Distributed Fault-Tolerant Control of Multi-Agent Systems: An Adaptive Learning Approach

Mohsen Khalili, Xiaodong Zhang, Yongcan Cao, Marios M. Polycarpou, and Thomas Parisini

***Abstract*—This paper focuses on developing a distributed leader-following fault-tolerant tracking control scheme for a class of high-order nonlinear uncertain multi-agent systems. Neural network based adaptive learning algorithms are developed to learn unknown fault functions, guaranteeing the system stability and cooperative tracking even in the presence of multiple simultaneous process and actuator faults in the distributed agents. The time-varying leader's command is only communicated to a small portion of follower agents through directed links, and each follower agent exchanges local measurement information only with its neighbors through a bidirectional but asymmetric topology. Adaptive fault-tolerant algorithms are developed for two cases, i.e., with full-state measurement and with only limited output measurement, respectively. Under certain assumptions, the closed-loop stability and asymptotic leader-follower tracking properties are rigorously established.**

***Index Terms*—Fault-Tolerant Control, Learning Systems, Multi-Agent Systems, Cooperative Tracking, Nonlinear Uncertain Systems.**

I. INTRODUCTION

Cooperative control of multi-agent systems (MAS) using distributed consensus algorithms has recently 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 the initial conditions of the agents. For the tracking problem, there is a leader agent acting as a command generator, and all follower agents are synchronized to track the leader, despite the leader's command being received only by a small portion of followers. Since such distributed MAS need to operate reliably even in the presence of faults in some agents, the development of fault-tolerant control (FTC) schemes is a crucial step in achieving dependable and safe operations.

M. Khalili and X. Zhang are with the Department of Electrical Engineering, Wright State University, Dayton, OH 45435, USA (e-mail: khalili.4@wright.edu, xiaodong.zhang@wright.edu).

Y. Cao is with the Department of Electrical and Computer Engineering, University of Texas, San Antonio, TX 78249, USA (e-mail: yongcan.cao@utsa.edu).

M. Polycarpou is with the KIOS Research and Innovation Center of Excellence and the Department of Electrical and Computer Engineering, University of Cyprus, Nicosia 1678, Cyprus (e-mail: mpolycar@ucy.ac.cy).

T. Parisini is with the Department of Electrical and Electronic Engineering, Imperial College London, UK, with the KIOS Research and Innovation Center of Excellence, Cyprus, and also with the Dept. of Engineering and Architecture, University of Trieste, Italy (e-mail: t.parisini@gmail.com).

This work has been partially supported by the European Union's Horizon 2020 Research and Innovation Programme under grant agreement No. 739551 (KIOS CoE).

Adaptive approximators such as neural networks combined with learning estimation algorithms are suitable for approximating unknown nonlinear dynamics or fault functions in the fault-tolerant control design, thereby enhancing the robustness and detectability of the fault diagnosis scheme, as well as improving the capability for fault accommodation. So far, considerable effort has focused on the development of adaptive-approximator-based leader-following tracking algorithms for MAS with first-order agent dynamics [4], [5], second-order dynamics [6]–[8], and high-order dynamics in the Brunovsky form [9], [10]. On the other hand, limited results are available on leader-following FTC design for MAS with more general dynamics. For instance, leader-following tracking algorithms for MAS with general linear and Lipschitz nonlinear dynamics have been developed by assuming the absence of faults [11] or by considering actuator faults [12]–[14]. However, the agent models in [12]–[14] do not consider process faults, which are crucial for the safe operation of MAS. Furthermore, the results in [11]–[14] are based on a critical assumption that the Laplacian matrix of the communication graph is symmetric. In practice, the distributed leader-follower FTC problem naturally requires the consideration of graphs with an *asymmetric* Laplacian matrix, which is significantly more challenging.

This paper presents a distributed adaptive cooperative tracking FTC method for accommodating both process and actuator faults in a class of high-order nonlinear uncertain multi-agent systems. Neural network based adaptive approximators are employed in the FTC design, to learn the unknown fault function and to guarantee the system stability and leader-following performance in the presence of faults by modifying the feedback control law via parameter adaptation. In the leader-following topology under consideration, the time-varying leader's command is only communicated to a small subset of follower agents, and each follower agent exchanges measurement information only with its neighbors through a bidirectional but *asymmetric* interconnection topology. It is worth noting that the asymmetric weights of the graph under consideration do not assume the critical detail-balanced condition considered in the literature [15], [16], which significantly increases the complexity of FTC design for achieving asymptotic leader-following tracking in the presence of faults. For instance, the stability analysis methods in [11]–[14], which utilize the symmetric property of the Laplacian matrix for undirected graphs, are not applicable. Note that undirected graphs and the graphs satisfying detail-balanced condition are special cases of the intercommunication graph considered in this paper.

Distributed adaptive FTC schemes for *first-order* and *second-order* multi-agent systems under asymmetric graph were previously presented in [17] and [18], respectively. This paper significantly extends these results by considering more general high-order agent dynamics under bidirectional but *asymmetric* communication links among the followers. Specifically, the FTC problem under consideration is investigated for two cases: (i) with full-state measurement and (ii) with only limited output measurements. Appropriately chosen Lyapunov functions are presented to circumvent the technical difficulty in the design and analysis of the adaptive learning scheme. The proposed fault-tolerant cooperative tracking algorithms are developed to achieve asymptotic leader-following tracking performance even in the presence of multiple process and actuator faults in distributed agents. The recent conference paper [19] describes the full-state measurement case with detailed proof omitted and does not consider the case of the limited output measurements presented in this paper.

The rest of the paper is organized as follows. Fault-tolerant leader-follower consensus control of multi-agent systems with full-state measurements is described in Section II. The closed-loop stability and consensus performance for multi-agent systems with partial state measurements is investigated in Section III. In Section IV, simulation examples are used to illustrate the effectiveness of the FTC method. Finally, Section V provides some concluding remarks.

II. Distributed FTC: Full-State Measurements

A. *Distributed Multi-Agent Systems*

In this paper, the overall leader-follower system including the leader is represented by the graph $\mathcal{G}$, which has a fixed communication topology that is bidirectional but asymmetric among followers. An example of the distributed FTC architecture under consideration is shown in Figure 1. As it is shown, the leader's command (i.e., the state of node 0) is only communicated to a small subset of follower agents (only agent 2 in this example), and each follower agent exchanges local measurement information only with its neighbors through an asymmetric bidirectional interconnection topology. It is assumed that the leader has a directed path to all followers, which ensures that the information exchange among agents is sufficient for the team to achieve the desired team goal. For instance, this condition is required to exclude isolated followers, as described in [9]. Neural network based adaptive approximators are employed by the FTC component of each agent to learn the unknown fault function. The learned fault function information is utilized by each agent in the control law (as well as the state estimator in the case of partial state measurement) to guarantee that the state of each agent tracks the state of the time-varying leader via parameter adaptation.

Process faults occur due to any undesirable changes in the behaviors of the system components and therefore affect the dynamics of the system, whereas actuator faults represent the discrepancies between the input command of the actuators and their actual output. Most practical process faults are nonlinear functions of the system state. For example, a leakage fault in a thermal system or a chemical process is, in general, a nonlinear

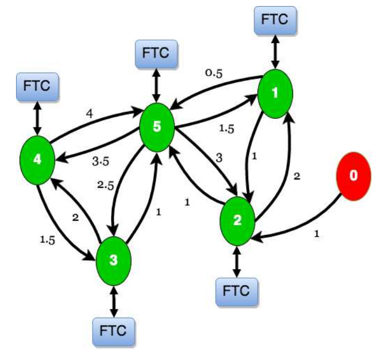

Fig. 1. An example of the distributed FTC architecture

function of the pressure and the temperature. Specifically, we consider a set of M agents where the dynamics of the ith agent, $i = 1, \cdots, M$, is described by the following differential equation

$$\begin{aligned}\dot{x}_i &= Ax_i + g(x_i) + Bu_i + \beta_i(t - T_{iu})B\theta_i u_i \\ &\quad + D\eta_i(x_i, t) + \beta_i(t - T_{if})Ff_i(x_i)\end{aligned} \tag{1}$$

where $x_i \in \Re^n$ and $u_i \in \Re^m$ are the state and input vector of the ith agent, respectively. Additionally, $g : \Re^n \mapsto \Re^n$, $\eta_i : \Re^n \times \Re^+ \mapsto \Re^v$, and $f_i : \Re^n \mapsto \Re^s$ are smooth vector fields, A, B, D, and F are matrices with appropriate dimensions, and the pair (A, B) is stabilizable. Specifically, the vector fields g and η_i are the known nominal nonlinearity and unknown modeling uncertainty in the "healthy" dynamics (in the absence of faults) of the ith agent, respectively.

The terms $\beta_i(t - T_{if})Ff_i(x_i)$ and $\beta_i(t - T_{iu})B\theta_i u_i$ in (1) represent the changes in the dynamics of ith agent due to the occurrence of process and actuator faults, respectively. Specifically, $\beta_i(t - T_{if})$ and $\beta_i(t - T_{iu})$ are the time profiles of process and actuator faults which occur at some unknown time T_{if} and T_{iu}, respectively. Furthermore, $f_i(x_i)$ is the unknown process fault function, and $\theta_i u_i$ is an actuator fault represented by partial loss of effectiveness of the actuators. Specifically, the matrix $\theta_i \triangleq \mathrm{diag}\{\theta_{i1}, \cdots, \theta_{im}\}$, where each unknown constant $\theta_{id} \in [\bar{\theta}_{id}, 0]$ characterizes the actuator fault parameter associated with actuator u_{id}, for $d = 1, \cdots, m$. The case of $\theta_{id} = 0$ corresponds to a healthy actuator, while the case of $\bar{\theta}_{id} \leq \theta_{id} < 0$ implies the actuator is partially faulty, where the constant $\bar{\theta}_{id} \in (-1, 0)$ is a lower bound chosen to maintain the controllability of the distributed agents. 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., $t < T_{if}$ for process faults or $t < T_{iu}$ for actuator faults), and satisfies $0 < \beta_i \leq 1$ after fault occurrence (i.e., $t \geq T_{if}$ for process faults or $t \geq T_{iu}$ for actuator faults). For instance, the time profile of abrupt faults can be modeled as a step function, and the time profile of incipient (slowly developing) faults can be modeled as a drift-type fault or an exponential term with an unknown fault-evolution rate [20]. Therefore, both incipient and abrupt faults are considered in this paper. Note that the system model (1) allows the occurrence of multiple simultaneous faults in multiple agents.

Without loss of generality, let the leader be identified as agent number 0 with a reference state $x_0 \in \Re^n$ satisfying $\dot{x}_0(t) = Ax_0 + Bu_0 + g(x_0)$, where $u_0 \in \Re^m$ is the control input. Then, the following assumptions are made:

Assumption 1.1: The leader's state x_0 follows a bounded unknown reference trajectory, and the control input u_0 is bounded by an unknown constant.

Assumption 1.2: The modeling uncertainty of the "healthy" agents, represented by $\eta_i(x_i, t)$ in (1), satisfies

$$|\eta_i(x_i,t)| \leq \omega_i \, \bar{\eta}_i(x_i,t)\,, \quad \forall x_i \in \Re^n \tag{2}$$

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

Assumption 1.3: The nominal nonlinearity $g(x_i)$ in (1) satisfies the following condition: for all $x_i \in \Re^n$ and $\hat{x}_i \in \Re^n$,

$$|g(x_i) - g(\hat{x}_i)| \leq \sigma|x_i - \hat{x}_i|\,, \tag{3}$$

where σ is a known constant.

Assumption 1.4: The matrices D and F lie within the range space of the matrix B, which implies there exist matrices $\bar{D}$ and $\bar{F}$ with appropriate dimensions such that

$$B\bar{D} = D \text{ and } B\bar{F} = F\,. \tag{4}$$

As in [9] and [21], Assumption 1.1 is needed to achieve cooperative tracking control for a time-varying leader. It is worth noting that the bound on the uncertainty given in Assumption 1.2 is allowed to be a function of the agent state x_i, which is less restrictive than the constant bounds assumed in [8]–[10]. The condition given in Assumption 1.3 on the nominal nonlinearity $g(x_i)$ is needed for FTC design as in [11], [12] and [22]. Note that $\eta_i(x_i, t)$ and $f_i(x_i)$ are unknown. Assumption 1.4 provides the so-called *matched uncertainty* condition [23], required to ensure the robustness of the FTC algorithms with respect to faults and modeling uncertainty.

Remark 1: The distributed nonlinear agent model described by (1) is more general than the high-order agent models considered in the literature. For instance, a class of Lipschitz nonlinear agents were considered in [12], where the absence of modeling uncertainty and process faults were assumed. The agent model considered in [9] and [10] is assumed to be in the Brunovsky form, where $g(x_i) = 0$ and $B = D = F = [0, \cdots, 0, 1]^T$. Additionally, agents with linear dynamics and matching uncertainties (i.e., $g = 0$, $D = F = B$) were considered in [13] under the assumption of undirected graphs, while unbalanced graphs are considered in this paper. It is worth noting that all these aforementioned high-order agent models in the literature satisfy Assumptions 1.1–1.4 described above.

B. Distributed Fault-Tolerant Control Design

Adaptive approximators such as neural-network models can be used to approximate the unknown process fault function $f_i(x_i)$. Specifically, we consider linearly parametrized network (e.g., radial-basis-function networks with fixed centers and variances) described as follows:

$$\hat{f}_i(x_i, \hat{\vartheta}_i) = \hat{\vartheta}_i^T \varphi_i(x_i)\,, \tag{5}$$

where $\varphi_i(\cdot)$ represents the collective vector of fixed basis functions, and $\hat{\vartheta}_i$ are the adjustable weights of the nonlinear approximator. In the presence of a process fault, $\hat{f}_i$ provides the adaptive structure for approximating the unknown fault function $f_i(x_i)$ by adapting the weight vector $\hat{\vartheta}_i(t)$.

In the presence of process and actuator faults, by adding and subtracting the term $\hat{f}_i(x_i, \vartheta_i)$, the system dynamics described by (1) can be rewritten as

$$\begin{aligned}\dot{x}_i &= Ax_i + g(x_i) + Bu_i + B\beta_i\theta_i u_i + D\eta_i(x_i,t) \\ &\quad + F\left[\hat{f}_i(x_i,\vartheta_i) + \beta_i f_i(x_i) - \hat{f}_i(x_i,\vartheta_i)\right],\end{aligned}$$

where ϑ_i is the *unknown* optimal weight matrix [24]. By defining the residual approximation error for the ith agent as $\delta_i \triangleq f_i(x_i) - \hat{f}_i(x_i, \vartheta_i)$, we have

$$\begin{aligned}\dot{x}_i &= Ax_i + g(x_i) + Bu_i + B\beta_i\theta_i u_i + D\eta_i(x_i,t) \\ &\quad + F\left[\hat{f}_i(x_i,\vartheta_i) + (\beta_i - 1)\hat{f}_i(x_i,\vartheta_i) + \beta_i\delta_i(x_i)\right].\end{aligned} \tag{6}$$

For each network, the following assumption is made:

Assumption 1.5: for each $i = 1, \cdots, M$, the neural network residual approximation error satisfies

$$|\delta_i(x_i)| \leq \alpha_i \bar{\delta}_i(x_i)\,, \quad \forall x_i \in \Re^n \tag{7}$$

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

Remark 2: It is worth noting that the bound on the residual approximation error in the above assumption is allowed to be a function of the agent state, which is less restrictive than the constant bound assumed in [9], [10], [13], and [22]. The bounding functions $\bar{\delta}_i$ can possibly be obtained by making use of certain limited knowledge on the fault. In the worst case scenario when there is no information about the bound on the residual approximation error, the bound can be considered as an unknown constant (i.e., $\bar{\delta}_i = 1$). Furthermore, based on the above assumption, the residual error (which may grow unbounded as x_i goes to infinity) is bounded by a known function $\bar{\delta}_i$, which in general will grow unbounded as x_i becomes very large (for example the known function $\bar{\delta}_i$ may have terms like x_i^2). The term will appear in the FTC and will not allow the trajectory to become unbounded due to the high gain. On the other hand, as x_i becomes quite large the controller output will also become very large possibly reaching saturation or it may encounter other problems in the presence of measurement noise.

We let α_i^0 and κ_i represent unknown constants defined as

$$\alpha_i^0 \triangleq \sup_{t \geq T_{if}} \max\left\{\left|\beta_i(t - T_{if})\,\alpha_i\right|, \left|\left[\beta_i(t - T_{if}) - 1\right]\vartheta_i\right|\right\}, \tag{8}$$

$$\kappa_i \triangleq \sup_{t \geq 0} \max\left\{\left|u_0 + Kx_0\right|, \left|\left[\beta_i(t - T_{iu}) - 1\right]\theta_i\right|\right\}, \tag{9}$$

where $K \in \Re^{m \times n}$ is a design gain matrix. Note that the fault time profile β_i satisfies $0 \leq \beta_i \leq 1$. Then, based on Assumptions 1.1 and 1.5, the finite constants α_i^0 and κ_i, defined respectively by (8) and (9) always exist.

Let N_i denote the set of neighbors of agent i. Based on the system model (6) and the neural network model (5), the following fault-tolerant controller for the ith agent is chosen:

$$u_i = (I_m + \hat{\theta}_i)^{-1}\bar{u}_i \tag{10}$$

$$\begin{aligned}\bar{u}_i \triangleq& -\rho_i B^T P \sum_{j\in N_i} b_{ij}\tilde{x}_{ij} - \hat{\kappa}_i \bar{U}_i sgn\Big(\sum_{j\in N_i} b_{ij}B^T P\,\tilde{x}_{ij}\Big)\\ &-Kx_i - \bar{D}\,\hat{\omega}_i\,\bar{\eta}_i sgn\Big(\sum_{j\in N_i} b_{ij}D^T P\,\tilde{x}_{ij}\Big)\\ &-\bar{F}\Big[\hat{\alpha}_i\bar{\Delta}_i(x_i)sgn\Big(\sum_{j\in N_i} b_{ij}F^T P\,\tilde{x}_{ij}\Big)+\hat{f}_i(x_i,\hat{\vartheta}_i)\Big]\end{aligned} \tag{11}$$

$$\dot{\rho}_i = \bar{\Lambda}_i\left(\sum_{j\in N_i} b_{ij}B^T P\,\tilde{x}_{ij}\right)^T\left(\sum_{j\in N_i} b_{ij}B^T P\,\tilde{x}_{ij}\right) \tag{12}$$

$$\dot{\hat{\vartheta}}_{ih} = \Gamma_i\left(\sum_{j\in N_i} b_{ij}F_h^T P\,\tilde{x}_{ij}\right)\varphi_i(x_i) \tag{13}$$

$$\dot{\hat{\alpha}}_i = \Upsilon_i\left|\sum_{j\in N_i} b_{ij}\,F^T P\,\tilde{x}_{ij}\right|\bar{\Delta}_i(x_i) \tag{14}$$

$$\dot{\hat{\theta}}_{id} = \mathcal{P}_{\bar{\theta}_{id}}\left\{\bar{\Gamma}_i\sum_{j\in N_i} b_{ij}B_d^T P\,\tilde{x}_{ij}u_{id}\right\} \tag{15}$$

$$\dot{\hat{\kappa}}_i = \bar{\Upsilon}_i\left|\sum_{j\in N_i} b_{ij}B^T P\,\tilde{x}_{ij}\right|\bar{U}_i \tag{16}$$

$$\dot{\hat{\omega}}_i = \Lambda_i\left|\sum_{j\in N_i} b_{ij}D^T P\,\tilde{x}_{ij}\right|\bar{\eta}_i\,, \tag{17}$$

where $\rho_i(t)$ is a time-varying coupling gain, $\hat{\theta}_i \triangleq \mathrm{diag}\{\hat{\theta}_{i1},\cdots,\hat{\theta}_{im}\}$ with $\hat{\theta}_{id}$ in (15) being an estimation of the actuator fault parameter θ_{id}, for $d = 1,\cdots,m$, the projection operator $\mathcal{P}$ restricts $\hat{\theta}_{id}$ to the corresponding set $[\bar{\theta}_{id}, 0]$, B_d is the dth column of matrix B, u_{id} is the dth component of control input u_i, I_m is a $m\times m$ identity matrix, $\tilde{x}_{ij} \triangleq x_i - x_j$, for $j \in N_i$, b_{ij} are constant design gains defined later in (23), $K \in \Re^{m\times n}$ is a design gain matrix chosen to make $\tilde{A} \triangleq A - BK$ Hurwitz, $\hat{\kappa}_i$ is an estimation of the unknown positive constant bound κ_i described in (9), $\bar{D}$ and $\bar{F}$ are given in Assumption 1.4, $\hat{f}_i$ provides the adaptive approximation of unknown process fault functions (see (5)), $\hat{\vartheta}_{ih}$ is an estimation of the hth column of the neural network optimal weight matrix ϑ_i, for $h = 1,\cdots,s$, F_h is the hth column of matrix F, $\bar{\Delta}_i(x_i) \triangleq \bar{\delta}_i(x_i) + |\varphi_i(x_i)|$, $\bar{U}_i \triangleq 1 + |u_i|$, $\hat{\alpha}_i$ and $\hat{\omega}_i$ are estimates of the unknown bounding constants α_i^0 described in (8) and ω_i in (2), respectively, Γ_i is a positive definite learning rate matrix, $\bar{\Lambda}_i$, Υ_i, $\bar{\Gamma}_i$, $\bar{\Upsilon}_i$, and Λ_i are positive learning rate constants, and $sgn(\cdot)$ is the sign function defined to take zero value at zero. Furthermore, P is a positive definite design matrix, which will be defined in Theorem 1.

Remark 3: In the control law (10)–(11), the term $-\rho_i B^T P\sum_{j\in N_i} b_{ij}\tilde{x}_{ij}$ guarantees the convergence of cooperative tracking error for the ideal case of the autonomous leader (i.e., $u_0 = 0$) and the absence of faults and modeling uncertainty. The term $-\hat{\kappa}_i\bar{U}_i sgn\big(\sum_{j\in N_i} b_{ij}B^T P\,\tilde{x}_{ij}\big)$ with the adaptive law (16) is designed to guarantee the robustness of leader-follower tracking with respect to a time-varying leader with an unknown input, and the term $-\bar{D}\,\hat{\omega}_i\,\bar{\eta}_i sgn\big(\sum_{j\in N_i} b_{ij}D^T P\,\tilde{x}_{ij}\big)$ with the adaptive law (17) is designed to achieve the robustness to modeling uncertainty η_i. The adaptive term $(I_m+\hat{\theta}_i)^{-1}$ in the control law (10) and the adaptive law (15) are used to compensate for the effect of actuator faults. The term $-\bar{F}\hat{\alpha}_i\,\bar{\Delta}_i sgn\big(\sum_{j\in N_i} b_{ij}F^T P\,\tilde{x}_{ij}\big)$ in (11) and the adaptive law (14) are designed to deal with the neural network residual approximation error. Lastly, the term $\hat{f}_i(x_i,\hat{\vartheta}_i)$ in (11) is the neural-network approximator with the adaptive law (13) designed to approximate unknown process fault functions.

C. Stability Analysis

The following Lemmas are needed for the design and analysis of the distributed FTC algorithms:

Lemma 1: ([18], Lemma 1) Suppose $\mathcal{H} \in \Re^{(M+1)\times(M+1)}$ is the Laplacian matrix of intercommunication graph as if the communication between the leader and followers is bidirectional. Then, the matrix

$$\Omega \triangleq \chi\mathcal{H} + \mathcal{H}^T\chi \tag{18}$$

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

Lemma 2: ([18], Lemma 2) Suppose the Laplacian matrix $\mathcal{H}$ and the diagonal matrix χ, defined in Lemma 1, have the following decomposition:

$$\mathcal{H} = \begin{bmatrix}\mathcal{H}_0 & \mathcal{H}_{12}\\ \mathcal{H}_{21} & \hat{\mathcal{H}}\end{bmatrix},\quad \chi = \begin{bmatrix}\chi_0 & 0\\ \mathbf{0}_M & \hat{\chi}\end{bmatrix}, \tag{19}$$

where $\mathcal{H}_0 \in \Re$, $\mathcal{H}_{12} \in \Re^{1\times M}$, $\mathcal{H}_{21} \in \Re^{M\times 1}$, $\hat{\mathcal{H}} \in \Re^{M\times M}$, $\chi_0 \in \Re$, $\hat{\chi} \in \Re^{M\times M}$, and $\mathbf{0}_M$ is a $M\times 1$ column vector of zeros. Then, the matrix

$$\Psi \triangleq \hat{\chi}\hat{\mathcal{H}} + \hat{\mathcal{H}}^T\hat{\chi} \tag{20}$$

is positive definite.

Let us define $\gamma \triangleq \lambda_{min}(\Psi)$ and $\varrho \triangleq \lambda_{max}(\Psi)$, where $\lambda_{min}(\cdot)$ and $\lambda_{max}(\cdot)$ denote the minimum and maximum eigenvalues, respectively. Note that based on Lemma 2, γ and ϱ are both positive.

Theorem 1: If there exist a symmetric positive definite matrix $P \in \Re^{n\times n}$, and positive constant μ, such that

$$\tilde{A}^T P + P\tilde{A} + \mu P^2 + \frac{\sigma^2}{\mu}I_n - 2PBB^T P < 0\,, \tag{21}$$

where I_n is the identity matrix, and σ is the constant defined in (3). Then, in the presence of actuator and process faults, the distributed adaptive-approximation-based fault-tolerant controller (10) with controller gains (23) and adaptive laws (12) – (17) guarantees the following properties:

1) All the signals are uniformly bounded.
2) The leader-follower consensus is achieved asymptotically, i.e., $x_i(t) \to x_0(t)$ as $t \to \infty$.

Proof: Using some algebraic manipulations, we can rewrite (10) as $u_i = \bar{u}_i - \hat{\theta}_i u_i$. Note that $(I_m + \beta_i\theta_i)u_i = u_i + \beta_i\theta_i u_i = \bar{u}_i - \hat{\theta}_i u_i + \beta_i\theta_i u_i$. By using (11) and substituting u_i into (6), and by using the leader's dynamics (i.e., $\dot{x}_0(t) = g(x_0) + Ax_0 + Bu_0$), the tracking error dynamics are given by

$$\begin{aligned}\dot{\tilde{x}}_i &= \tilde{A}\tilde{x}_i - \rho_i BB^TP\sum_{j\in N_i} b_{ij}\tilde{x}_{ij} + g(x_i) - g(x_0) + D\eta_i \\ &-B\bar{D}\hat{\omega}_i\bar{\eta}_i sgn\big(\sum_{j\in N_i} b_{ij}D^TP\,\tilde{x}_{ij}\big) + B\tilde{\theta}_i u_i - Bu_0 \\ &-BKx_0 - B\hat{\kappa}_i\bar{U}_i sgn\big(\sum_{j\in N_i} b_{ij}B^TP\,\tilde{x}_{ij}\big) + F\tilde{\vartheta}_i^T\varphi_i \\ &+F\beta_i\delta_i - B\bar{F}\hat{\alpha}_i\,\bar{\Delta}_i(x_i)sgn\big(\sum_{j\in N_i} b_{ij}F^TP\,\tilde{x}_{ij}\big) \\ &+(\beta_i-1)F\hat{f}_i(x_i,\vartheta_i) + (\beta_i-1)B\theta_i u_i\,, \end{aligned} \tag{22}$$

where $\tilde{x}_i = \tilde{x}_{i0} \triangleq x_i - x_0$, $\tilde{\vartheta}_i \triangleq \vartheta_i - \hat{\vartheta}_i$ is the network parameter estimation error associated with the ith agent, and $\tilde{\theta}_i \triangleq \theta_i - \hat{\theta}_i$ is the actuator fault parameter estimation error.

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

$$b_{ij} = \begin{cases} \chi_i k_{ij} + \chi_j k_{ji}\,, & \text{for} \quad j \neq 0 \\ \chi_i k_{i0} + \chi_0 \bar{k}_i\,, & \text{for} \quad j = 0 \end{cases} \tag{23}$$

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 b_{ij} given in (23) are the ith row and jth column entries of Ψ defined in Lemma 2. Therefore, using (22) and (23) and the definition of Ψ in Lemma 2, we can represent the collective closed-loop state dynamics as

$$\begin{aligned}\dot{\tilde{x}} &= (I_M\otimes\tilde{A})\tilde{x} - (\bar{\rho}\Psi\otimes BB^TP)\,\tilde{x} + \tilde{g} + \xi - \bar{\xi} \\ &\quad + U + \Delta + \tilde{f} + \varpi, \end{aligned} \tag{24}$$

where $\otimes$ represents the kronecker product, $\tilde{x} \in \Re^{nM}$ is the column stack vector of $\tilde{x}_i$, $\bar{\rho} = \mathrm{diag}\{\rho_1,\cdots,\rho_M\}$, and the vectors $\xi \in \Re^{nM}$, $\bar{\xi} \in \Re^{nM}$, $\tilde{g} \in \Re^{nM}$, $U \in \Re^{nM}$, $\Delta \in \Re^{nM}$, $\tilde{f} \in \Re^{nM}$, and $\varpi \in \Re^{nM}$ are defined as

$$\xi \triangleq \big[(D\eta_1)^T,\,\cdots,\,(D\eta_M)^T\big]^T \tag{25}$$

$$\tilde{g} \triangleq \big[(g(x_1)-g(x_0))^T,\,\cdots,\,(g(x_M)-g(x_0))^T\big]^T \tag{26}$$

$$U \triangleq \big[(BU_1)^T,\;\cdots,\;(BU_M)^T\big]^T \tag{27}$$

$$\Delta \triangleq \big[(F\Delta_1)^T,\;\cdots,\;(F\Delta_M)^T\big]^T \tag{28}$$

$$\tilde{f} \triangleq \big[(F\tilde{\vartheta}_1^T\varphi_1)^T,\;\cdots,\;(F\tilde{\vartheta}_M^T\varphi_M)^T\big]^T \tag{29}$$

$$\varpi \triangleq \big[(B\tilde{\theta}_1u_1)^T,\;\cdots,\;(B\tilde{\theta}_Mu_M)^T\big]^T \tag{30}$$

$$\bar{\xi} \triangleq \big[(\bar{\xi}_1)^T,\,\cdots,\,(\bar{\xi}_M)^T\big]^T \tag{31}$$

$$\begin{aligned}\bar{\xi}_i \triangleq\; & D\hat{\omega}_i\bar{\eta}_i sgn\big(\sum_{j\in N_i} b_{ij}D^TP\tilde{x}_{ij}\big) + B\hat{\kappa}_i\bar{U}_i \\ & \cdot sgn(\sum_{j\in N_i} b_{ij}B^TP\tilde{x}_{ij}) + F\hat{\alpha}_i\bar{\Delta}_i sgn(\sum_{j\in N_i} b_{ij}F^TP\tilde{x}_{ij}),\end{aligned}$$

for $i = 1,\cdots,M$, $U_i \triangleq -(u_0 + Kx_0) + (\beta_i - 1)\theta_i u_i$, and $\Delta_i \triangleq \beta_i\delta_i + (\beta_i-1)\hat{f}_i$. We consider the following Lyapunov function candidate:

$$\begin{aligned} V \;=\; & \tilde{x}^T(\Psi\otimes P)\tilde{x} + \underline{\tilde{\vartheta}}^T(I_s\otimes\Gamma)^{-1}\underline{\tilde{\vartheta}} + \tilde{\omega}^T(\Lambda)^{-1}\tilde{\omega} \\ & + \tilde{\alpha}^T(\Upsilon)^{-1}\tilde{\alpha} + \underline{\tilde{\theta}}^T(I_m\otimes\bar{\Gamma})^{-1}\underline{\tilde{\theta}} + \tilde{\kappa}^T\bar{\Upsilon}^{-1}\tilde{\kappa} \\ & + \tilde{\rho}^T(\bar{\Lambda})^{-1}\tilde{\rho}\,, \end{aligned} \tag{32}$$

where P is a positive definite matrix, $\underline{\tilde{\vartheta}} = [\,\underline{\tilde{\vartheta}}_1^T,\cdots,\underline{\tilde{\vartheta}}_M^T\,]^T$ is the collective neural network parameter estimation errors represented by $\underline{\tilde{\vartheta}}_i = [\,\tilde{\vartheta}_{i1}^T,\cdots,\tilde{\vartheta}_{is}^T\,]^T$ with $\tilde{\vartheta}_{ih} = \vartheta_{ih} - \hat{\vartheta}_{ih}$ for $h = 1,\cdots,s$, $\tilde{\alpha} = [\,\tilde{\alpha}_1,\cdots,\tilde{\alpha}_M\,]^T$, $\tilde{\kappa} = [\,\tilde{\kappa}_1,\cdots,\tilde{\kappa}_M\,]^T$, and $\tilde{\omega} = [\,\tilde{\omega}_1,\cdots,\tilde{\omega}_M\,]^T$ are the collective bounding parameter estimation errors defined as $\tilde{\alpha}_i = \alpha_i^0 - \hat{\alpha}_i$, $\tilde{\kappa}_i = \kappa_i - \hat{\kappa}_i$, and $\tilde{\omega}_i = \omega_i - \hat{\omega}_i$, respectively, $\underline{\tilde{\theta}} = [\,\underline{\tilde{\theta}}_1^T,\cdots,\underline{\tilde{\theta}}_M^T\,]^T$ is the collective actuator fault parameter estimation errors represented by $\underline{\tilde{\theta}}_i = [\,\tilde{\theta}_{i1},\cdots,\tilde{\theta}_{im}\,]^T$ with $\tilde{\theta}_{id} = \theta_{id} - \hat{\theta}_{id}$, for $d = 1,\cdots,m$, $\tilde{\rho} = [\,\rho_1 - \rho,\cdots,\rho_M - \rho\,]^T$ is the collective coupling gain estimation errors, ρ is a positive constant, and $\Gamma = \mathrm{diag}\{\Gamma_1,\cdots,\Gamma_M\}$, $\bar{\Gamma} = \mathrm{diag}\{\bar{\Gamma}_1,\cdots,\bar{\Gamma}_M\}$, $\Lambda = \mathrm{diag}\{\Lambda_1,\cdots,\Lambda_M\}$, $\Upsilon = \mathrm{diag}\{\Upsilon_1,\cdots,\Upsilon_M\}$, $\bar{\Upsilon} = \mathrm{diag}\{\bar{\Upsilon}_1,\cdots,\bar{\Upsilon}_M\}$, and $\bar{\Lambda} = \mathrm{diag}\{\bar{\Lambda}_1,\cdots,\bar{\Lambda}_M\}$, are constant learning rate matrices. Then, the time derivative of the Lyapunov function (32) along the solution of (24) is

$$\begin{aligned}\dot{V} \;=\; & \tilde{x}^T\Big[(\Psi\otimes P)(I_M\otimes\tilde{A}) + (I_M\otimes\tilde{A})^T(\Psi\otimes P)\Big]\tilde{x} \\ & -\tilde{x}^T\big[(\Psi\otimes P)(\bar{\rho}\Psi\otimes\bar{P}) + (\bar{\rho}\Psi\otimes\bar{P})^T(\Psi\otimes P)\big]\tilde{x} \\ & +2\Big\{\tilde{x}^T(\Psi\otimes P)\big[\xi - \bar{\xi} + U + \Delta + \tilde{g} + \tilde{f} + \varpi) \\ & + \underline{\tilde{\vartheta}}^T(I_s\otimes\Gamma)^{-1}\dot{\underline{\tilde{\vartheta}}} + \tilde{\omega}^T(\Lambda)^{-1}\dot{\tilde{\omega}} + \tilde{\alpha}^T(\Upsilon)^{-1}\dot{\tilde{\alpha}} \\ & + \underline{\tilde{\theta}}^T(I_m\otimes\bar{\Gamma})^{-1}\dot{\underline{\tilde{\theta}}} + \tilde{\kappa}^T(\bar{\Upsilon})^{-1}\dot{\tilde{\kappa}} + \tilde{\rho}^T(\bar{\Lambda})^{-1}\dot{\tilde{\rho}}\Big\}, \end{aligned} \tag{33}$$

where $\bar{P} \triangleq BB^TP$. By using the properties that $(\check{A}\otimes\check{B})^T = \check{A}^T\otimes\check{B}^T$, $(\check{A}\otimes\check{C})(\check{B}\otimes\check{D}) = \check{A}\check{B}\otimes\check{C}\check{D}$, and $\check{A}\otimes\check{B} + \check{A}\otimes\check{C} = \check{A}\otimes(\check{B}+\check{C})$ for matrices $\check{A}$, $\check{B}$, $\check{C}$, and $\check{D}$ of appropriate dimensions, we have

$$\begin{aligned} & \tilde{x}^T\big[(\Psi\otimes P)(I_M\otimes\tilde{A}) + (I_M\otimes\tilde{A}^T)(\Psi\otimes P)\big]\tilde{x} \\ & \quad = \;\tilde{x}^T\big[\Psi\otimes\big(P\tilde{A} + \tilde{A}^TP\big)\big]\tilde{x}\,. \end{aligned} \tag{34}$$

Additionally, as shown in the proof of Theorem 2 in [21], we can easily show that

$$\begin{aligned} & -\tilde{x}^T\big[(\Psi\otimes P)(\bar{\rho}\Psi\otimes\bar{P}) + (\bar{\rho}\Psi\otimes\bar{P})^T(\Psi\otimes P)\big]\tilde{x} \\ & \quad = -2\,\tilde{x}^T(\Psi\bar{\rho}\Psi\otimes PBB^TP)\,\tilde{x} \\ & \quad = -2\sum_{i=1}^M\rho_i\Big(\sum_{j\in N_i} b_{ij}B^TP\,\tilde{x}_{ij}\Big)^T\Big(\sum_{j\in N_i} b_{ij}B^TP\,\tilde{x}_{ij}\Big). \end{aligned} \tag{35}$$

Note that based on Lemma 2, $\Psi^T = \Psi$. Furthermore, by using the property that $2\hat{a}^T\hat{b} \le \bar{c}\,\hat{a}^T\hat{a} + \frac{1}{\bar{c}}\hat{b}^T\hat{b}$ for any positive constant $\bar{c}$ and vectors $\hat{a}$ and $\hat{b}$, we have

$$\begin{aligned} & 2\,\tilde{x}^T(\Psi\otimes P)\,\tilde{g} = 2\,\tilde{x}^T(\Psi^{\frac{1}{2}}\otimes P)(\Psi^{\frac{1}{2}}\otimes I_n)\,\tilde{g} \\ & \quad \le \mu\,\tilde{x}^T(\Psi\otimes P^2)\tilde{x} + \frac{1}{\mu}\tilde{g}^T\big(\Psi\otimes I_n\big)\,\tilde{g}\,, \end{aligned} \tag{36}$$

where $\Psi^{\frac{1}{2}}$ is a positive definite matrix such that $(\Psi^{\frac{1}{2}})^2 = \Psi$, and μ is a positive constant. Based on (3) given in Assumption 1.3, we have

$$\begin{aligned}\tilde{g}^T\big(\Psi \otimes I_n\big)\,\tilde{g} &= \sum_{i=1}^{M} \tilde{g}_i^T \sum_{j\in N_i} b_{ij}(\tilde{g}_i - \tilde{g}_j) \\ &= \frac{1}{2}\sum_{i=1}^{M}\sum_{j\in N_i} b_{ij}(\tilde{g}_i - \tilde{g}_j)^T(\tilde{g}_i - \tilde{g}_j) = \frac{1}{2}\sum_{i=1}^{M}\sum_{j\in N_i} b_{ij}|\tilde{g}_i - \tilde{g}_j|^2 \\ &\le \frac{1}{2}\sum_{i=1}^{M}\sigma^2 \sum_{j\in N_i} b_{ij}|\tilde{x}_i - \tilde{x}_j|^2 = \sigma^2\tilde{x}^T\big(\Psi\otimes I_n\big)\,\tilde{x}\,, \end{aligned} \tag{37}$$

where $\tilde{g}_i \triangleq g(x_i) - g(x_0)$ as defined in (26). Therefore, by substituting (37) into (36), we have

$$2\,\tilde{x}^T(\Psi\otimes P)\,\tilde{g} \le \mu\,\tilde{x}^T(\Psi\otimes P^2)\tilde{x} + \frac{\sigma^2}{\mu}\,\tilde{x}^T\big(\Psi\otimes I_n\big)\,\tilde{x}\,. \tag{38}$$

Based on (29) and (30), we have

$$\begin{aligned}\tilde{x}^T(\Psi\otimes P)(\tilde{f}+\varpi) \;=\; \sum_{i=1}^{M}\Big[&\sum_{j\in N_i} b_{ij}\,\tilde{x}_{ij}^T PF\tilde{\vartheta}_i^T\varphi_i \\ &+ \sum_{j\in N_i} b_{ij}\,\tilde{x}_{ij}^T PB\tilde{\theta}_i u_i\Big]. \end{aligned}\tag{39}$$

By using (25), (27), (28), and (31), we have

$$\begin{aligned}&\tilde{x}^T(\Psi\otimes P)(\xi - \bar{\xi} + U + \Delta) \\ &= \sum_{i=1}^{M}\sum_{j\in N_i} b_{ij}\,\tilde{x}_{ij}^T P(D\eta_i + BU_i + F\Delta_i - \bar{\xi}_i) \\ &= \sum_{i=1}^{M}\Big[(\sum_{j\in N_i} b_{ij}\,D^T\,P\,\tilde{x}_{ij})^T\eta_i - (\sum_{j\in N_i} b_{ij}\,D^T\,P\,\tilde{x}_{ij})^T\hat{\omega}_i\bar{\eta}_i \\ &\quad \cdot sgn(\sum_{j\in N_i} b_{ij}D^TP\tilde{x}_{ij}) + (\sum_{j\in N_i} b_{ij}B^TP\tilde{x}_{ij})^T U_i \\ &\quad -(\sum_{j\in N_i} b_{ij}\,B^T\,P\,\tilde{x}_{ij})^T\hat{\kappa}_i\bar{U}_i sgn(\sum_{j\in N_i} b_{ij}\,B^TP\,\tilde{x}_{ij}) \\ &\quad +(\sum_{j\in N_i} b_{ij}\,F^T\,P\,\tilde{x}_{ij})^T\Delta_i - (\sum_{j\in N_i} b_{ij}\,F^T\,P\,\tilde{x}_{ij})^T \\ &\quad \cdot\hat{\alpha}_i\bar{\Delta}_i sgn(\sum_{j\in N_i} b_{ij}\,F^TP\,\tilde{x}_{ij})\Big]. \end{aligned}\tag{40}$$

Based on Assumption 1.1, by using (2) and (7) given in assumptions 1.2 and 1.5, respectively, and the property that $(\hat{a})^T sgn\big(\hat{a}\big) \ge |\hat{a}|$, it follows from (40) that

$$\begin{aligned}&\tilde{x}^T(\Psi\otimes P)(\xi - \bar{\xi} + U + \Delta) \\ &\le \sum_{i=1}^{M}\Big[\tilde{\omega}_i\Big|\sum_{j\in N_i} b_{ij}\,D^TP\,\tilde{x}_{ij}\Big|\bar{\eta}_i + \tilde{\kappa}_i\Big|\sum_{j\in N_i} b_{ij}\,B^TP\,\tilde{x}_{ij}\Big|\bar{U}_i \\ &\quad + \tilde{\alpha}_i\Big|\sum_{j\in N_i} b_{ij}\,F^TP\,\tilde{x}_{ij}\Big|\bar{\Delta}_i\Big]. \end{aligned}\tag{41}$$

By using the adaptive law (12), we have

$$\begin{aligned}&2\tilde{\rho}^T(\bar{\Lambda})^{-1}\dot{\tilde{\rho}} \\ &\quad= 2\sum_{i=1}^{M}(\rho_i - \rho)\Big(\sum_{j\in N_i} b_{ij}B^TP\,\tilde{x}_{ij}\Big)^T\Big(\sum_{j\in N_i} b_{ij}B^TP\,\tilde{x}_{ij}\Big) \\ &\quad= 2\sum_{i=1}^{M}\rho_i\Big(\sum_{j\in N_i} b_{ij}B^TP\,\tilde{x}_{ij}\Big)^T\Big(\sum_{j\in N_i} b_{ij}B^TP\,\tilde{x}_{ij}\Big) \\ &\qquad -2\rho\,\tilde{x}^T(\Psi^2\otimes P\bar{P})\,\tilde{x}. \end{aligned}\tag{42}$$

Additionally, using the eigenvalue properties of functions of a square matrix [25], it can be shown that for the positive definite matrix Ψ, $\Psi^2 - \gamma\Psi$ is positive semidefinite (i.e., $\Psi^2 - \gamma\Psi \ge 0$), where γ is the smallest eigenvalue of the matrix Ψ defined in theorem 1. Also, for matrices $\check{E}$ and $\check{F}$, eigenvalues of the matrix $\check{E}\otimes\check{F}$ are products of eigenvalues of $\check{E}$ and $\check{F}$ (Theorem 6 in [26]). Thus, for the positive semidefinite matrix $P\bar{P}$, it can be obtained that $\tilde{x}^T\big[(\Psi^2 - \gamma\Psi)\otimes P\bar{P}\big]\tilde{x} \ge 0$ or equivalently $\tilde{x}^T\big(\Psi^2\otimes P\bar{P}\big)\tilde{x} \ge \gamma\tilde{x}^T\big(\Psi\otimes P\bar{P}\big)\tilde{x}$. Therefore, we have

$$-2\rho\,\tilde{x}^T(\Psi^2\otimes P\bar{P})\,\tilde{x} \;\le\; -2\rho\,\gamma\,\tilde{x}^T(\Psi\otimes PBB^TP)\,\tilde{x}. \tag{43}$$

Let us define

$$Q^c \triangleq \tilde{A}^TP + P\tilde{A} + \mu P^2 + \frac{\sigma^2}{\mu}I_n - 2\rho\gamma PBB^TP\,. \tag{44}$$

Therefore, by substituting (34) – (43) into (33), and by using the adaptive laws given by (13) – (17), we obtain

$$\dot{V} \le \tilde{x}(\Psi\otimes Q^c)\tilde{x}\,. \tag{45}$$

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 [24]. By selecting ρ sufficiently large such that $\rho\gamma \ge 1$, using (21), positive definiteness of Ψ due to Lemma 2, and the property that eigenvalues of the matrix $\Psi\otimes Q^c$ are products of eigenvalues of Ψ and Q^c (Theorem 6 in [26]), we know $\dot{V}$ is negative semidefinite. Thus, we conclude that $\tilde{x}_i$, ρ_i, $\hat{\vartheta}_i$, $\hat{\theta}_i$, $\hat{\kappa}_i$, $\hat{\omega}_i$, and $\hat{\alpha}_i$ are uniformly bounded. By integrating both sides of (45), it can be 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 (10), (6), and the smoothness of the function g_i, we have $u_i \in L_\infty$ and $\dot{x}_i \in L_\infty$. Since $\tilde{x}_i \in L_\infty \cap L_2$, $\dot{\tilde{x}}_i \in L_\infty$, using Barbalat's Lemma [27], $\tilde{x} \to 0$ as $t \to \infty$, hence concluding the proof. □

Remark 4: The condition (21) can be transformed into standard linear matrix inequalities. Then, a feasible solution to (21) can be possibly found by using LMI tools. Furthermore, the sign function used in this paper may possibly create chattering problems, which could be remedied by using a smooth approximation of the sign function, for instance, the hyperbolic tangent function (see, e.g., page 397 in [24]). In [19] and [28], a continuous adaptive FTC scheme utilizing the hyperbolic function is presented, guaranteeing the cooperative tracking error converges to a small neighborhood around zero, which can be made as small as possible by using suitable design parameters. Interested readers can refer to Section 5.2 in [28] for more details.

III. Distributed FTC: Input-Output Agent Systems

In this section, the results in Section II are extended to a class of input-output multi-agent systems where only partial state measurements are available.

A. Distributed Multi-Agent System Model

Consider a set of M agents where the dynamics of the ith agent, $i = 1, \cdots, M$, is described by the following differential equation

$$\begin{aligned} \dot{x}_i &= Ax_i + g(x_i) + Bu_i + \beta_i(t-T_{iu})B\theta_i u_i \\ &\quad + D\eta_i(x_i,t) + \beta_i(t-T_{if})Ff_i(y_i) \qquad (46) \\ y_i &= Cx_i\,, \end{aligned}$$

where $x_i \in \Re^n$, $u_i \in \Re^m$, and $y_i \in \Re^l$ are the state, input, and output vector of the ith agent, respectively. Additionally, $g : \Re^n \mapsto \Re^n$, $\eta_i : \Re^n \times \Re^+ \mapsto \Re^v$, and $f_i : \Re^l \mapsto \Re^s$ are smooth vector fields representing the known nonlinearity, unknown modeling uncertainty, and process fault in the state dynamics of the ith agent, respectively, A, B, C, D and F are matrices with appropriate dimensions, and the pairs (A, B) and (A, C) are stabilizable and detectable, respectively. The changes in the dynamics of ith agent due to the occurrence of process and actuator faults in (46) are represented by $\beta_i(t-T_{if})Ff_i(y_i)$ and $\beta_i(t-T_{iu})B\theta_i u_i$, which occur at some unknown time T_{if} and T_{iu}, respectively.

Without loss of generality, let the leader be identified as agent number 0 with unknown state x_0 and a reference output y_0, where $\dot{x}_0(t) = g(x_0) + Ax_0 + Bu_0$, and $y_0 = Cx_0$. The objective of this section is to design a distributed FTC scheme which guarantees that the state of ith agent (i.e., $x_i(t)$) should track the state of the time-varying leader (i.e., $x_0(t)$) by utilizing only local output measurements and state estimation information exchanged between neighboring agents, even in the presence of process and actuator faults.

Analogously, the linearly parametrized neural network model used to estimate the unknown process fault function $f_i(y_i)$ is described as follows:

$$\hat{f}_i(y_i, \hat{\vartheta}_i) = \hat{\vartheta}_i^T \varphi_i(y_i)\,. \qquad (47)$$

Let us denote the residual approximation error as $\delta_i(y_i)$. Assumptions 1.2 and 1.5 will be modified as follows:

Assumption 2.1: The modeling uncertainty in each local agent, represented by $\eta_i(x_i, t)$ in (46), satisfies

$$|\eta_i(x_i,t)| \;\leq\; \omega_i\,\bar{\eta}_i(y_i,t)\,, \quad \forall x_i \in \Re^n,\ \forall y_i \in \Re^l \qquad (48)$$

where $\bar{\eta}_i$ is a known positive bounding function and ω_i is an unknown constant.

Assumption 2.2: For each $i = 1, \cdots, M$, the network residual approximation error satisfies

$$|\delta_i(y_i)| \leq \alpha_i \bar{\delta}_i(y_i)\,,\ \ \forall y_i \in \Re^l \qquad (49)$$

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

Note that in Assumptions 2.1 and 2.2, the bounds on modeling uncertainty and network approximation error are allowed to be a function of agent outputs, which is less restrictive than the constant bound considered in [9] and [10].

B. Distributed FTC Design

Since the agent state is not available for control design, a state estimator is needed for estimating the state. Let θ_i^e represent an unknown constant defined as

$$\theta_i^e \triangleq \sup_{t\geq T_{iu}} \max\left\{ \big|\big|\beta_i(t-T_{iu})\,\theta_i\big|\big| \right\}. \qquad (50)$$

Note that the fault time profile $\beta_i(t - T_{iu})$ satisfies $0 < \beta_i \leq 1$ for $t \geq T_{iu}$. Therefore, the finite constant θ_i^e defined by (50) always exists. Then, by using (46), the following state estimator is chosen:

$$\begin{aligned} \dot{\hat{x}}_i &= A\hat{x}_i + g(\hat{x}_i) + Bu_i + L\tilde{y}_i^e - B\hat{\theta}_i^e\,|u_i|\,sgn\big(G_B\,\tilde{y}_i^e\big) \\ &\quad -F\Big[\hat{\alpha}_i^e\bar{\Delta}_i(y_i)sgn\big(\bar{F}^TG_B\,\tilde{y}_i^e\big) + \hat{f}_i^e(y_i,\hat{\vartheta}_i^e)\Big] \\ &\quad -D\,\hat{\omega}_i^e\,\bar{\eta}_i sgn(\bar{D}^TG_B\,\tilde{y}_i^e) \qquad (51) \\ \hat{y}_i &= C\hat{x}_i\,, \\ \dot{\hat{\vartheta}}_{ih}^e &= \Gamma_i\big(\bar{F}_h^TG_B\,\tilde{y}_i^e\big)\varphi_i^e(y_i)\,, \qquad (52) \\ \dot{\hat{\alpha}}_i^e &= \Upsilon_i\,\big|\bar{F}^TG_B\,\tilde{y}_i^e\big|\,\bar{\Delta}_i(y_i)\,, \qquad (53) \\ \dot{\hat{\omega}}_i^e &= \Lambda_i\,\big|\bar{D}^TG_B\,\tilde{y}_i^e\big|\,\bar{\eta}_i(y_i,t)\,, \qquad (54) \\ \dot{\hat{\theta}}_i^e &= \bar{\Gamma}_i\,\big|G_B\,\tilde{y}_i^e\big|\cdot\big|u_i\big|\,, \qquad (55) \end{aligned}$$

where $\hat{x}_i$ and $\hat{y}_i$ represent the estimated local state and output variables of the ith agent, respectively, $\tilde{y}_i^e \triangleq y_i - \hat{y}_i$ denotes the output estimation error of the ith agent, $\bar{D}$ and $\bar{F}$ are given in (4), $L \in \Re^{n\times l}$ is a design gain matrix chosen such that the matrix $\bar{A} \triangleq A - LC$ is Hurwitz, $\hat{\theta}_i^e$ is an estimation of the actuator fault magnitude parameter θ_i^e defined in (50), $\hat{f}_i^e = (\hat{\vartheta}_i^e)^T\varphi_i^e(y_i)$ provides the adaptive online approximation of the unknown process fault for the state estimator, $\hat{\vartheta}_i^e$ is an estimation of the neural network parameter matrix ϑ_i, $\hat{\vartheta}_{ih}^e$ is the hth row of $\hat{\vartheta}_i^e$, for $h = 1, \cdots, s$, φ_i^e is the collective vector of fixed basis functions, $\bar{F}_h$ is the hth column of matrix $\bar{F}$, $\hat{\alpha}_i^e$ is an estimation of the unknown bounding constant α_i^0 described in (8), $\hat{\omega}_i^e$ is an estimation of the unknown bounding constant ω_i, $\bar{\Delta}_i(y_i) \triangleq \bar{\delta}_i(y_i) + |\varphi_i(y_i)|$, $\bar{U}_i \triangleq 1 + |u_i|$, Γ_i is a positive definite learning rate matrix, $\bar{\Gamma}_i$, Υ_i, and Λ_i are positive learning rate constants, G_B is a design matrix satisfying

$$PB \;=\; C^TG_B^T\,, \qquad (56)$$

and P is a positive definite design matrix to be defined in Theorem 2.

In the presence of process and actuator faults, by adding and subtracting the term $\hat{f}_i(y_i, \vartheta_i)$, and defining the residual approximation error for the ith agent as $\delta_i \triangleq f_i(y_i) - \hat{f}_i(y_i, \vartheta_i)$, the system dynamics described by (46) can be rewritten as

$$\begin{aligned} \dot{x}_i &= Ax_i + g(x_i) + B(I_m + \beta_i\theta_i)u_i + D\eta_i(x_i,t) \\ &\quad +F\big[\hat{f}_i(y_i,\vartheta_i) + (\beta_i - 1)\hat{f}_i(y_i,\vartheta_i) + \beta_i\delta_i(y_i)\big] \qquad (57) \\ y_i &= Cx_i\,. \end{aligned}$$

Then, based on the system model (57), the neural network model (47), and Assumption 2.2, the adaptive neural controller

(10)–(17) for input-output systems are adjusted as follows:

$$u_i = (I_m + \hat{\theta}_i^c)^{-1}\tau_i \,, \tag{58}$$

$$\begin{aligned}\tau_i \triangleq & -\rho B^T P \sum_{j\in N_i} b_{ij}\big(\hat{x}_i - \hat{x}_j\big) - \hat{\kappa}_i \bar{U}_i\, sgn\big(\sum_{j\in N_i} b_{ij} G_B\, \tilde{y}_{ij}^c\big) \\ & -K\hat{x}_i - \bar{\eta}_i\, \bar{D}\, \hat{\omega}_i^c sgn\big(\sum_{j\in N_i} b_{ij}\bar{D}^T G_B \tilde{y}_{ij}^c\big) \\ & -\bar{F}\big[\hat{\alpha}_i^c\, \bar{\Delta}_i sgn\big(\sum_{j\in N_i} b_{ij}\, \bar{F}^T G_B\, \tilde{y}_{ij}^c\big) + \hat{f}_i^c(y_i, \hat{\vartheta}_i^c)\big], \end{aligned} \tag{59}$$

$$\dot{\hat{\vartheta}}_{ih}^c = \Gamma_i \left(\sum_{j\in N_i} b_{ij} \bar{F}_h{}^T G_B\, \tilde{y}_{ij}^c \right)\varphi_i^c(y_i) \,, \tag{60}$$

$$\dot{\hat{\alpha}}_i^c = \Upsilon_i \left| \sum_{j\in N_i} b_{ij}\, G_F\, \tilde{y}_{ij}^c \right| \bar{\Delta}_i(y_i) \,, \tag{61}$$

$$\dot{\hat{\theta}}_{id}^c = \mathcal{P}_{\bar{\theta}_{id}} \left\{ \bar{\Gamma}_i \sum_{j\in N_i} b_{ij} (G_{Bd}\, \tilde{y}_{ij}^c)^T u_{id} \right\}, \tag{62}$$

$$\dot{\hat{\kappa}}_i = \bar{\Upsilon}_i \left| \sum_{j\in N_i} b_{ij} G_B\, \tilde{y}_{ij}^c \right| \bar{U}_i \,, \tag{63}$$

$$\dot{\hat{\omega}}_i^c = \Lambda_i \left| \sum_{j\in N_i} b_{ij} \bar{D}^T G_B\, \tilde{y}_{ij}^c \right| \bar{\eta}_i(y_i, t) \,, \tag{64}$$

where $\tilde{y}_{ij}^c \triangleq y_i - y_j$, $\hat{\theta}_i^c = \text{diag}\{\hat{\theta}_{i1}^c, \cdots, \hat{\theta}_{im}^c\}$ with each component $\hat{\theta}_{id}^c$ in (62) being an estimation of the actuator fault magnitude parameter θ_{id}, ρ is a positive design constant, $\hat{f}_i^c = (\hat{\vartheta}_i^c)^T \varphi_i^c(y_i)$ provides the adaptive online approximation of the unknown process fault for the designed fault-tolerant control algorithm, $\hat{\vartheta}_i^c$ is an estimation of the neural network parameter matrix ϑ_i, $\hat{\vartheta}_{ih}^c$ is the hth row of $\hat{\vartheta}_i^c$, for $h = 1, \cdots, s$, φ_i^c is the collective vector of fixed basis functions, $\hat{\kappa}_i$ is an estimation of the unknown positive constant bound κ_i described in (9), $\hat{\omega}_i^c$ is an estimation of the unknown positive bounding constant ω_i, G_{Bd} is the dth row of matrix G_B, and $\hat{\alpha}_i^c$ is an estimation of the unknown bounding constant α_i^0 described in (8).

C. Stability Analysis

For each agent, by using (57) and (51), the collective state estimation error dynamics are

$$\dot{\tilde{x}}^e = (I_M \otimes \bar{A})\, \tilde{x}^e + \tilde{g}^e + \xi - \bar{\xi}^e + \tilde{f}^e + \Delta + U^e, \tag{65}$$

where $\tilde{x}^e$ is the column stack vector of the state estimation errors $\tilde{x}_i^e \triangleq x_i - \hat{x}_i$, the vectors ξ and Δ are defined in (25) and (28), and the vectors $\tilde{g}^e \in \Re^{Mn}$, $\tilde{f}^e \in \Re^{Mn}$, $U^e \in \Re^{Mn}$, and $\bar{\xi}^e \in \Re^{Mn}$ are defined as

$$\tilde{g}^e \triangleq \Big[\big(g(x_1) - g(\hat{x}_1)\big)^T, \,\cdots,\, \big(g(x_M) - g(\hat{x}_M)\big)^T\Big]^T, \tag{66}$$

$$\tilde{f}^e \triangleq \left[\Big(F(\tilde{\vartheta}_1^e)^T \varphi_1^e\Big)^T, \;\cdots,\; \Big(F(\tilde{\vartheta}_M^e)^T \varphi_M^e\Big)^T\right]^T, \tag{67}$$

$$U^e \triangleq \big[(B\,\beta_1\,\theta_1\,u_1)^T, \;\cdots,\; (B\,\beta_M\,\theta_M\,u_M)^T\big]^T, \tag{68}$$

$$\bar{\xi}^e \triangleq \big[(\bar{\xi}_1^e)^T, \,\cdots,\, (\bar{\xi}_M^e)^T\big]^T, \tag{69}$$

$$\begin{aligned}\bar{\xi}_i^e \triangleq & \; D\hat{\omega}_i^e \bar{\eta}_i sgn\big(\bar{D}^T G_B\, \tilde{y}_i^e\big) + F\hat{\alpha}_i^e \bar{\Delta}_i(y_i) sgn\big(\bar{F}^T G_B \tilde{y}_i^e\big) \\ & + B\hat{\theta}_i^e\, |u_i|\, sgn\big(G_B\, \tilde{y}_i^e\big)\,, \end{aligned}$$

for $i = 1, \cdots, M$, where $\tilde{\vartheta}_i^e = \vartheta_i - \hat{\vartheta}_i^e$ is the network parameter estimation error associated with the ith agent.

Additionally, using some algebraic manipulations, we can rewrite (58) as $u_i = \tau_i - \hat{\theta}_i u_i$. Therefore, by using (59) and (4), and substituting u_i into (57), the closed-loop dynamics are given by

$$\begin{aligned}\dot{x}_i = & \; g(x_i) + \tilde{A}x_i + BK\tilde{x}_i^e + D\eta_i - D\hat{\omega}_i^c \bar{\eta}_i sgn\big(\sum_{j\in N_i} b_{ij}\bar{D}^T G_B \tilde{y}_{ij}^c\big) \\ & -\rho BB^T P \sum_{j\in N_i} b_{ij}(\hat{x}_i - \hat{x}_j) - B\hat{\kappa}_i \bar{U}_i sgn\big(\sum_{j\in N_i} b_{ij} G_B\, \tilde{y}_{ij}^c\big) \\ & + F\left[(\tilde{\vartheta}_i^c)^T \varphi_i^c + \beta_i\delta_i - \hat{\alpha}_i^c\, \bar{\Delta}_i sgn\big(\sum_{j\in N_i} b_{ij} \bar{F}^T G_B \tilde{y}_{ij}^c\big)\right] \\ & + (\beta_i - 1)F\hat{f}_i^c(y_i, \vartheta_i) + B\tilde{\theta}_i^c u_i + (\beta_i - 1)B\theta_i u_i \,, \\ y_i = & \; Cx_i \,, \end{aligned} \tag{70}$$

where $\tilde{\vartheta}_i^c = \vartheta_i - \hat{\vartheta}_i^c$ is the network parameter estimation error associated with the ith agent, and $\tilde{\theta}_i^c = \theta_i - \hat{\theta}_i^c$ is the actuator fault parameter estimation error corresponding to the ith agent.

Note that the term $\hat{x}_i - \hat{x}_j$ in (70) can be rewritten as $(\hat{x}_i - x_i) + (x_i - x_0) - (\hat{x}_j - x_j) - (x_j - x_0)$. Therefore, by using (23) and the definition of Ψ in Lemma 2, the collective tracking error dynamics are given by

$$\begin{aligned}\dot{\tilde{x}}^c = & \; (I_M \otimes \tilde{A})\tilde{x}^c + \rho(\Psi \otimes BB^T P)\tilde{x}^e - \rho(\Psi \otimes BB^T P)\tilde{x}^c + \tilde{g} \\ & + (I_M \otimes BK)\tilde{x}^e + \xi - \bar{\xi}^c + U + \tilde{f}^c + \Delta + \varpi^c \,, \end{aligned} \tag{71}$$

where $\tilde{x}^c$ is the column stack vector of tracking errors $\tilde{x}_i^c \triangleq x_i - x_0$, the vectors ξ, $\tilde{g}$, U, and Δ are defined in (25), (26), (27), and (28), and the vectors $\tilde{f}^c \in \Re^{Mn}$, $\varpi^c \in \Re^{Mn}$, and $\bar{\xi}^c \in \Re^{Mn}$ are defined as

$$\tilde{f}^c \triangleq \left[\Big(F(\tilde{\vartheta}_1^c)^T \varphi_1^c\Big)^T, \;\cdots,\; \Big(F(\tilde{\vartheta}_M^c)^T \varphi_M^c\Big)^T\right]^T, \tag{72}$$

$$\varpi^c \triangleq \big[(B\tilde{\theta}_1^c u_1)^T, \;\cdots,\; (B\tilde{\theta}_M^c u_M)^T\big]^T, \tag{73}$$

$$\bar{\xi}^c \triangleq \big[(\bar{\xi}_1^c)^T, \,\cdots,\, (\bar{\xi}_M^c)^T\big]^T, \tag{74}$$

$$\begin{aligned}\bar{\xi}_i^c \triangleq & \; D\hat{\omega}_i^c \bar{\eta}_i sgn\big(\sum_{j\in N_i} b_{ij}\bar{D}^T G_B \tilde{y}_{ij}^c\big) + B\hat{\kappa}_i \bar{U}_i \\ & \cdot sgn\big(\sum_{j\in N_i} b_{ij} G_B \tilde{y}_{ij}^c\big) + F\hat{\alpha}_i^c \bar{\Delta}_i sgn\big(\sum_{j\in N_i} b_{ij} \bar{F}^T G_B \tilde{y}_{ij}^c\big), \end{aligned}$$

for $i = 1, \cdots, M$.

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

$$\begin{aligned} V = & \; (\tilde{x}^c)^T(\Psi \otimes P)\tilde{x}^c + (\tilde{x}^e)^T(I_M \otimes P)\tilde{x}^e + (\tilde{\alpha}^c)^T(\Upsilon)^{-1}\tilde{\alpha}^c \\ & + (\underline{\tilde{\vartheta}}^c)^T(I_s \otimes \Gamma)^{-1}\underline{\tilde{\vartheta}}^c + (\underline{\tilde{\theta}}^c)^T(I_m \otimes \bar{\Gamma})^{-1}\underline{\tilde{\theta}}^c + (\tilde{\omega}^c)^T \\ & \cdot(\Lambda)^{-1}\tilde{\omega}^c + (\underline{\tilde{\vartheta}}^e)^T(I_s \otimes \Gamma)^{-1}\underline{\tilde{\vartheta}}^e + (\tilde{\alpha}^e)^T(\Upsilon)^{-1}\tilde{\alpha}^e \\ & + (\tilde{\theta}^e)^T(\bar{\Gamma})^{-1}\tilde{\theta}^e + (\tilde{\omega}^e)^T(\Lambda)^{-1}\tilde{\omega}^e + \tilde{\kappa}^T(\bar{\Upsilon})^{-1}\tilde{\kappa}\,, \end{aligned} \tag{75}$$

where $\underline{\tilde{\vartheta}}^c = [(\underline{\tilde{\vartheta}}_1^c)^T, \cdots, (\underline{\tilde{\vartheta}}_M^c)^T]^T$ and $\underline{\tilde{\vartheta}}^e = [(\underline{\tilde{\vartheta}}_1^e)^T, \cdots, (\underline{\tilde{\vartheta}}_M^e)^T]^T$ are the collective neural network parameter estimation errors represented by $\underline{\tilde{\vartheta}}_i^c = [(\vartheta_{i1} - \hat{\vartheta}_{i1}^c)^T, \cdots, (\vartheta_{is} - \hat{\vartheta}_{is}^c)^T]^T$ and $\underline{\tilde{\vartheta}}_i^e = [(\vartheta_{i1} - \hat{\vartheta}_{i1}^e)^T, \cdots, (\vartheta_{is} - \hat{\vartheta}_{is}^e)^T]^T$, respectively, $\tilde{\alpha}^c = [\tilde{\alpha}_1^c, \cdots, \tilde{\alpha}_M^c]^T$, $\tilde{\alpha}^e = [\tilde{\alpha}_1^e, \cdots, \tilde{\alpha}_M^e]^T$,

$\tilde{\omega}^c = [\tilde{\omega}_1^c, \cdots, \tilde{\omega}_M^c]^T$, $\tilde{\omega}^e = [\tilde{\omega}_1^e, \cdots, \tilde{\omega}_M^e]^T$, and $\tilde{\kappa} = [\tilde{\kappa}_1, \cdots, \tilde{\kappa}_M]^T$ are the collective bounding parameter estimation errors defined as $\tilde{\alpha}_i^c = \alpha_i^0 - \hat{\alpha}_i^c$, $\tilde{\alpha}_i^e = \alpha_i^0 - \hat{\alpha}_i^e$, $\tilde{\omega}_i^c = \omega_i - \hat{\omega}_i^c$, $\tilde{\omega}_i^e = \omega_i - \hat{\omega}_i^e$, and $\tilde{\kappa}_i = \kappa_i - \hat{\kappa}_i$, respectively, $\underline{\tilde{\theta}}^c = [\underline{\tilde{\theta}}_1^c, \cdots, \underline{\tilde{\theta}}_M^c]^T$ and $\tilde{\theta}^e$ are the collective actuator fault parameter estimation errors represented by $\underline{\tilde{\theta}}_i^c = [\theta_{i1} - \hat{\theta}_{i1}^c, \cdots, \theta_{im} - \hat{\theta}_{im}^c]^T$ and $\tilde{\theta}_i^e = \theta_i^e - \hat{\theta}_i^e$, respectively, and Γ, $\bar{\Gamma}$, Υ, $\bar{\Upsilon}$, and Λ are constant learning rate matrices defined in (32).

Then, the time derivative of the Lyapunov function (75) along the solution of (71) and (65) is given by

$$
\begin{aligned}
\dot{V} = {} & (\tilde{x}^c)^T \Big[(\Psi \otimes P)(I_M \otimes \tilde{A}) + (I_M \otimes \tilde{A})^T(\Psi \otimes P)\Big] \tilde{x}^c \\
& -\rho\, (\tilde{x}^c)^T \left[(\Psi \otimes P)(\Psi \otimes \bar{P}) + (\Psi \otimes \bar{P})^T(\Psi \otimes P)\right] \tilde{x}^c \\
& +(\tilde{x}^e)^T \left[(I_M \otimes P)(I_M \otimes \bar{A}) + (I_M \otimes \bar{A})^T(I_M \otimes P)\right] \tilde{x}^e \\
& +2 \Big\{ (\tilde{x}^c)^T(\Psi \otimes P) \big(\rho\, (\Psi \otimes \bar{P})\tilde{x}^e + (I_M \otimes BK)\tilde{x}^e\big) \\
& +(\tilde{x}^c)^T(\Psi \otimes P) \Big[\tilde{g} + \tilde{f}^c + \varpi^c + \xi - \bar{\xi}^c + U + \Delta\Big] \\
& +(\tilde{x}^e)^T(I_M \otimes P) \Big[\tilde{g}^e + \varpi^e + \tilde{f}^e + \xi - \bar{\xi}^e + \Delta + U^e\Big] \\
& +(\underline{\tilde{\vartheta}}^c)^T(I_s \otimes \Gamma)^{-1}\underline{\tilde{\vartheta}}^c + (\underline{\tilde{\theta}}^c)^T(I_m \otimes \bar{\Gamma})^{-1}\underline{\tilde{\theta}}^c \\
& +(\tilde{\alpha}^c)^T(\Upsilon)^{-1}\tilde{\alpha}^c + (\tilde{\omega}^c)^T(\Lambda)^{-1}\tilde{\omega}^c + (\tilde{\alpha}^e)^T(\Upsilon)^{-1}\tilde{\alpha}^e \\
& +(\tilde{\theta}^e)^T(\bar{\Gamma})^{-1}\tilde{\theta}^e + (\tilde{\omega}^e)^T(\Lambda)^{-1}\tilde{\omega}^e + \tilde{\kappa}^T(\bar{\Upsilon})^{-1}\tilde{\kappa} \\
& +(\underline{\tilde{\vartheta}}^e)^T(I_s \otimes \Gamma)^{-1}\underline{\tilde{\vartheta}}^e \Big\}
\end{aligned} \tag{76}
$$

We have

$$
\begin{aligned}
& (\tilde{x}^e)^T \left[(I_M \otimes P)(I_M \otimes \bar{A}) + (I_M \otimes \bar{A}^T)(I_M \otimes P)\right] \tilde{x}^e \\
& \quad = \quad (\tilde{x}^e)^T \left[I_M \otimes \big(P\bar{A} + \bar{A}^T P\big)\right] \tilde{x}^e ,
\end{aligned} \tag{77}
$$

and

$$
(\tilde{x}^c)^T(\Psi \otimes P) \left[\rho\, (\Psi \otimes \bar{P}) + I_M \otimes BK\right] \tilde{x}^e = (\tilde{x}^c)^T Q\, \tilde{x}^e, \tag{78}
$$

where $Q \triangleq \rho\, (\Psi^2 \otimes P\bar{P}) + \Psi \otimes PBK$. Furthermore, similar to (38), we obtain

$$
\begin{aligned}
& 2(\tilde{x}^e)^T(I_M \otimes P)\, \tilde{g}^e \\
& \quad \le \mu\, (\tilde{x}^e)^T(I_M \otimes P^2)\, \tilde{x}^e + \frac{\sigma^2}{\mu}(\tilde{x}^e)^T \big(I_M \otimes I_n\big)\, \tilde{x}^e.
\end{aligned} \tag{79}
$$

By using (4), (48) and (49) given in assumptions 1.4, 2.1 and 2.2, respectively, using (56), and the property that $(\hat{a})^T sgn\big(\hat{a}\big) \ge |\hat{a}|$, it can be shown that

$$
\begin{aligned}
& (\tilde{x}^c)^T(\Psi \otimes P)(\xi - \bar{\xi}^c + U + \Delta) \\
& = \sum_{i=1}^{M} \Big[(\sum_{j\in N_i} b_{ij}\, \bar{D}^T G_B\, C\, \tilde{x}_{ij}^c)^T \eta_i \\
& \quad -(\sum_{j\in N_i} b_{ij}\, \bar{D}^T G_B\, C\, \tilde{x}_{ij}^c)^T \hat{\omega}_i^c \bar{\eta}_i sgn(\sum_{j\in N_i} b_{ij} \bar{D}^T G_B \tilde{y}_{ij}^c) \\
& \quad -(\sum_{j\in N_i} b_{ij} G_B C \tilde{x}_{ij}^c)^T U_i - (\sum_{j\in N_i} b_{ij}\, G_B\, C\, \tilde{x}_{ij}^c)^T \\
& \quad \cdot \hat{\kappa}_i \bar{U}_i sgn(\sum_{j\in N_i} b_{ij}\, G_B\, \tilde{y}_{ij}^c) + (\sum_{j\in N_i} b_{ij}\, \bar{F}^T G_B\, C\, \tilde{x}_{ij}^c)^T \Delta_i \\
& \quad -(\sum_{j\in N_i} b_{ij}\, \bar{F}^T G_B\, C\, \tilde{x}_{ij}^c)^T \hat{\alpha}^c \bar{\Delta}_i sgn(\sum_{j\in N_i} b_{ij} \bar{F}^T G_B \tilde{y}_{ij}^c) \Big] \\
& \le \sum_{i=1}^{M} \Bigg[\tilde{\omega}_i^c \Big|\sum_{j\in N_i} b_{ij}\, \bar{D}^T G_B\, \tilde{y}_{ij}^c\Big| \bar{\eta}_i + \tilde{\kappa}_i \Big|\sum_{j\in N_i} b_{ij}\, G_B\, \tilde{y}_{ij}^c\Big| \bar{U}_i \\
& \quad + \tilde{\alpha}_i^c \Big|\sum_{j\in N_i} b_{ij}\, \bar{F}^T G_B\, \tilde{y}_{ij}^c\Big| \bar{\Delta}_i \Bigg] ,
\end{aligned} \tag{80}
$$

where $\tilde{x}_{ij}^c = \tilde{x}_i^c - \tilde{x}_j^c = x_i - x_j$. Similarly,

$$
\begin{aligned}
& (\tilde{x}^e)^T(I_M \otimes P)(\xi - \bar{\xi}^e + U^e + \Delta) \\
& \le \sum_{i=1}^{M} \Bigg[\tilde{\omega}_i^e \left|\bar{D}^T G_B\, \tilde{y}_i^e\right| \bar{\eta}_i + \tilde{\theta}_i^e \left|G_B\, \tilde{y}_i^e\right| \cdot |u_i| \\
& \quad + \tilde{\alpha}_i^e \left|\, \bar{F}^T G_B\, \tilde{y}_i^e\right| \bar{\Delta}_i \Bigg] .
\end{aligned} \tag{81}
$$

Let us define

$$
Q^e \triangleq \bar{A}^T P + P\bar{A} + \mu P^2 + \frac{\sigma^2}{\mu} I_n \,. \tag{82}
$$

Therefore, by applying (34) – (38), and the above inequalities in (76), and using adaptive laws (60) – (64) and (52) – (55), the Lyapunov function derivative $\dot{V}$ satisfies

$$
\dot{V} \le \begin{bmatrix} \tilde{x}^c \\ \tilde{x}^e \end{bmatrix}^T \begin{bmatrix} \Psi \otimes Q^c & Q \\ 0 & I_M \otimes Q^e \end{bmatrix} \begin{bmatrix} \tilde{x}^c \\ \tilde{x}^e \end{bmatrix} , \tag{83}
$$

where Q^c and Q are defined in (44) and (78), respectively. By using positive definiteness of Ψ due to Lemma 2, and the property that eigenvalues of the matrix $\Psi \otimes Q^c$ are products of eigenvalues of Ψ and Q^c (Theorem 6 in [26]), we know $\dot{V}$ is negative semidefinite if the matrices Q^c and Q^e are negative definite (see (84) and (85)). Thus, we conclude that $\tilde{x}_i^c$, $\tilde{x}_i^e$, $\hat{\vartheta}_i^c$, $\hat{\theta}_i^c$, $\hat{\alpha}_i^c$, $\hat{\omega}_i^c$, $\hat{\vartheta}_i^e$, $\hat{\theta}_i^e$, $\hat{\alpha}_i^e$, $\hat{\omega}_i^e$, and $\hat{\kappa}_i$ are uniformly bounded. Then, the proof can be concluded by using a similar reasoning logic as reported in the analysis of Theorem 1.

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

Theorem 2: If there exists a symmetric positive definite matrix $P \in \Re^{n\times n}$, a matrix $G_B \in \Re^{m\times l}$, positive constants ρ and μ such that (56) and the following LMIs are satisfied:

$$
\tilde{A}^T P + P\tilde{A} + \mu P^2 + \frac{\sigma^2}{\mu} I_n - 2\rho\, \gamma\, PBB^T P < 0 \,, \tag{84}
$$

$$
\bar{A}^T P + P\bar{A} + \mu P^2 + \frac{\sigma^2}{\mu} I_n < 0 \,, \tag{85}
$$

where I_n is the identity matrix, and σ is the Lipschitz constant defined in (3). Then, the state estimator (51) – (55) and the adaptive control law (58) – (64) with distributed controller gains given by (23) guarantee the following properties:

1) All the signals are uniformly bounded.
2) The leader-follower consensus is achieved asymptotically with a time-varying leader state, i.e., $x_i(t) - x_0(t) \to 0$ as $t \to \infty$.
3) The state estimation error for each distributed agent converges to zero asymptotically, i.e., $\hat{x}_i(t) - x_i(t) \to 0$ as $t \to \infty$.

Remark 5: Two neural network based adaptive approximators are employed in the FTC method, including $\hat{f}_i^e(y_i, \hat{\vartheta}_i^e(t))$ in the state estimator (51) and $\hat{f}_i^c(y_i, \hat{\vartheta}_i^c(t))$ in the control law (59). Note that the objective of the approximator $\hat{f}_i^e$ is to learn the unknown fault function, while the objective of $\hat{f}_i^c$ is to modify the feedback control law via parameter adaptation so as to stabilize the system and guarantee leader-following performance in the presence of faults. Hence, the adaptive designs are different (see (52) and (60)). Furthermore, conditions (84), (85), and (56) can be transformed into standard linear matrix inequalities. Then, a feasible solution to (84), (85), and (56) can be possibly found by using LMI tools.

IV. Simulation Results

In this section, a simulation example of a networked multi-agent system consisting of 5 flexible link robotic arms is considered to illustrate the effectiveness of the distributed FTC method developed for two cases, i.e., with only limited output measurement and with full-state measurement, respectively.

A. Case 1: Input-Output Agents

The dynamics of each agent given in [11], [29] can be easily put into the general form (46), where the state of the ith agent $x_i \in \Re^4$, for $i = 1, \cdots, 5$, is consisting of the motor position, motor velocity, link position and link velocity, respectively, and u_i is the input of the ith agent representing the motor torque. Specifically, we have

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -48.6 & -1.25 & 48.6 & 0 \\ 0 & 0 & 0 & 10 \\ 19.5 & 0 & -19.5 & 0 \end{bmatrix}, \; C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix},$$

$B = [0\,, 21.6\,, 0\,, 0]^T$, $D = F = [0\,, 1\,, 0\,, 0]^T$, and the nominal nonlinear term $g(x_i) = [0,\, 0,\, 0,\, -0.333 sin(x_{i3})]^T$. The modeling uncertainty in the dynamics is assumed to be an unmodeled Coulomb friction in the motor given by $\eta_i = -1.5 sgn(x_{i2})$, which is bounded by unknown constant ω_i (i.e., $\bar{\eta}_i = 1$ in (48)). By using the LMI toolbox, we obtain

$$P = \begin{bmatrix} 515 & 18.2 & -442 & 59.2 \\ 18.2 & 1.24 & -16.1 & 0 \\ -442 & -16.1 & 410.5 & -39 \\ 59.2 & 0 & -39 & 101.8 \end{bmatrix}, \quad \rho = 0.1\,,$$

$$L = \begin{bmatrix} 0 & 0 & 0 \\ 432 & 41 & -368 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \; K = [20\,,\, 1.9\,,\, -17\,,\, -8.4]\,,$$

$$G_B = [393.12\,, \quad 26.78\,, \quad -347.76]\,,\; \mu = 0.01\,.$$

The objective is to have each agent follow a virtual leader given by $\dot{x}_0 = g(x_0) + Ax_0 + Bu_0$ with zero initial condition and the input $u_0 = sin(0.1t)$. The intercommunication graph of agents is shown in Figure 1. As can be seen, the virtual leader's command is only communicated with the second agent (i.e., $k_{20} = 1$). We choose $\bar{k}_2 = 0.5$. Then, the left eigenvector of Ψ associated with the zero eigenvalue is $\bar{\chi} = [0.425, 0.142, 0.212, 0.402, 0.521, 0.566]^T$. The matrix Ψ defined in Lemma 2 has the minimum eigenvalue of $\gamma = 0.072$ and the maximum eigenvalue of $\varrho = 10.84$.

The two adaptive approximators (i.e., $\hat{f}_i^e(y_i, \hat{\vartheta}_i^e(t))$ in the state estimator (51) and $\hat{f}_i^c(y_i, \hat{\vartheta}_i^c(t))$ in the control law (58) are both implemented as radial basis function (RBF) neural networks. Each RBF network consists of 5 neurons with 5 adjustable parameters. The center of radial basis functions are equally distributed on interval $[-3, 3]$ with a variance of 1. The initial values of the parameter vector is set to zero. We set the learning rates as $\Gamma_i = 3$ and $\bar{\Gamma}_i = 0.2$ and consider an unknown constant bound on the network approximation error, i.e., $\bar{\delta}_i = 1$. The learning rates are chosen as $\Upsilon_i = 1.5$, $\bar{\Upsilon}_i = 3$, and $\Lambda_i = 0.5$.

We consider an actuator fault with a magnitude of $\theta_1 = -0.65$ and a process fault leading to extra abnormal viscous friction in the motor (i.e., $f_i(x_i) = -1.5\, x_{i2}$) that occur abruptly (i.e., β_i is a step function) to agent 1 at $T_{iu} = 10$ second and $T_{if} = 20$ second, respectively. Note that for $t \geq 20$ second, both faults are simultaneously affecting the local agent dynamics. Regarding the performance of the FTC schemes, as can be seen from Figure 2, the leader-following consensus is achieved using the proposed adaptive FTC even in the presence of faults, while the agents cannot achieve the leader-following consensus without the FTC controller (see Figure 3).

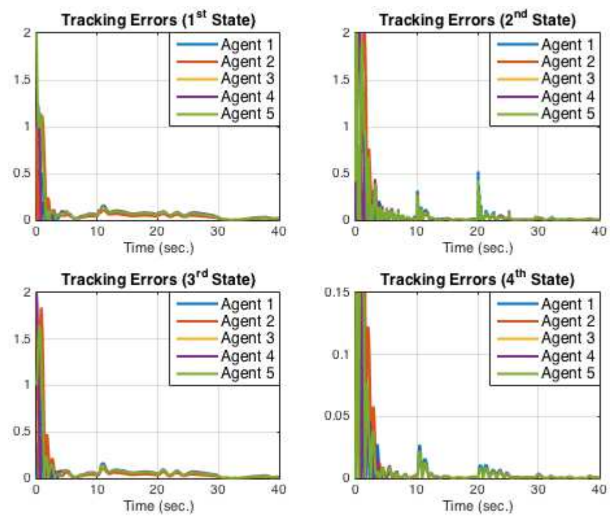

Fig. 2. Tracking errors with distributed adaptive FTC: input-output agents

B. Case 2: Full-State Measurements

The dynamics of each agent given in [11] can be easily put into the general form (1), where the matrices A, B, D, F, the nominal nonlinear term $g(x_i)$, the unmodeled Coulomb friction in the motor given by η_i, the design matrices P and K, and the design constant μ are given in Section IV-A.

The virtual leader is given by $\dot{x}_0 = g(x_0) + Ax_0 + Bu_0$ with zero initial condition and the input $u_0 = sin(0.1t)$, and the intercommunication graph of agents is shown in Figure 1. The adaptive approximator $\hat{f}_i(x_i, \hat{\vartheta}_i(t))$ in the control law (10) is implemented as radial basis function (RBF) neural networks, where each RBF network consists of 5 neurons with 5 adjustable parameters. The center of radial basis functions

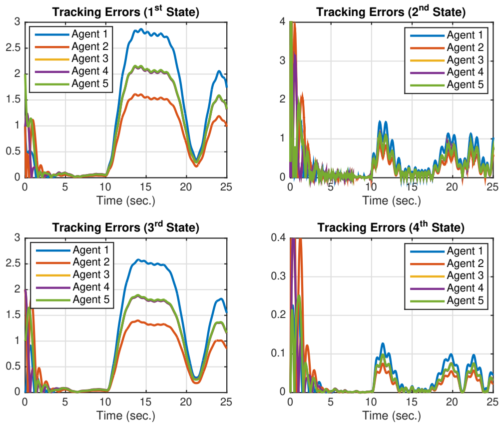

Fig. 3. Tracking errors without distributed adaptive FTC: input-output agents

are equally distributed on interval $[-3, 3]$ with a variance of 1. The initial values of the parameter vector is set to zero. We use the same learning rates as given in Section IV-A. The learning rate $\bar{\Lambda}_i = 0.5$ is also chosen.

We consider an actuator fault with a magnitude of $\theta_1 = -0.65$ and a process fault leading to extra abnormal viscous friction in the motor (i.e., $f_i(x_i) = -1.5\, x_{i2}$) that occur abruptly (i.e., β_i is a step function) to agent 1 at $T_{iu} = 10$ second and $T_{if} = 20$ second, respectively. As can be seen from Figure 4, the leader-following consensus is achieved using the proposed adaptive FTC even in the presence of faults, while the agents cannot achieve the leader-following consensus without the FTC controller (see Figure 5).

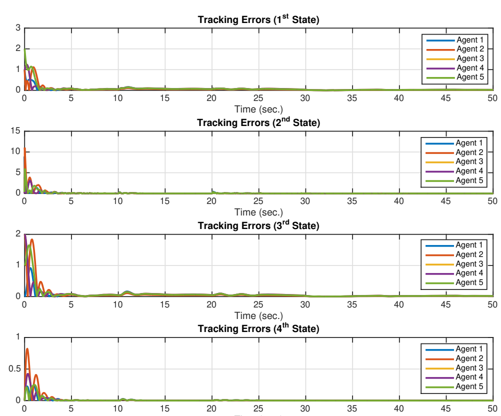

Fig. 4. Tracking errors with distributed adaptive FTC: full-state measurements

V. Conclusions

In this paper, the problem of distributed FTC design for a class of high-order nonlinear uncertain multi-agent systems under a bidirectional intercommunication topology with asymmetric weights is investigated. The FTC schemes are developed for two cases, i.e., with full-state measurement and with only limited output measurement, respectively. Adaptive learning algorithms are developed to maintain leader-following consensus with a time-varying leader, even in the presence of process and actuator faults. For the case of partial state measurement, the minimum eigenvalue of the matrix associated with the communication topology is needed, which is considered topology global information. The extensions to directed communication links among the followers, more general agent models, and removing the topology global information in the case of partial state measurement are interesting topics for future research.

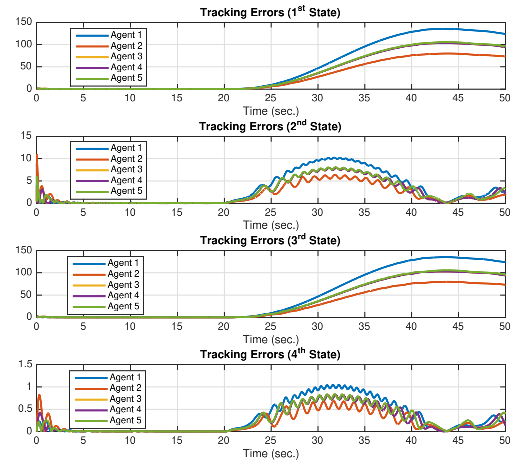

Fig. 5. Tracking errors without distributed adaptive FTC: full-state measurements

References

[1] W. Ren and R. Beard, *Distributed Consensus in Multi-vehicle Cooperative Control: Theory and Applications*. London: Springer-Verlag, 2008.

[2] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods for Multi-agent Networks*. Princeton, NJ: Princeton University Press, 2010.

[3] Z. Qu, *Cooperative control of dynamical systems: applications to autonomous vehicles*. New York: Springer-Verlag, 2009.

[4] 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, no. 8, pp. 1351–1358, 2010.

[5] A. Das and F. L. Lewis, "Distributed adaptive control for synchronization of unknown nonlinear networked systems," *Automatica*, vol. 46, no. 12, pp. 2014–2021, 2010.

[6] G. Chen and Y. D. Song, "Cooperative tracking control of nonlinear multi-agent systems using self-structuring neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 8, pp. 1496–1507, 2014.

[7] 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.

[8] W. Liu and J. Huang, "Cooperative adaptive output regulation for second-order nonlinear multi-agent systems with jointly connected switching networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, pp. 695–705, 2017.

[9] 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.

[10] Q. Shen, B. Jiang, P. Shi, and J. Zhao, "Cooperative adaptive fuzzy tracking control for networked unknown nonlinear multi-agent systems with time-varying actuator faults," *IEEE Transactions on Fuzzy Systems*, vol. 22, pp. 494–504, 2013.

[11] Z. Li, W. Ren, X. Liu, and M. Fu, "Consensus of multi-agent systems with general linear and lipschitz nonlinear dynamics using distributed adaptive protocols," *IEEE Transactions on Automatic Control*, vol. 58, pp. 1786–1791, 2013.

[12] 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, no. 6, pp. 3923–3931, 2015.

[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] G. Chen, F. L. Lewis, and L. Xie, "Finite-time distributed consensus via binary control protocols," *Automatica*, vol. 47, pp. 1962–1968, 2011.

[16] 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.

[17] M. Khalili, X. Zhang, M. M. Polycarpou, T. Parisini, and Y. Cao, "Distributed adaptive fault-tolerant control of uncertain multi-agent systems," *Automatica*, vol. 87, pp. 142–151, 2018.

[18] M. Khalili, X. Zhang, Y. Cao, M. M. Polycarpou, and T. Parisini, "Distributed adaptive fault–tolerant leader–following formation control of nonlinear uncertain second–order multi–agent systems," *International Journal of Robust and Nonlinear Control*, vol. 28, pp. 4287–4308, 2018.

[19] M. Khalili, X. Zhang, Y. Cao, M. M. Polycarpou, and T. Parisini, "Distributed adaptive fault-tolerant control of a class of high-order nonlinear uncertain multi-agent systems," in *Proceedings of the 56th IEEE Conference on Decision and Control*, (Melbourne, Australia), December 2017.

[20] 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, no. 4, pp. 576–593, 2002.

[21] Z. Li, X. Liu, W. Ren, and L. Xie, "Distributed tracking control for linear multi-agent systems with a leader of bounded unknown input," *IEEE Transactions on Automatic Control*, vol. 58, no. 2, pp. 518–523, 2013.

[22] X. Li and G. Yang, "Adaptive fault-tolerant synchronization control of a class of complex dynamical networks with general input distribution matrices and actuator faults," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 3, pp. 559–569, 2017.

[23] C. Edwards and S. Spurgeon, *Sliding mode control: theory and applications*. Crc Press, 1998.

[24] J. Farrell and M. M. Polycarpou, *Adaptive Approximation Based Control*. Hoboken, NJ: J. Wiley, 2006.

[25] C. T. Chen, *Linear System Theory and Design*. New York, NY: Oxford University Press, 1999.

[26] H. Zhang and F. Ding, "On the kronecker products and their applications," *Journal of Applied Mathematics*, vol. 86, pp. 1–8, 2013.

[27] P. A. Ioannou and J. Sun, *Robust Adaptive Control*. Englewood Cliffs, NJ: Prentice Hall, 1996.

[28] M. Khalili, *Distributed Adaptive Fault-Tolerant Control of Nonlinear Uncertain Multi-Agent Systems*. PhD thesis, Wright State University, 2017.

[29] R. Rajamani and Y. M. Cho, "Existence and design of observers for nonlinear systems: Relation to distance to unobservability," *International Journal of Control*, vol. 69, no. 5, pp. 717–731, 1998.