text{sgn}\left(\sum_{j \in N_i} z_{ij}(\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij})\right) \quad (59)$$

$$\dot{\hat{\alpha}}_i^p = \Upsilon_i \left| \sum_{j \in N_i} z_{ij}(\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) \right| \cdot |g_i^w(x_i)| \quad (60)$$

$$\dot{\hat{\kappa}}_i = \tilde{\Upsilon}_i \left| \sum_{j \in N_i} z_{ij}(\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) \right|, \quad (61)$$

where $\hat{\theta}_i^w$ is an estimation of the unknown process fault parameter vector θ_i^w , $\hat{\alpha}_i^p$ is an estimation of the unknown constant $\alpha_{m_i}^p$ (see (56)), $\hat{\kappa}_i$ is an estimation of the unknown positive constant bound κ on $|\dot{v}_0|$ (see Assumption 2), Γ_i is a symmetric positive definite learning rate matrix, and Υ_i and $\tilde{\Upsilon}_i$ are positive learning rate constants. Then, we have the following results:

Theorem 2. *Suppose that Assumptions 1–2 hold. Assume that process fault s occurs at time T_i and that it is isolated at time T_{isol} . Then, by using the distributed controller gains given by (31), the fault-tolerant controller (57) and fault parameter adaptive laws (58) – (61) guarantee that all states are bounded, and the leader-following formation is achieved asymptotically with a time-varying reference state, i.e. $p_i(t) - p_j(t) \rightarrow \bar{p}_{ij}$ and $v_i(t) \rightarrow v_0(t)$ as $t \rightarrow \infty$.*

Proof: See Appendix A. □

4.2.2 Adaptive Fault-Tolerant Controller for Actuator Fault

After the isolation of an actuator fault, i.e., for $t \geq T_{isol}$, the dynamics of the system takes on the following form:

$$\dot{x}_i = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} (\phi_i(x_i) + (1 + \theta_i(t))u_i + \eta_i(x_i, t)). \quad (62)$$

The following adaptive fault-tolerant controller is adopted:

$$u_i = \frac{1}{1 + \bar{\tau}_{\theta_i}} \bar{\tau}_i \quad (63)$$

$$\bar{\tau}_i \triangleq -\phi_i(x_i) - \sum_{j \in N_i} z_{ij}(\ell \tilde{p}_{ij} + \gamma \tilde{v}_{ij}) - (\bar{\eta}_i + \hat{\kappa}_i) \text{sgn}\left(\sum_{j \in N_i} z_{ij}(\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij})\right) \quad (64)$$

$$\bar{\tau}_{\theta_i} \triangleq \begin{cases} 0, & \text{for } \Xi_i \bar{\tau}_i \geq 0 \\ \hat{\theta}_i, & \text{for } \Xi_i \bar{\tau}_i < 0 \end{cases} \quad (65)$$

$$\dot{\hat{\theta}}_i = \begin{cases} 0, & \text{for } \Xi_i \bar{\tau}_i \geq 0 \\ \mathcal{P}\left[\bar{\Gamma}_i \sum_{j \in N_i} z_{ij}(\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) u_i\right], & \text{for } \Xi_i \bar{\tau}_i < 0 \end{cases} \quad (66)$$

$$\dot{\hat{\kappa}}_i = \tilde{\Upsilon}_i \left| \sum_{j \in N_i} z_{ij}(\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) \right|, \quad (67)$$

where $\hat{\theta}_i$ is an estimation of the unknown lower bound on the actuator fault magnitude parameter $\theta_i(t)$ (i.e. θ_i^*) with the projection operator \mathcal{P} ensuring the denominator in the control law (63) stays at a safe distance from the point of singularity (as defined in (29)).

Then, we have the following theorem:

Theorem 3. *Suppose that Assumptions 1–2 hold. Assume that an actuator fault is isolated at time T_{isol} . Then, by using the distributed controller gains given by (31), the fault-tolerant controller (63) and adaptive laws (66) – (67) guarantee that the leader-follower formation is achieved asymptotically with a time-varying reference state, i.e., $p_i(t) - p_j(t) \rightarrow \bar{p}_{ij}$ and $v_i(t) \rightarrow v_0(t)$ as $t \rightarrow \infty$.*

Proof: See Appendix B. □

Remark 10: Compared with the first fault-tolerant controller (22), the second fault-tolerant controller employed after fault isolation (i.e., (57) for process faults or (63) for actuator faults) exploits the valuable information on the functional structure of the isolated fault and has the following advantages. First, the critical Assumption 3 on neural network approximation error required by the first fault-tolerant controller is no longer needed by the second controller. The neural network model employed by the first controller (22) typically needs more adaptive weights/parameters than the second controller to provide a good approximation of the unknown fault function, hence increasing computational complexity and implementation cost. Second, the first fault-tolerant controller (22) is more complicated, because it needs to take into account both process and actuator faults, while the second fault-tolerant controller only needs to handle a specific fault type that has been isolated. Therefore, the adaptive control algorithm can be significantly simplified compared with the fault-tolerant controller (22) – (29) employed after fault detection but before isolation, which is important for real-time implementation in practical applications.

5 Simulation Results

In this section, a simulation example of 5 agents is considered to illustrate the effectiveness of the distributed fault-tolerant formation control method. Motivated by the automated highway system considered in [37], the dynamics of the i th vehicle is given by

$$\dot{x}_i = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \frac{1}{m} (-A_\rho v_i^2 - d_f v_i + (1 + \theta_i(t))u_i + \eta_i + \beta_i(t - T_i)f_i(x_i)), \quad i = 1, \dots, 5, \quad (68)$$

where $x_i = [p_i, v_i]^T$ is the state of the i th agent consisting of the position p_i and velocity v_i , u_i is the input of i th agent representing the applied force in the longitudinal direction, m is the mass of the vehicle, A_ρ is aerodynamic drag coefficient, and d_f is a constant friction coefficient. The model (68), representing the longitudinal dynamics of autonomous ground vehicles, can be easily put into the general form (1) by letting the nominal term in the dynamics of each agent $\phi_i(x_i) = \frac{1}{m}(-A_\rho v_i^2 - d_f v_i)$. Note that the fault-tolerant controller design presented in this paper enables such ground mobile robots to follow predefined formations with speed tracking, even in the presence of process and actuator faults and modeling uncertainty.

The communication graph of the agents is given in Figure 1. As we can see, the leader only communicates to a small subset of followers, and each follower only communicates to its directly connected neighbors. The objective is to design the controller u_i to have each agent follow a virtual leader x_0 and also keep a predefined formation around the leader even in the presence of modeling uncertainty η_i and the possible occurrence of process and actuator faults. The unknown modeling uncertainty in the local dynamics of the agents are assumed to be a sinusoidal signal $\eta_i = 0.5\cos(v_i(t))$ bounded by $\bar{\eta}_i = 1$. The virtual leader x_0 is given by $\dot{x}_0 = \begin{bmatrix} v_0 \\ \sin(t) \end{bmatrix}$ with zero initial condition. The constant desired relative positions between the agents are $\bar{p}_{12} = 2$, $\bar{p}_{15} = 8$, $\bar{p}_{20} = 2$, $\bar{p}_{21} = -2$, $\bar{p}_{25} = 6$, $\bar{p}_{34} = 2$, $\bar{p}_{35} = 4$, $\bar{p}_{43} = -2$, $\bar{p}_{45} = 2$, $\bar{p}_{51} = -8$, $\bar{p}_{52} = -6$, $\bar{p}_{53} = -4$, $\bar{p}_{54} = -2$. The other model parameters used in the simulation example are $m = 1 \text{ kg}$, $A_\rho = 0.5 \text{ N s}^2/\text{m}^2$, and $d_f = 0.6 \text{ N s}/\text{m}$.

The virtual leader only communicates with the second agent (i.e., $k_{20} = 1$). It can be shown that the detail-balanced condition described in [18] and [19] is not satisfied. We choose $\bar{k}_2 = 0.5$. Then, the left eigenvector of $\bar{\Psi}$ associated with the zero eigenvalue is $\bar{\chi} = [0.425, 0.142, 0.212, 0.402, 0.521, 0.566]^T$. The matrix $\bar{\Psi}$ has the minimum eigenvalue of $\mu_{min} = 0.072$. We choose $\ell = 6$, $\gamma = 60$, $\epsilon = 0.15$, and $\rho = 1.5$ so that the conditions given in Lemma 3 are satisfied.

Before the fault is detected in the i th agent, the following nominal controller is employed which guarantees the leader-follower formation in the absence of faults.

$$\begin{aligned} u_i &= - \sum_{j \in N_i} z_{ij} (\ell \tilde{p}_{ij} + \gamma \tilde{v}_{ij}) - \phi_i(x_i) - (\bar{\eta}_i + \hat{\kappa}_i) \operatorname{sgn} \left(\sum_{j \in N_i} z_{ji} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) \right), \\ \dot{\hat{\kappa}}_i &= \Upsilon_i \left| \sum_{j \in N_i} z_{ji} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) \right|, \end{aligned}$$

Note that the baseline controller is a special case of the adaptive FTC described in Section 3. The fault class (see (5)) under consideration is defined as

1. A process fault results in extra abnormal friction in the vehicle dynamics, which physically can be caused by faults in the brake system or tires. Based on (68), the fault model is given by $f_i^1 = \theta_i^1 g_i^1$, where $g_i^1 = -d_f v_i$ is the fault functional structure, and the fault parameter $\theta_i^1 \in [0, 4]$ represents the unknown changes in the friction constant. Specifically, as a result of the fault, the friction constant d_f may increase from 0.6 Ns/m up to 3 Ns/m , representing the significance of the extra friction.
2. An actuator fault described by $f_i^2 = \theta_i^2(t) g_i^2$, where $g_i^2 = u_i$, and the fault magnitude $\theta_i^2(t) \in [-0.8, 0]$. The case of $\theta_i^2 = 0$ corresponds to a healthy actuator, whereas $-0.8 \leq \theta_i^2(t) < 0$ implies a partial actuator fault.

The estimator gain for the fault detection estimator is chosen as $\lambda_{p_i}^0 = \lambda_{v_i}^0 = 1$. For the fault isolation estimator, $\lambda_{p_i}^s = \lambda_{v_i}^s = 1$ has been chosen. A radial basis function (RBF) neural network is used for approximation of the fault after its detection but before its isolation. The RBF network considered in this paper consists of 2 neurons with 2 adjustable gain parameters. The center of radial basis functions are equally distributed on interval $[-10, 10]$ with a variance of 5. The initial parameter vector of the neural network is set to zero. We set the learning rates as $\Gamma_i = 0.05$, $\tilde{\Upsilon} = 0.05$ and $\bar{\Gamma}_i = 20$ and consider an unknown constant bound on the network approximation error, i.e., $\bar{\delta}_i = 1$. The learning rate is chosen as $\Upsilon_i = 0.5$.

We consider a process fault (i.e., $f_1^1 = \theta_1^1 g_1^1$) with a magnitude of $\theta_1^1 = 3$ and an exponential time profile occurs to agent 1 at $T_1 = 50$ second (i.e., $\beta_1(t - T_1) = 1 - e^{-(t-50)}$) and an actuator fault (i.e., $f_3^2 = \theta_3^2(t) g_3^2$) with a magnitude of $\theta_3^2(t) = -0.5 - 0.1 * \sin(t)$ occurs to agent 3 at 70 second, respectively. As can be seen from the top plots in Figures 2 and 3, the residual generated by the local FDE designed for agent 1 exceeds its threshold at approximately $t = 52.6$ second, and the residual generated by the local FDE designed for agent 3 exceeds its threshold a second after fault occurrence. Therefore, the process and actuator faults in agents 1 and 3 are timely detected, respectively. Once the faults are detected, the local fault isolation estimators are activated to determine the particular type of fault that has occurred. It can be seen from the second and third plots in Figure 2 that, for agent 1, the residual corresponding to the FIE associated with the first fault type always remains below the threshold, while the residual corresponding to the FIE associated with the second fault type exceeds the threshold at approximately $t = 59.4$ second. Similarly, it can be seen from the second and third plots in Figure 3 that, for agent 3, the residual corresponding to the FIE associated with the second fault type always remains below the threshold, while the residual corresponding to the FIE associated with the first fault type exceeds the

threshold at approximately $t = 77.5$ second. Thus, based on the fault isolation decision scheme described in Section 4.1, the occurrence of fault type 1 and fault type 2 can be concluded in agents 1 and 3, respectively.

After fault isolation, the neural-network-based adaptive fault-tolerant controller is reconfigured to accommodate the specific fault type that has been isolated. We set the learning rate $\Gamma_i = 20$ with a zero initial condition (see (58)). Regarding the performance of the adaptive fault-tolerant controllers, as can be seen from Figure 4(c) and Figure 5, the leader-following formation is achieved using the proposed adaptive FTCs even after fault occurrence, while the agents cannot achieve the leader-following formation and become unstable without the fault-tolerant controllers (see Figure 4(a)). Thus, the benefits of the FTC method can be clearly seen. Additionally, the tracking error of the agents is shown in Figure 4(b), when the second adaptive fault tolerant controller is not exploited. By comparing Figure 4(c) and Figure 4(b), it can be seen that the control performance is improved when the second adaptive FTC algorithm is exploited. It is also worth noting that the adaptive algorithm employed after fault isolation significantly reduces the computation complexity and is easier to implement.

6 Conclusions

In this paper, we investigate the problem of distributed FTC design for a class of second-order uncertain multi-agent systems under bidirectional intercommunication topology with possibly asymmetric weights. By using on-line diagnostic information, adaptive FTC controllers are developed to achieve the leader-following formation with a time-varying leader in the presence of faults. The closed-loop stability and leader-following formation properties at different stages of the fault diagnosis process are rigorously established, including the time-period between fault detection and isolation, and after fault isolation. The extensions to systems with more general structure is an interesting topic for future research.

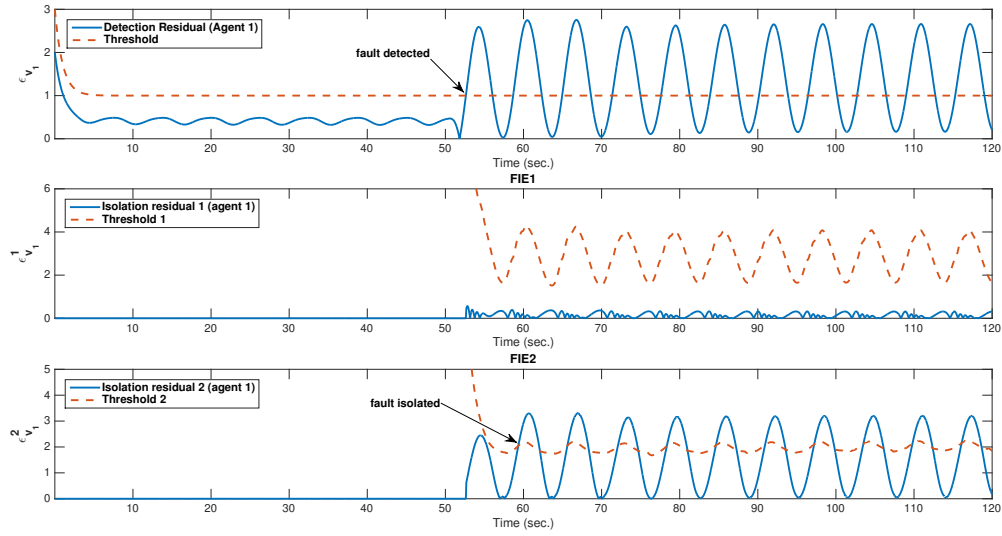


Figure 2: The fault detection and isolation residuals (solid blue line) and the corresponding thresholds (dashed red line) generated by the FDE and two FIEs of agent 1

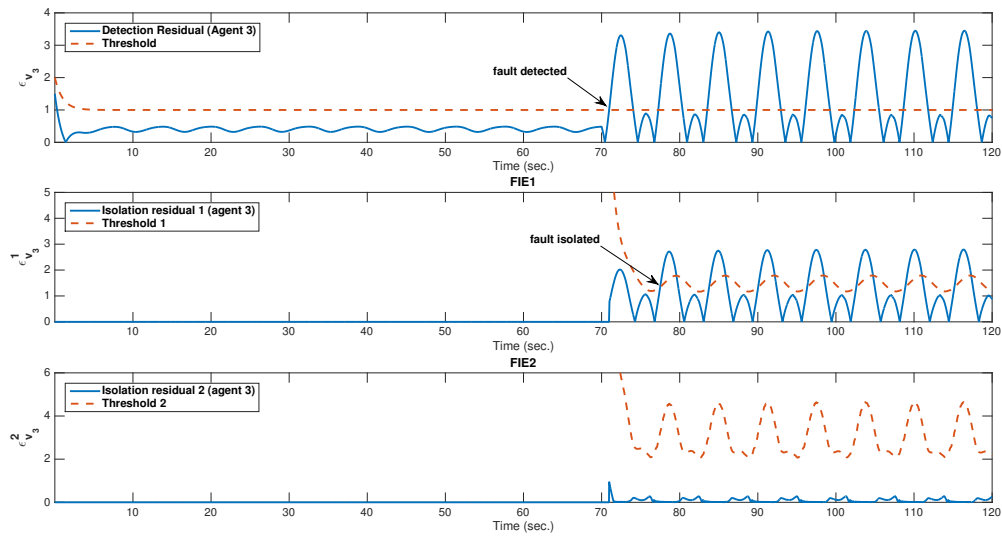


Figure 3: The fault detection and isolation residuals (solid blue line) and the corresponding thresholds (dashed red line) generated by the FDE and two FIEs of agent 3

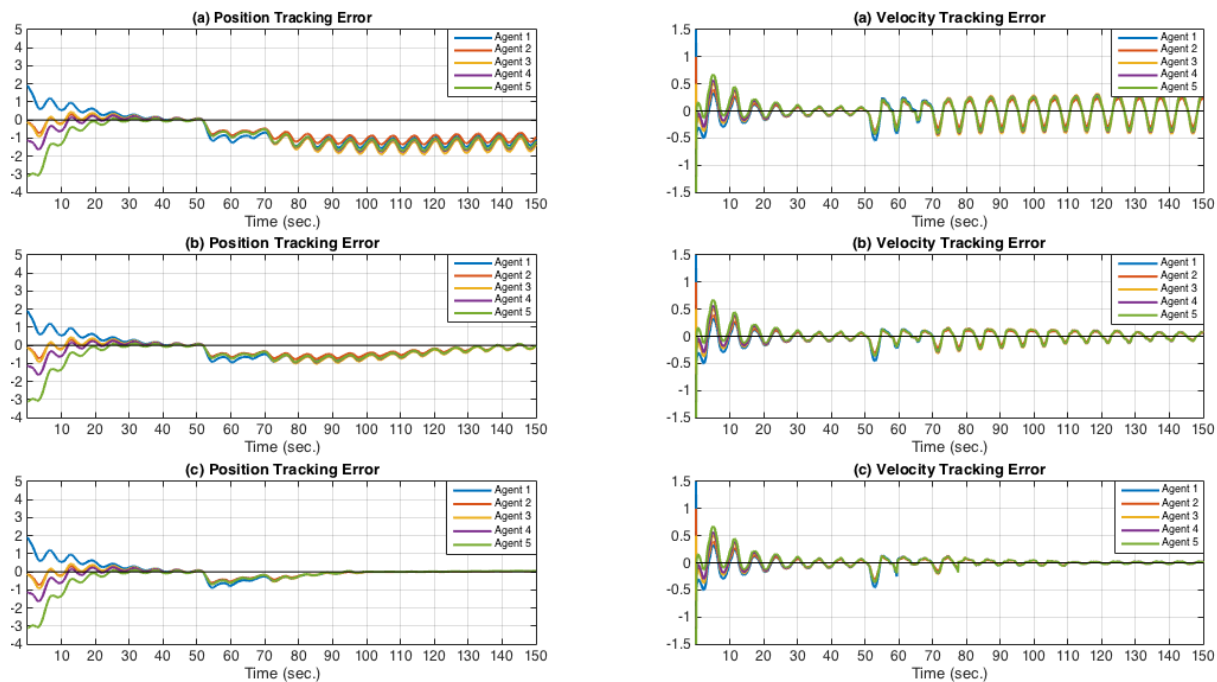


Figure 4: (a) Tracking errors without the adaptive FTC algorithms, (b) Tracking errors without the second adaptive FTC, (c) Tracking errors with both adaptive FTCs

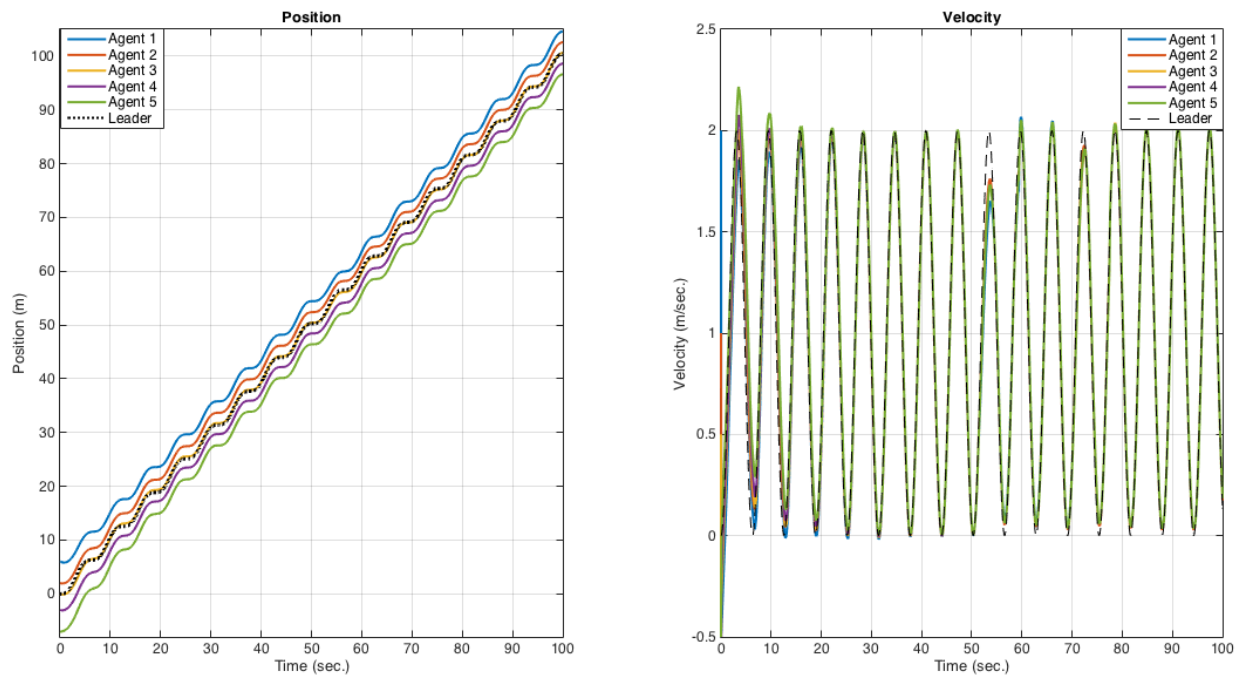


Figure 5: State trajectories with adaptive fault-tolerant controllers

Appendix A. Proof of Theorem 2

Proof: Based on (55) and (57), the closed-loop system dynamics are given by

$$\begin{aligned}\dot{p}_i &= v_i \\ \dot{v}_i &= - \sum_{j \in N_i} z_{ij} (\ell \tilde{p}_{ij} + \gamma \tilde{v}_{ij}) + \eta_i - (\bar{\eta}_i + \hat{\kappa}_i) \text{sgn}(\Xi_i) + (\tilde{\theta}_i^w)^T g_i^w(x_i) + (\beta_i - 1) \theta_i^T g_i^w(x_i) - \psi_i^p.\end{aligned}$$

We can represent the collective tracking error dynamics as

$$\dot{\tilde{x}} = A\tilde{x} + \begin{bmatrix} \mathbf{0}_M \\ \zeta - \bar{\zeta} - \mathbf{1}_M \dot{v}_0 + \tilde{f}^p + \Delta^p - \psi^p \end{bmatrix}, \quad (69)$$

where A is given in Lemma 3, $\tilde{x} = [\tilde{p}^T, \tilde{v}^T]^T$ is defined in a similar way as in (32), the terms ζ and $\bar{\zeta}$ are defined in (33) and (34), and $\tilde{f}^p \in \mathfrak{R}^M$, $\Delta^p \in \mathfrak{R}^M$ and $\psi^p \in \mathfrak{R}^M$ are defined as

$$\tilde{f}^p \triangleq [(\tilde{\theta}_1^w)^T g_1^w, \dots, (\tilde{\theta}_M^w)^T g_M^w]^T, \quad (70)$$

$$\Delta^p \triangleq [\Delta_1^p, \dots, \Delta_M^p]^T, \quad (71)$$

$$\psi^p \triangleq [\psi_1^p, \dots, \psi_M^p]^T, \quad (72)$$

and $\tilde{\theta}_i^w = \theta_i^w - \hat{\theta}_i^w$ is the fault parameter estimation error corresponding to the i th agent, $\Delta_i^p \triangleq (\beta_i - 1)(\theta_i^w)^T g_i^w(x_i)$, and ψ_i^p is defined in (59). We consider the following Lyapunov function candidate:

$$V = \tilde{x}^T P \tilde{x} + (\tilde{\theta}^w)^T (\Gamma)^{-1} \tilde{\theta}^w + (\tilde{\alpha}^p)^T (\Upsilon)^{-1} \tilde{\alpha}^p + \tilde{\kappa}^T (\tilde{\Upsilon})^{-1} \tilde{\kappa}, \quad (73)$$

where P is defined in Lemma 3, $\tilde{\theta}^w = [(\tilde{\theta}_1^w)^T, \dots, (\tilde{\theta}_M^w)^T]^T$ is the collective parameter estimation errors, and $\tilde{\alpha}^p = [\tilde{\alpha}_1^p, \dots, \tilde{\alpha}_M^p]^T$ is the bounding collective parameter estimation errors defined as $\tilde{\alpha}_i^p = \alpha_{m_i}^p - \hat{\alpha}_i^p$. Then, using (41), (42), (43), and a similar reasoning logic for (70), (71) and (72), the time derivative of the Lyapunov function (73) along the solution of (69) is given by

$$\begin{aligned}\dot{V} &= \tilde{x}^T Q \tilde{x} + 2 \sum_{i=1}^M \left[\sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) (\eta_i - \dot{v}_0) - \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) (\bar{\eta}_i + \hat{\kappa}_i) \text{sgn}(\Xi_i) - \tilde{\kappa}_i^T (\tilde{\Upsilon}_i)^{-1} \dot{\hat{\kappa}}_i \right. \\ &\quad \left. + (\tilde{\theta}_i^w)^T \left(\sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) g_i^w - (\Gamma_i)^{-1} \dot{\hat{\theta}}_i^w \right) + \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) (\Delta_{p_i} - \psi_i^p) - \tilde{\alpha}_i^p (\Upsilon_i)^{-1} \dot{\hat{\alpha}}_i^p \right],\end{aligned}$$

where Q is defined in Lemma 3. Therefore, using (49) and choosing the adaptive laws as (58) and (60), we have $\dot{V} \leq \tilde{x}^T Q \tilde{x}$. Then, the proof can be concluded by using a similar reasoning logic as reported in the proof of Theorem 1. \square

Appendix B. Proof of Theorem 3

Proof: Using some algebraic manipulations, we can rewrite (63) as $u_i = \bar{\tau}_i - \bar{\tau}_{\theta_i} u_i$. Therefore, by substituting u_i in (62), the closed-loop system dynamics are given by

$$\begin{aligned}\dot{p}_i &= v_i \\ \dot{v}_i &= - \sum_{j \in N_i} z_{ij} (\ell \tilde{p}_{ij} + \gamma \tilde{v}_{ij}) + \eta_i - (\bar{\eta}_i + \hat{\kappa}_i) \text{sgn}(\Xi_i) + (\theta_i(t) - \bar{\tau}_{\theta_i}) u_i.\end{aligned}$$

We can represent the collective tracking error dynamics as

$$\dot{\tilde{x}} = A\tilde{x} + \begin{bmatrix} \mathbf{0}_M \\ \zeta - \bar{\zeta} - \mathbf{1}_M \dot{v}_0 + \varpi \end{bmatrix}, \quad (74)$$

where A is given in Lemma 3, $\tilde{x} = [\tilde{p}^T, \tilde{v}^T]^T$ is defined in a similar way as in (32), and the terms ζ , $\bar{\zeta}$, and ϖ are defined in (33), (34) and (38), respectively. We consider the following Lyapunov function candidate:

$$V = \tilde{x}^T P \tilde{x} + \tilde{\theta}^T (\Gamma)^{-1} \tilde{\theta} + \tilde{\kappa}^T (\tilde{\Upsilon})^{-1} \tilde{\kappa}, \quad (75)$$

where P is defined in Lemma 3, and $\tilde{\theta} = [\tilde{\theta}_1, \dots, \tilde{\theta}_M]^T$ is the collective estimation errors of actuator fault parameter lower bound defined as $\tilde{\theta}_i = \theta_i^* - \hat{\theta}_i$. Then, using (41), (42), (43), and (47), the time derivative of the Lyapunov function (75) along the solution of (74) is given by

$$\begin{aligned} \dot{V} = & \tilde{x}^T Q \tilde{x} + 2 \sum_{i=1}^M \left[\sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) (\eta_i - \dot{v}_0) - \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) (\bar{\eta}_i + \hat{\kappa}_i) \operatorname{sgn}(\Xi_i) - \tilde{\kappa}_i^T (\tilde{\Upsilon}_i)^{-1} \dot{\hat{\kappa}}_i \right. \\ & \left. + \sum_{j \in N_i} z_{ij} (\epsilon \tilde{p}_{ij} + \rho \tilde{v}_{ij}) (\theta_i(t) - \bar{\tau}_{\theta_i}) u_i - \tilde{\theta}_i (\bar{\Gamma}_i)^{-1} \dot{\hat{\theta}}_i \right], \end{aligned}$$

where Q is defined in Lemma 3. Therefore, using (49) and choosing the adaptive laws as (66) – (67), we have $\dot{V} \leq \tilde{x}^T Q \tilde{x}$. Then, the proof can be concluded by using a similar reasoning logic as reported in the proof of Theorem 1. \square

References

- [1] Z. Qu, *Cooperative Control of Dynamical Systems: Applications to Autonomous Vehicles*. New York: Springer-Verlag, 2009.
- [2] W. Ren and R. Beard, *Distributed Consensus in Multi-vehicle Cooperative Control: Theory and Applications*. London: Springer-Verlag, 2008.
- [3] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods for Multiagent Networks*. Princeton, NJ: Princeton University Press, 2010.
- [4] Y. Hu, H. Su, and J. Lam, “Adaptive consensus with a virtual leader of multiple agents governed by locally lipschitz nonlinearity,” *International Journal of Robust and Nonlinear Control*, vol. 23, no. 9, pp. 978–990, 2013.
- [5] A. Das and F. L. Lewis, “Cooperative adaptive control for synchronization of second-order systems with unknown nonlinearities,” *International Journal of Robust and Nonlinear Control*, vol. 21, no. 13, pp. 1509–1524, 2011.
- [6] L. Cheng, Z. Hou, M. Tan, Y. Lin, and W. Zhang, “Neural-network-based adaptive leader-following control for multi-agent systems with uncertainties,” *IEEE Transactions on Neural Networks*, vol. 21, pp. 1351–1358, August 2010.
- [7] X. Yu, Z. Liu, and Y. Zhang, “Fault-tolerant formation control of multiple UAVs in the presence of actuator faults,” *International Journal of Robust and Nonlinear Control*, vol. 26, pp. 2668–2685, 2016.
- [8] E. Semsar-Kazerooni and K. Khorasani, “Team consensus for a network of unmanned vehicles in presence of actuator faults,” *IEEE Transactions on Control Systems Technology*, vol. 18, no. 5, pp. 1155–1161, 2010.
- [9] J. Li, “Fault tolerant consensus of multi-agent systems with linear dynamics,” *Mathematical Problems in Engineering*, vol. 1, pp. 465–471, January 2013.

- [10] Z. Zuo, J. Zhang, and Y. Wang, “Adaptive fault-tolerant tracking control for linear and lipschitz nonlinear multi-agent systems,” *IEEE Transactions on Industrial Electronics*, vol. 62, pp. 3923–3931, June 2015.
- [11] X. Wang and G. H. Yang, “Distributed reliable H_∞ consensus control for a class of multi-agent systems under switching networks: A topology-based average dwell time approach,” *International Journal of Robust and Nonlinear Control*, vol. 26, pp. 2767–2787, 2016.
- [12] L. Zhao and Y. Jia, “Neural network-based adaptive consensus tracking control for multi-agent systems under actuator faults,” *International Journal of Systems Science*, 2014.
- [13] C. Deng and G. Yang, “Distributed adaptive fault-tolerant containment control for a class of multi-agent systems with non-identical matching non-linear functions,” *IET Control Theory & Applications*, vol. 10, pp. 273–281, 2016.
- [14] D. Ye, X. Zhao, and B. Cao, “Distributed adaptive fault-tolerant consensus tracking of multi-agent systems against time-varying actuator faults,” *IET Control Theory & Applications*, vol. 10, pp. 554–563, 2016.
- [15] L. B. Wu, X. Q. He, and D. Q. Zhang, “Cooperative adaptive fuzzy control for a class of uncertain non-linear multi-agent systems with time delays,” *Journal of Control and Decision*, vol. 4, pp. 131–152, 2017.
- [16] Q. Shen, B. Jiang, P. Shi, and J. Zhao, “Cooperative adaptive fuzzy tracking control for networked unknown nonlinear multiagent systems with time-varying actuator faults,” *IEEE Transactions on Fuzzy Systems*, vol. 22, pp. 494–504, June 2013.
- [17] W. Wang, D. Wang, and Z. Peng, “Fault-tolerant containment control of uncertain nonlinear systems in strict-feedback form,” *International Journal of Robust and Nonlinear Control*, vol. 27, pp. 497–511, 2017.
- [18] G. Chen, F. L. Lewis, and L. Xie, “Finite-time distributed consensus via binary control protocols,” *Automatica*, vol. 47, pp. 1962–1968, 2011.
- [19] Y. Zhang, Y. Yang, Y. Zhao, and G. Wen, “Distributed finite-time tracking control for nonlinear multi-agent systems subject to external disturbances,” *International Journal of Control*, vol. 86, no. 1, pp. 29–40, 2013.
- [20] J. Farrell and M. M. Polycarpou, *Adaptive Approximation Based Control*. Hoboken, NJ: J. Wiley, 2006.
- [21] M. Khalili, X. Zhang, M. M. Polycarpou, T. Parisini, and Y. Cao, “Distributed adaptive fault-tolerant control of uncertain multi-agent systems,” in *Proceedings of the 9th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes*, pp. 66–71, September 2015. (an extended journal version has been accepted for publication in *Automatica*).
- [22] M. Khalili, X. Zhang, Y. Cao, M. M. Polycarpou, and T. Parisini, “Distributed adaptive fault-tolerant control of nonlinear uncertain second-order multi-agent systems,” in *Proceedings of the 54th IEEE Conference on Decision and Control*, pp. 4480–4485, December 2015.
- [23] X. Zhang, M. M. Polycarpou, and T. Parisini, “A robust detection and isolation scheme for abrupt and incipient faults in nonlinear systems,” *IEEE Transactions on Automatic Control*, vol. 47, pp. 576–593, 2002.

- [24] A. Emami-Naeini, M. M. Akhter, and S. M. Rock, “Effect of model uncertainty on failure detection: the threshold selector,” *IEEE Transactions on Automatic Control*, vol. 33, pp. 1106–1115, 1988.
- [25] X. Zhang, L. Tang, and J. Decastro, “A nonlinear adaptive estimation based approach to robust fault diagnosis of aircraft engines,” *IEEE Transactions on Control Systems Technology*, vol. 21, pp. 861–868, May 2013.
- [26] H. Zhang and F. L. Lewis, “Adaptive cooperative tracking control of higher-order nonlinear systems with unknown dynamics,” *Automatica*, vol. 48, pp. 1432–1439, 2012.
- [27] X. Zhang, T. Parisini, and M. M. Polycarpou, “Adaptive fault-tolerant control of nonlinear systems: a diagnostic information-based approach,” *IEEE Transactions on Automatic Control*, vol. 49, pp. 1259–1274, August 2004.
- [28] A. Fekih, “Fault-tolerant flight control design for effective and reliable aircraft systems,” *Journal of Control and Decision*, vol. 1, pp. 299–316, 2014.
- [29] M. Liu, L. Zhang, P. Shi, and Y. Zhao, “Sliding mode control of continuous-time markovian jump systems with digital data transmission,” *Automatica*, vol. 80, pp. 200–209, June 2017.
- [30] D. Ye, L. Su, J. L. Wang, and Y. N. Pan, “Adaptive reliable H_∞ optimization control for linear systems with time-varying actuator fault and delays,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, pp. 1635–1643, July 2017.
- [31] G. Chen and Y. D. Song, “Cooperative tracking control of nonlinear multiagent systems using self-structuring neural networks,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, pp. 1496–1507, August 2014.
- [32] H. Zhang, F. L. Lewis, and Z. Qu, “Lyapunov, adaptive, and optimal design techniques for cooperative systems on directed communication graphs,” *IEEE Transactions on Industrial Electronics*, vol. 59, no. 7, pp. 3026–3041, 2012.
- [33] C. T. Chen, *Linear System Theory and Design*. New York, NY: Oxford University Press, 1999.
- [34] Y. Cao and W. Ren, “Distributed coordinated tracking with reduced interaction via a variable structure approach,” *IEEE Transactions on Automatic Control*, vol. 57, pp. 33–48, January 2012.
- [35] P. A. Ioannou and J. Sun, *Robust Adaptive Control*. Englewood Cliffs, NJ: Prentice Hall, 1996.
- [36] Z. Li, X. Liu, W. Ren, and L. Xie, “Distributed tracking control for linear multiagent systems with a leader of bounded unknown input,” *IEEE Transactions on Automatic Control*, vol. 58, no. 2, pp. 518–523, 2013.
- [37] J. T. Spooner and K. M. Passino, “Adaptive control of a class of decentralized nonlinear systems,” *IEEE Transactions on Automatic Control*, vol. 41, no. 2, pp. 280–284, 1996.