CrossMark

Finite-Time Unknown Input Observer-Based Distributed Fault Diagnosis for Multi-agent Systems with Disturbances

Xingxing Chen[1] · **Ke Zhang**[1] · **Bin Jiang**[1]

Abstract In this study, based on the finite-time unknown input observer (FT-UIO), the problem of distributed fault diagnosis (FD) for multi-agent systems (MAS) with actuator faults and disturbances is investigated. Firstly, using the communication topology and local information, the global multi-agent system dynamics are represented with respect to a certain agent. Then, in order to construct a FT-UIO at the selected agent, the system will be decomposed into three subsystems after coordinate transformation. Furthermore, constituted by two distinct observers, the FT-UIO is designed to converge in a pre-assigned finite time, which is chosen as the time delay of the observers. With the state estimation obtained from the FT-UIO in the selected agent, a FD algorithm is also proposed to diagnose the faults occurred in its neighbor agents. Finally, simulation results are presented to illustrate the effectiveness and advantages of the proposed FT-UIO-based distributed FD scheme for MAS.

Keywords Multi-agent systems · Fault diagnosis · Finite time · Unknown input observer

---

✉ Ke Zhang
  kezhang@nuaa.edu.cn

  Xingxing Chen
  xxchen9952@163.com

  Bin Jiang
  binjiang@nuaa.edu.cn

[1] College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China

Birkhäuser

1 Introduction

As a powerful tool to handle complicated system, multi-agent technology has been the focus of many researchers and has been applied in many significance fields, for instance, in automotive control systems [17], satellite constellations [19], unmanned aerial vehicles [18], [14], and so on. In order to investigate the issue of MAS, there have been a considerable number of fruitful results [8,13,22], which include consensus problems, formation control, and cooperative control.

The structure of MAS is distributed and there isn't a centralized entity to monitor the whole system. Thus, if an agent becomes faulty, the fault may spread to other agents through the special communication topology in the MAS and then the performance of the entire system will suffer a difficult situation. Consequently, in order to improve the safety and reliability of system, the FD scheme, including fault detection, fault isolation, and fault estimation (FE), must be considered. FD technology is mainly used to monitor the system state, determine the location of the occurred fault and eventually obtain detailed fault information. Its application to a wide range of industrial and commercial processes has attracted a great deal of attention for several decades [2,6,16,20,26].

It should be noted that most of the existing works on FD are based on centralized systems. Owing to the fact that the MAS is a distributed system, where all of the measurements are not available to each agent, it makes these works on FD difficult to be directly applied to MAS. Therefore, distributed FD schemes should be proposed to deal with the FD for the MAS instead of the centralized FD method. In [24], for a class of interconnected nonlinear systems, the method of distributed FD was proposed. Qin et al. [15] presented a survey of FD for swarm systems. A dynamic neural network-based fault detection and isolation (FDI) scheme was studied to perform the formation flying mission of satellites in [21]. And in [25], for MAS with directed graphs, the design of adaptive technique-based distributed FE observer was put forward.

A model-based scheme is built upon the precise mathematical model, whereas the presence of uncertainties and external disturbances is not avoidable in practical systems. Thus, to weaken or eliminate such unexpected effects of uncertainties, the design of unknown input observer (UIO) has been extensively studied because of its insensitivity to the unknown input. An UIO has also provided a useful method to achieve FD with robustness against unknown inputs, in which the residual is designed to be insensitive to unknown inputs and some researchers pay more attention to the UIO-based FD approaches [7,11,28].

The stability of a system remains in a priority position in the processes of analysis and design. In general, the stability which we often said in the field of dynamical systems, such as the Lyapunov stability and bounded-input bounded-output stability are all asymptotically stable. In many applications, however, it is desirable that the trajectories of a dynamical system can converge to a equilibrium state in finite time rather than merely asymptotically. Besides, the state of the asymptotically stable system is not limited in a circle, as long as it can be bounded. In these cases, the concept of the finite-time stability (FTS) is put forward, which is different from traditional notions of stability and has been focused on the convergence performance and transient performance of the system. Some early results on FTS are shown

in [3,23]. In [12], FTS for time-varying nonlinear dynamical systems is proposed. Extensions of FTS to discrete switched singular positive systems can be found in [9].

In these observer-based FD schemes, the effect of FD depends mainly on the steady-state performance of the observer without taking the transient performance into consideration. However, owing to the fact that MAS is a high cost as well as high investment technique, its FD technique has to be accurate and fast. Hence, both transient performance and convergence performance are required by the FD for MAS. So far, despite a few academic results on application of FTS for MAS in terms of consensus problems and cooperative control [4,5,10,27], the body of application of FTS for FD research on MAS is much smaller because such research is more challenging.

Motivated by the above discussion, the problem of FT-UIO-based distributed FD for MAS with actuator faults and disturbances is put forward in this paper. And different from the traditional way of constructing observers in every agent, the method proposed in this paper is putting one observer in a certain agent to diagnose the faults occurred in its neighbor agents, so that only with one or two observers, the distributed FD for MAS can also be achieved. The main contributions of this study are summarized in four aspects: (I) under the communication topology, the global multi-agent system dynamics is represented with respect to a certain agent and the output of the system becomes the local information which can be used by the selected agent; (II) based on the system dynamics obtained, by coordinate transformation, the system will be decomposed into three subsystems and one of them is independent of faults as well as disturbances, which can improve the robustness of FD ; (III) with the reduced-order system obtained in (II), the FT-UIO, constituted by two distinct UIOs, is constructed at the selected agent and is designed to converge in a pre-assigned finite time, which is chosen as the time delay of the UIOs; (IV) using the state estimation from the FT-UIO in the selected agent, a FD algorithm is also proposed to diagnose the faults occurred in its neighbor agents, and then not only the location of the faults but also the detailed information of the faults can be obtained in a finite time.

The remainder of this paper is organized as follows. Preliminaries and problem formulation are presented in Sect. 2. Section 3 contains the procedure of system decomposition and the design of FT-UIO-based distributed FD for MAS. Simulation results are provided to illustrate the effectiveness of the proposed methods in Sect. 4. Finally, a conclusion is made in Sect. 5.

In this paper, some notations are used as follows. B^{T} denotes the transpose of a matrix B. $\otimes$ stands for the Kronecker product. $|N|$ means the cardinality of the set N. $C^{\dagger}$ stands for the left pseudoinverse of a matrix C. I denotes an identity matrix with appropriate dimension.

2 Preliminaries and Problem Formulation

2.1 Graph Theory

Consider a weighted undirected graph $\mathcal{G}$ of a network, in which $V = \{v_0, v_1, v_2 \ldots, v_N\}$ denotes the set of nodes, and $\varepsilon = \{(v_i, v_j) : v_i, v_j \in v\}$ denotes the set of edges. Denote the associated adjacency matrix $\Lambda = [a_{ij}] \in \mathbb{R}^{N \times N}$, where a_{ij} means the weight of an edge (v_i, v_j) and $a_{ij} = 1$ if $(v_i, v_j) \in \varepsilon$, otherwise $a_{ij} = 0$. For the undirected graph used by this paper, we define $a_{ii} = 0$ without considering the connectivity of the node itself. The neighborhood set of the node v_p is denoted by $N_p = \{v_j \in V : (v_p, v_j) \in \varepsilon, p \neq j\}$. Define $G = \text{diag}(g_1, \ldots, g_N)$, $g_i = \sum_{j=1}^{N} a_{ij}$ as the degree matrix and the Laplacian matrix of $\mathcal{G}$ as $L = G - \Lambda$.

2.2 Problem Formulation

Consider a single system with actuator fault of each agent labeled by node v_i in the graph $\mathcal{G}$ as follows:

$$\begin{cases} \dot{x}_i(t) = Ax_i(t) + Bu_i(t) + Ef_i(t) + Dd_i(t) \\ y_i(t) = Cx_i(t) \end{cases} \tag{1}$$

where $x_i(t) \in \mathbb{R}^n$ is the state, $u_i(t) \in \mathbb{R}^m$ is the input, $y_i(t) \in \mathbb{R}^{p_1}$ is the output, $f_i(t) \in \mathbb{R}^{q_1}$ represents the actuator fault and $d_i(t) \in \mathbb{R}^{l_1}$ is the external disturbance and model uncertainties [i.e. $\Delta Ax_i(t) + \Delta Bu_i(t)$] [1]. A, B, E, C, and D are constant real matrices of appropriate dimensions. It is assumed that matrices E and D are of full column rank.

Based on undirected graph theory, the global fault multi-agent system dynamics can be represented as

$$\begin{cases} \dot{x}(t) = (I_N \otimes A)x(t) + (I_N \otimes B)u(t) + (I_N \otimes E)f(t) + (I_N \otimes D)d(t) \\ y(t) = (I_N \otimes C)x(t) \end{cases} \tag{2}$$

where

$$x(t) = \left[x_1^T(t), x_2^T(t), \ldots, x_N^T(t)\right]^T, u(t) = \left[u_1^T(t), u_2^T(t), \ldots, u_N^T(t)\right]^T,$$

$$y(t) = \left[y_1^T(t), y_2^T(t), \ldots, y_N^T(t)\right]^T, f(t) = \left[f_1^T(t), f_2^T(t), \ldots, f_N^T(t)\right]^T,$$

$$d(t) = \left[d_1^T(t), d_2^T(t), \ldots, d_N^T(t)\right]^T.$$

With respect to the agent p, the systems are rewritten as

$$
\begin{cases}
\dot{x}(t) = (I_N \otimes A)x(t) + (I_N \otimes B)u(t) + (I_N \otimes E)_r f_r(t) + (I_N \otimes E)_{-r} f_{-r}(t) \\
\qquad + (I_N \otimes D)d(t) \\
y_p(t) = (\Lambda_p \otimes I_n)x(t)
\end{cases}
$$

(3)

where $f_r(t)$ is the fault occurred in agent r ($r \in N_p$), and $f_{-r}(t)$ is the matrix obtained by removing component $f_r(t)$ of $f(t)$. Correspondingly, $(I_N \otimes E)_r$ is the distribution matrix of $f_r(t)$ as well as the submatrix of $I_N \otimes E$. And $(I_N \otimes E)_{-r}$, the distribution matrix of $f_{-r}(t)$, is the surplus of $I_N \otimes E$ after removing $(I_N \otimes E)_r$. $y_p(t) \in \mathbb{R}^{|\overline{N}_p|n \times Nn}$ is the local information which can be used by agent p. $|\overline{N}_p|$ denotes the cardinality of the set $\overline{N}_p$, where $\overline{N}_p = N_p \cup \{p\}$, i.e., the set $\overline{N}_p$ consists of agent p and its neighbor agents. $\Lambda_p \otimes I_n$ is the local information matrix of full row rank. The matrix $\Lambda_p \in \mathbb{R}^{|\overline{N}_p| \times N}$ is also full row rank and is a submatrix of the associated adjacency matrix Λ.

Define $M = \begin{bmatrix} (I_N \otimes E)_{-r} & I_N \otimes D \end{bmatrix}$, $\omega(t) = \begin{bmatrix} f_{-r}(t) \\ d(t) \end{bmatrix}$, then (3) can be reorganized as follows

$$
\begin{cases}
\dot{x}(t) = (I_N \otimes A)x(t) + (I_N \otimes B)u(t) + (I_N \otimes E)_r f_r(t) + M\omega(t) \\
y_p(t) = (\Lambda_p \otimes I_n)x(t)
\end{cases}
$$

(4)

Due to the fact that $y_p(t)$ is the local information which can be used by agent p, through an UIO technology, we can construct an observer in agent p to estimate the state in system (4). Then with a FD algorithm, the estimation of $f_r(t)$ can be obtained.

3 Main Results

3.1 System Decomposition

For the sake of distributed finite-time FD for MAS with actuator faults $f(t)$ and disturbances $d(t)$, a FT-UIO will be constructed in the special agent p with respect to its neighbor agent r. Owing to the fact that $\omega(t)$ is consisted by $f_{-r}(t)$ and $d(t)$, one can obtain a reduced-order system which is independent of faults as well as disturbances by regarding the term $f_r(t)$ and $\omega(t)$ as unknown inputs. Through the following coordinate transformation, the system (4) can be decomposed into three subsystems.

Assumption 1 In this paper, we assume that M and $(I_N \otimes E)_r$ are of full column rank. Note that $\Lambda_p \otimes I_n$ is of full row rank, we can define $\operatorname{rank}(\Lambda_p \otimes I_n) = p$, $\operatorname{rank}(M) = l$, and $\operatorname{rank}((I_N \otimes E)_r) = q$, ($l + q \leq p$), then it can be assumed that $rank\begin{bmatrix} M & (I_N \otimes E)_r \end{bmatrix} = l + q$, $rank((\Lambda_p \otimes I_n)M) = l$, $\operatorname{rank}((\Lambda_p \otimes I_n)(I_N \otimes E)_r) = q$, and $\operatorname{rank}\begin{bmatrix} (\Lambda_p \otimes I_n)M & (\Lambda_p \otimes I_n)(I_N \otimes E)_r \end{bmatrix} = l + q$. So that the matrix $\begin{bmatrix} (\Lambda_p \otimes I_n)M & (\Lambda_p \otimes I_n)(I_N \otimes E)_r \end{bmatrix}$ has the left pseudoinverse $\begin{bmatrix} (\Lambda_p \otimes I_n)M & (\Lambda_p \otimes I_n)(I_N \otimes E)_r \end{bmatrix}^\dagger$, where $X^\dagger = (X^T X)^{-1} X^T$ and $X^\dagger X = I$.

Next, for the purpose of the state coordinate transformation, there exists a non-singular matrix

$$T = \begin{bmatrix} N & M & (I_N \otimes E)_r \end{bmatrix}, \quad T \in \mathbb{R}^{Nn \times Nn}, \tag{5}$$

where $N = P_1 N_0$, $N \in \mathbb{R}^{Nn \times (Nn-l-q)}$, and N_0 is chosen as an arbitrary matrix to make $\begin{bmatrix} N_0 & M & (I_N \otimes E)_r \end{bmatrix}$ non-singular. The square matrix $P_1 = I_{Nn} - \begin{bmatrix} M & (I_N \otimes E)_r \end{bmatrix} [(\Lambda_p \otimes I_n) M \ (\Lambda_p \otimes I_n)(I_N \otimes E)_r]^{\dagger} (\Lambda_p \otimes I_n)$ satisfies $P_1^2 = P_1$, which means P_1 is idempotent. As long as if the matrix $\begin{bmatrix} N_0 & M & (I_N \otimes E)_r \end{bmatrix}$ is non-singular, the matrix T is also non-singular.

With the transformation matrix T, the system (4) can be decomposed into the following form

$$\begin{cases} \dot{\overline{x}}(t) = \overline{A}\overline{x}(t) + \overline{B}u(t) + \overline{E} f_r(t) + \overline{M}\omega(t) \\ y_p(t) = \overline{C}_p \overline{x}(t) \end{cases} \tag{6}$$

where

$$x(t) = T\overline{x}(t) = T \begin{bmatrix} \overline{x}_1(t) \\ \overline{x}_2(t) \\ \overline{x}_3(t) \end{bmatrix}, \overline{A} = T^{-1}(I_N \otimes A)T = \begin{bmatrix} \overline{A}_{11} & \overline{A}_{12} & \overline{A}_{13} \\ \overline{A}_{21} & \overline{A}_{22} & \overline{A}_{23} \\ \overline{A}_{31} & \overline{A}_{32} & \overline{A}_{33} \end{bmatrix}$$

$$\overline{B} = T^{-1}(I_N \otimes B) = \begin{bmatrix} \overline{B}_1 \\ \overline{B}_2 \\ \overline{B}_3 \end{bmatrix}, \overline{M} = T^{-1}M = \begin{bmatrix} 0 \\ I_l \\ 0 \end{bmatrix}, \overline{E} = T^{-1}(I_N \otimes E)_r = \begin{bmatrix} 0 \\ 0 \\ I_q \end{bmatrix}$$

$$\overline{C}_p = (\Lambda_p \otimes I_n)T = \begin{bmatrix} (\Lambda_p \otimes I_n)N & (\Lambda_p \otimes I_n)M & (\Lambda_p \otimes I_n)(I_N \otimes E)_r \end{bmatrix}$$

with $\overline{x}_1(t) \in \mathbb{R}^{Nn-l-q}, \overline{x}_2(t) \in \mathbb{R}^l, \overline{x}_3(t) \in \mathbb{R}^q$.

Obviously, the differential equations of $\overline{x}_2(t)$ and $\overline{x}_3(t)$ are affected directly by the terms of $\omega(t)$ and $f_r(t)$, respectively. Removing the differential equations of $\overline{x}_2(t)$ and $\overline{x}_3(t)$, the system without $\omega(t)$ and $f_r(t)$ is written as follows

$$\begin{cases} \begin{bmatrix} I_{Nn-l-q} & 0 & 0 \end{bmatrix} \dot{\overline{x}}(t) = \begin{bmatrix} \overline{A}_{11} & \overline{A}_{12} & \overline{A}_{13} \end{bmatrix} \overline{x}(t) + \overline{B}_1 u(t) \\ y_p(t) = \begin{bmatrix} (\Lambda_p \otimes I_n)N & (\Lambda_p \otimes I_n)M & (\Lambda_p \otimes I_n)(I_N \otimes E)_r \end{bmatrix} \overline{x}(t) \end{cases} \tag{7}$$

Furthermore, in order to achieve the output transformation, there exists a non-singular matrix

$$U = \begin{bmatrix} (\Lambda_p \otimes I_n)M & (\Lambda_p \otimes I_n)(I_N \otimes E)_r & Q \end{bmatrix}, U \in \mathbb{R}^{p \times p} \tag{8}$$

where $Q = P_2 Q_0$, $Q \in \mathbb{R}^{p \times (p-l-q)}$, and Q_0 is chosen as an arbitrary matrix to make $\begin{bmatrix} (\Lambda_p \otimes I_n)M & (\Lambda_p \otimes I_n)(I_N \otimes E)_r & Q_0 \end{bmatrix}$ non-singular. The square matrix $P_2 = I_p - \Lambda_p \otimes I_n \begin{bmatrix} M & (I_N \otimes E)_r \end{bmatrix} [(\Lambda_p \otimes I_n)M \ (\Lambda_p \otimes I_n)(I_N \otimes E)_r]^{\dagger}$ satisfies $P_2^2 = P_2$, which means P_2 is idempotent. As long as the matrix $\begin{bmatrix} (\Lambda_p \otimes I_n)M & (\Lambda_p \otimes I_n)(I_N \otimes E)_r & Q_0 \end{bmatrix}$ is non-singular, the matrix U is also non-singular.

Define

$$U^{-1} = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix}, U_1 \in \mathbb{R}^{l \times p}, U_2 \in \mathbb{R}^{q \times p}, U_3 \in \mathbb{R}^{(p-l-q) \times p},$$

and we can obtain

$$U^{-1}U = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix} \left[(\Lambda_p \otimes I_n)M \ (\Lambda_p \otimes I_n)(I_N \otimes E)_r \ Q \right] = \begin{bmatrix} I_l & 0 & 0 \\ 0 & I_q & 0 \\ 0 & 0 & I_{p-l-q} \end{bmatrix}.$$

Lemma 1 [7] *From the information above, it also holds that* $U_1(\Lambda_p \otimes I_n)N = 0_{l \times (Nn-l-q)}$ *and* $U_2(\Lambda_p \otimes I_n)N = 0_{q \times (Nn-l-q)}$.

Then, multiplied by U^{-1} *on both sides of the output equation in system* (7) *, the following equations can be obtained*

$$U_1 y_p(t) = \overline{x}_2(t) \tag{9}$$

$$U_2 y_p(t) = \overline{x}_3(t) \tag{10}$$

$$U_3 y_p(t) = U_3(\Lambda_p \otimes I_n)N\overline{x}_1(t) \tag{11}$$

Note that the differential equations of $\overline{x}_2(t)$ and $\overline{x}_3(t)$ are affected directly by the terms of $\omega(t)$ and $f_r(t)$, respectively. Meanwhile, these vectors can be represented only by output measurements. Hence, by substituting the relations (9), (10) and (11) into the system (7), the reduced-order system independent of the unknown inputs is obtained as follows

$$\begin{cases} \dot{\overline{x}}_1(t) = \overline{A}_{11}\overline{x}_1(t) + \overline{B}_1 u(t) + \widetilde{E} y_p(t) \\ \overline{y}_p(t) = \widetilde{C}_p \overline{x}_1(t) \end{cases} \tag{12}$$

where $\widetilde{E} = \overline{A}_{12}U_1 + \overline{A}_{13}U_2$, $\widetilde{C}_p = U_3(\Lambda_p \otimes I_n)N$, and $\overline{y}_p(t) = U_3 y_p(t)$.

By regarding the term $f_r(t)$ and $\omega(t)$ as unknown inputs, the system (4) has been decomposed into three subsystems. The subsystem (12) we obtained is obviously independent of faults and disturbances, which can improve the robustness of FD. Then, based on (12), the FT-UIO, constituted by two distinct observers, can be designed in the special agent p with respect to its neighbor agent r.

3.2 FT-UIO-Based Fault Diagnosis Design

It is assumed that the pair $(\overline{A}_{11}, \widetilde{C}_p)$ is observable, then the two UIOs are described by

$$\dot{z}_1(t) = (\overline{A}_{11} - L_1\widetilde{C}_p)z_1(t) + \overline{B}_1 u(t) + L_1^* y_p(t) \tag{13}$$

$$\dot{z}_2(t) = (\overline{A}_{11} - L_2\widetilde{C}_p)z_2(t) + \overline{B}_1 u(t) + L_2^* y_p(t) \tag{14}$$

where $L_1^* = L_1 U_3 + \widetilde{E}$ and $L_2^* = L_2 U_3 + \widetilde{E}$. And the matrices L_1 and L_2 are the gain of observers to be designed.

By defining

$$z(t) = \begin{bmatrix} z_1(t) \\ z_2(t) \end{bmatrix}, \quad G = \begin{bmatrix} \overline{B}_1 \\ \overline{B}_1 \end{bmatrix}, \quad H = \begin{bmatrix} L_1^* \\ L_2^* \end{bmatrix},$$

$$F = \begin{bmatrix} F_1 & 0 \\ 0 & F_2 \end{bmatrix} = \begin{bmatrix} \overline{A}_{11} - L_1 \widetilde{C}_p & 0 \\ 0 & \overline{A}_{11} - L_2 \widetilde{C}_p \end{bmatrix},$$

these two observers can be combined in one equation. Using a delay τ to estimate the state $\overline{x}_1(t)$, the FT-UIO constructed in the special agent p with respect to its neighbor agent r is obtained as follows

$$\begin{cases} \dot{z}(t) = Fz(t) + Gu(t) + Hy_p(t), \, t \geq t_0 \\ \hat{\overline{x}}_1(t) = K[z(t) - e^{F\tau} z(t - \tau)] \end{cases} \tag{15}$$

Due to the delay, this observer has initial conditions for $z(t)$, $t \in [t_0 - \tau, t_0]$. Without loss of generality, we assume $z(t) = S \hat{\overline{x}}_1(t_0)$, $\forall t \in [t_0 - \tau, t_0]$, where $\hat{\overline{x}}_1(t_0) = \begin{bmatrix} I_{Nn-l-q} & 0 & 0 \end{bmatrix} T^{-1} \hat{x}(t_0)$ and $\hat{x}(t_0)$ is arbitrary but bounded.

Theorem 1 *After defining* $K = \begin{bmatrix} I_{Nn-l-q} & 0_{Nn-l-q} \end{bmatrix} \cdot \begin{bmatrix} S & e^{F\tau} S \end{bmatrix}^{-1}$ *and* $S = \begin{bmatrix} I_{Nn-l-q} \\ I_{Nn-l-q} \end{bmatrix}$, *if there exist a matrix* H *and an non-negative scalar* τ *such that the following conditions hold*

i) F is stable,
ii) $det[S, e^{F\tau} S] \neq 0$,

then the FT-UIO (15) *exists and can estimate the state* $\overline{x}_1(t)$ *of the reduced-order system* (12) *in finite time* τ *with a bounded estimation error* $e(t) = \hat{\overline{x}}_1(t) - \overline{x}_1(t)$ *($\forall t \in [t_0, t_0 + \tau]$).*

Proof The proof is consisted by two parts. In the first place, the boundedness of the estimation error $e(t) = \hat{\overline{x}}_1(t) - \overline{x}_1(t)$ during the time interval $[t_0, t_0 + \tau]$ is certified. Secondly, it is proved that the FT-UIO can converge in finite time τ, i.e. $\hat{\overline{x}}_1(t) = \overline{x}_1(t)$ $\forall t \geq t_0 + \tau$.

Since the FT-UIO exists, there is

$$\frac{d}{dt}(z(t) - S\overline{x}_1(t)) = \dot{z}(t) - S\dot{\overline{x}}_1(t)$$

$$= Fz(t) + Gu(t) + Hy_p(t) - S(\overline{A}_{11}\overline{x}_1(t) + \overline{B}_1 u(t) + \widetilde{E} y_p(t))$$

$$= F(z(t) - S\overline{x}_1(t)) + FS\overline{x}_1(t) + (H - S\widetilde{E})y_p(t)$$

$$+ (G - S\overline{B}_1)u(t) - S\overline{A}_{11}\overline{x}_1(t) \tag{16}$$

Note that

$$(H - S\widetilde{E})y_p(t) = \left(\begin{bmatrix} L_1 U_3 + \widetilde{E} \\ L_2 U_3 + \widetilde{E} \end{bmatrix} - \begin{bmatrix} \widetilde{E} \\ \widetilde{E} \end{bmatrix} \right) y_p(t)$$

$$= \begin{bmatrix} L_1 \\ L_2 \end{bmatrix} U_3 y_p(t)$$

$$= \begin{bmatrix} L_1 \\ L_2 \end{bmatrix} \overline{y}_p(t)$$

$$= \begin{bmatrix} L_1 \\ L_2 \end{bmatrix} \widetilde{C}_p \overline{x}_1(t)$$

$$G - S\overline{B}_1 = \begin{bmatrix} \overline{B}_1 \\ \overline{B}_1 \end{bmatrix} - \begin{bmatrix} I_{Nn-l-q} \\ I_{Nn-l-q} \end{bmatrix} \overline{B}_1$$

$$= 0$$

we can substitute the above equations to (16), then

$$\frac{d}{dt}(z(t) - S\overline{x}_1(t)) = F(z(t) - S\overline{x}_1(t)) + \left(\begin{bmatrix} \overline{A}_{11} - L_1\widetilde{C}_p & 0 \\ 0 & \overline{A}_{11} - L_2\widetilde{C}_p \end{bmatrix} \right.$$

$$\left. - \begin{bmatrix} \overline{A}_{11} \\ \overline{A}_{11} \end{bmatrix} + \begin{bmatrix} L_1\widetilde{C}_p \\ L_2\widetilde{C}_p \end{bmatrix} \right) \overline{x}_1(t)$$

$$= F(z(t) - S\overline{x}_1(t)). \tag{17}$$

On the basis of this equation, the boundedness and convergence can be proved.
Boundedness Multiplied by e^{-Ft} and then integrating from t_0 to t, with $t \in [t_0, t_0+\tau]$, on both sides of (17), it can be easily shown that

$$\Rightarrow e^{-Ft}\left[z(t) - S\overline{x}_1(t)\right] - e^{-F(t_0)}\left[z(t_0) - S\overline{x}_1(t_0)\right] = 0 \tag{18}$$

$$\Rightarrow z(t) - S\overline{x}_1(t) = e^{-F(t_0-t)}\left[z(t_0) - S\overline{x}_1(t_0)\right]$$

$$= e^{F(t-t_0)}\left[z(t_0) - S\overline{x}_1(t_0)\right] \tag{19}$$

$$= e^{F(t-t_0)}S\left[\hat{\overline{x}}_1(t_0) - \overline{x}_1(t_0)\right]$$

With the assumption that $z(t) = S\hat{\overline{x}}_1(t_0)$, $\forall t \in [t_0 - \tau, t_0]$, it can be obtained that $z(t - \tau) = S\hat{\overline{x}}_1(t_0)$, $t \in [t_0, t_0 + \tau]$. Using the FT-UIO (15), when $t \in [t_0, t_0 + \tau]$, one can further obtain

$$\hat{\overline{x}}_1(t) = \left[Kz(t) - Ke^{F\tau}z(t - \tau)\right]$$

$$= KS\overline{x}_1(t) + Ke^{F(t-t_0)}S\left[\hat{\overline{x}}_1(t_0) - \overline{x}_1(t_0)\right] - Ke^{F\tau}z(t - \tau)$$

$$= KS\overline{x}_1(t) + Ke^{F(t-t_0)}S\left[\hat{\overline{x}}_1(t_0) - \overline{x}_1(t_0)\right] - Ke^{F\tau}S\hat{\overline{x}}_1(t_0) \tag{20}$$

Note that $KS = I_{Nn-l-q}$, and $Ke^{F\tau}S = 0$, (20) becomes

$$\hat{\overline{x}}_1(t) = \overline{x}_1(t) + Ke^{F(t-t_0)}S\left[\hat{\overline{x}}_1(t_0) - \overline{x}_1(t_0)\right] \tag{21}$$

Then the estimation error is

$$
\begin{aligned}
e(t) &= \hat{\bar{x}}_1(t) - \bar{x}_1(t) \\
&= K e^{F(t-t_0)} S \big[\hat{\bar{x}}_1(t_0) - \bar{x}_1(t_0) \big]
\end{aligned}
\tag{22}
$$

Owing to the assumption that $\hat{\bar{x}}_1(t_0)$ and $\bar{x}_1(t_0)$ are bounded, it's obvious that the estimation error shown in (22) stays bounded $\forall t \in [t_0, t_0 + \tau]$.

Finite-Time Convergence Multiplied by e^{-Ft} and then integrating from t_0 to $t - \tau$, with $t \geq t_0 + \tau$, on both sides of (17), we have:

$$
\Rightarrow e^{-F(t-\tau)} \big[z(t-\tau) - S\bar{x}_1(t-\tau) \big] - e^{-F(t_0)} \big[z(t_0) - S\bar{x}_1(t_0) \big] = 0
\tag{23}
$$

$$
\begin{aligned}
\Rightarrow z(t-\tau) - S\bar{x}_1(t-\tau) &= e^{-F(t_0 - (t-\tau))} \big[z(t_0) - S\bar{x}_1(t_0) \big] \\
&= e^{F(t-\tau-t_0)} \big[z(t_0) - S\bar{x}_1(t_0) \big] \\
&= e^{F(t-\tau-t_0)} S \big[\hat{\bar{x}}_1(t_0) - \bar{x}_1(t_0) \big]
\end{aligned}
\tag{24}
$$

Using the FT-UIO (15), equations (19) and (24), when $t \geq t_0 + \tau$, one can further obtain

$$
\begin{aligned}
\hat{\bar{x}}_1(t) &= \big[Kz(t) - Ke^{F\tau} z(t-\tau) \big] \\
&= K S\bar{x}_1(t) + Ke^{F(t-t_0)} S \big[\hat{\bar{x}}_1(t_0) - \bar{x}_1(t_0) \big] \\
&\quad - Ke^{F\tau} S\bar{x}_1(t-\tau) - Ke^{F\tau} e^{F(t-\tau-t_0)} S \big[\hat{\bar{x}}_1(t_0) - \bar{x}_1(t_0) \big] \\
&= K S\bar{x}_1(t) + Ke^{F(t-t_0)} S[\hat{\bar{x}}_1(t_0) - \bar{x}_1(t_0)] \\
&\quad - Ke^{F\tau} S\bar{x}_1(t-\tau) - Ke^{F(t-t_0)} S \big[\hat{\bar{x}}_1(t_0) - \bar{x}_1(t_0) \big] \\
&= K S\bar{x}_1(t) - Ke^{F\tau} S\bar{x}_1(t-\tau)
\end{aligned}
\tag{25}
$$

Note that $KS = I_{Nn-l-q}$, and $Ke^{F\tau} S = 0$, (25) becomes

$$
\hat{\bar{x}}_1(t) = \bar{x}_1(t), \ t \geq t_0 + \tau
\tag{26}
$$

Therefore, the FT-UIO (15) can estimate the state $\bar{x}_1(t)$ of the reduced-order system (12) in finite time τ.   □

Theorem 2 *For any $i, j \in (1 \sim Nn - l - q)$, if L_1 and L_2 are chosen such that*

(i) $Re(\lambda_i(F_1)) \neq Re(\lambda_j(F_2))$,
(ii) $Re(\lambda_i(F_1)) < 0$,
(iii) $Re(\lambda_i(F_2)) < 0$,

then the conditions that F is stable and $det[S, e^{F\tau} S] \neq 0$ in Theorem 1 can be matched for almost all $\tau \in \mathbb{R}^+$. This ensures the existence of the matrix K and thus the existence of the FT-UIO (15) for almost any finite time τ.

Proof Since $(\overline{A}_{11}, \widetilde{C}_p)$ is observable and

$$F = \begin{bmatrix} F_1 & 0 \\ 0 & F_2 \end{bmatrix} = \begin{bmatrix} \overline{A}_{11} - L_1\widetilde{C}_p & 0 \\ 0 & \overline{A}_{11} - L_2\widetilde{C}_p \end{bmatrix},$$

as long as $Re(\lambda_i(F_1)) < 0$ and $Re(\lambda_i(F_2)) < 0$, F can be stable.
While

$$\begin{bmatrix} S & e^{F\tau}S \end{bmatrix} = \begin{bmatrix} I_{Nn-l-q} & e^{F_1\tau} \\ I_{Nn-l-q} & e^{F_2\tau} \end{bmatrix}$$

and the delay $\tau > 0$, if $Re(\lambda_i(F_1)) \neq Re(\lambda_j(F_2))$ for any $i, j \in (1 \sim Nn - l - q)$, then the condition $det[S, e^{F\tau}S] \neq 0$ can also be achieved. $\qquad\square$

Remark 1 From the relations (9), (10) and (11), the vectors $\overline{x}_2(t)$ and $\overline{x}_3(t)$ can be represented only by output measurements, respectively. Hence, using the system (6), the estimation of the state $x(t)$ in system (4) can be expressed as follows

$$\hat{x}(t) = T\hat{\overline{x}}(t) = T \begin{bmatrix} \hat{\overline{x}}_1(t) \\ \hat{\overline{x}}_2(t) \\ \hat{\overline{x}}_3(t) \end{bmatrix} = T \begin{bmatrix} \hat{\overline{x}}_1(t) \\ U_1 y_p(t) \\ U_2 y_p(t) \end{bmatrix} \qquad (27)$$

Therefore, through the FT-UIO, the exact estimation of the state $x(t)$ in system (4) with disturbance and actuator fault can be obtained in a pre-assigned finite time. Once the estimation of $x(t)$ is obtained, the estimation of $f_i(t)$ in system (4) can be found immediately. Similarly with the system (4), we have

$$\begin{bmatrix} 0 & 0 & I_q \end{bmatrix} \dot{\overline{x}}(t) = \dot{\overline{x}}_3(t) = \begin{bmatrix} \overline{A}_{31} & \overline{A}_{32} & \overline{A}_{33} \end{bmatrix} \overline{x}(t) + \overline{B}_3 u(t) + f_r(t) \qquad (28)$$

Considering the system (28) and equation (27), the estimation $\hat{f}_r(t)$ of $f_r(t)$ is given as

$$\begin{aligned} \hat{f}_r(t) &= \dot{\hat{\overline{x}}}_3(t) - \begin{bmatrix} \overline{A}_{31} & \overline{A}_{32} & \overline{A}_{33} \end{bmatrix} \hat{\overline{x}}(t) - \overline{B}_3 u(t) \\ &= U_2 \dot{y}_p(t) - \overline{A}_{31}\hat{\overline{x}}_1(t) - (\overline{A}_{32}U_1 + \overline{A}_{33}U_2)y_p(t) - \overline{B}_3 u(t) \end{aligned} \qquad (29)$$

Remark 2 The FD algorithm above contains the derivative of $y_p(t)$, and it is feasible as long as $\dot{y}_p(t)$ can be obtained. However, in case that it cannot be easily obtained for certain systems, $\dot{y}_{pf}(t)$ is proposed instead of $\dot{y}_p(t)$ [18] as

$$\dot{y}_{pf}(t) = -\frac{1}{\alpha}(y_{pf}(t) - y_p(t)) \qquad (30)$$

By this method, (29) can be rewritten as

$$\hat{f}_r(t) = U_2 \dot{y}_{pf}(t) - \overline{A}_{31}\hat{\overline{x}}_1(t) - (\overline{A}_{32}U_1 + \overline{A}_{33}U_2)y_p(t) - \overline{B}_3 u(t) \qquad (31)$$

Remark 3 In this paper, a FT-UIO is constructed in a special agent p to monitor its neighbor agents r ($r \in N_p$). And it is assumed that $f_r(t)$ is the fault occurred in agent r. In fact, the $f_r(t)$ can be all of the faults occurred in the neighbor agents around agent p so that the FT-UIO constructed in agent p can diagnose the faults occurred in its neighbor agents simultaneously.

4 Simulation Results

To show the effectiveness of the method, we consider a group of linearized dynamic models in the vertical plane of a VTOL aircraft as an illustrative example, which can be described as

$$\begin{cases} \dot{x}_i(t) = Ax_i(t) + Bu_i(t) + Ef_i(t) + Dd_i(t) \\ y_i(t) = Cx_i(t) \end{cases} \tag{32}$$

where

$$x_i(t) = \begin{bmatrix} \text{horizontal velocity (kt)} \\ \text{vertical velocity (kt)} \\ \text{pitch rate (deg/s)} \\ \text{pitch angle (deg)} \end{bmatrix}$$

$$u_i(t) = \begin{bmatrix} \text{collective pitch control} \\ \text{longitudinal cyclic pitch angle} \end{bmatrix}$$

$$A = \begin{bmatrix} -0.0366 & 0.0271 & 0.0188 & -0.4555 \\ 0.0482 & -1.0100 & 0.0024 & -4.0208 \\ 0.1002 & 0.3681 & -0.7070 & 1.4200 \\ 0.0000 & 0.0000 & 1.0000 & 0.0000 \end{bmatrix},$$

$$B = \begin{bmatrix} 0.4422 & 0.1761 \\ 3.5446 & -7.5922 \\ -5.5200 & 4.4900 \\ 0.0000 & 0.0000 \end{bmatrix},$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad D = [0\ 0\ 1\ 0]^{\text{T}}.$$

It is assumed that the band-limited white noise is added as the unknown disturbance $d_i(t)$ and the actuator fault which usually occurs in the input channel is also taken into consideration with assumption $E = B$. The structure of the MAS is described by Fig. 1, with the associated adjacency matrix $\Lambda = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$.

Fig. 1 Topology structure of the multi-agent system
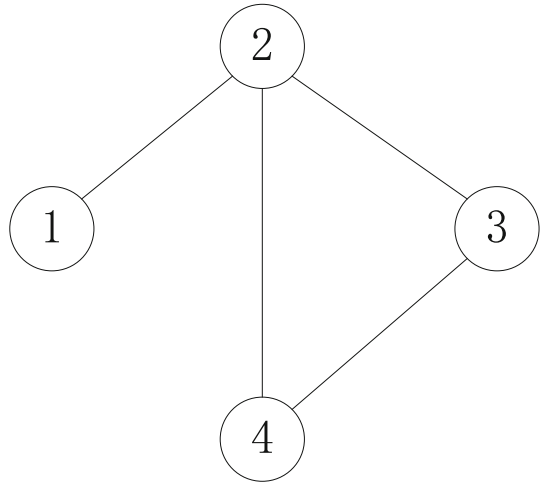
In this study, a FT-UIO is constructed at agent 2 to diagnose the faults occurred in its neighbor agents, which are agents 1, 3 and 4. The press-assigned finite time designed as the delay in FT-UIO is chosen as $\tau = 1$, with $\alpha = 0.001$. According to Lemma 4, using MATLAB, we can obtain these important parameters by placing the poles at $[-2 \ -1 \ -2 \ -2]$ and $[-7 \ -6 \ -7 \ -7]$, respectively.

$$
I_N \otimes E_{-r} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0.4422 & 3.5446 & -5.5200 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.1761 & -7.5922 & 4.4900 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^{\mathrm{T}}
$$

$$
I_N \otimes E_r = \begin{bmatrix}
0.4422 & 0.1761 & 0 & 0 & 0 & 0 \\
3.5446 & -7.5922 & 0 & 0 & 0 & 0 \\
-5.5200 & 4.4900 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.4422 & 0.1761 & 0 & 0 \\
0 & 0 & 3.5446 & -7.5922 & 0 & 0 \\
0 & 0 & -5.5200 & 4.4900 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.4422 & 0.1761 \\
0 & 0 & 0 & 0 & 3.5446 & -7.5922 \\
0 & 0 & 0 & 0 & -5.5200 & 4.4900 \\
0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

$$L_1 = \begin{bmatrix} 3.4404 & 1.7202 & -1.7202 & -1.7202 \\ 0.7202 & 0 & 0 & 0 \\ -1.7202 & 0 & 0 & 1.7202 \\ -1.7202 & 0 & 1.7202 & 0 \end{bmatrix}$$

$$L_2 = \begin{bmatrix} 13.4404 & 6.7202 & -6.7202 & -6.7202 \\ 5.7202 & 0 & 0 & 0 \\ -6.7202 & 0 & 0 & 6.7202 \\ -6.7202 & 0 & 6.7202 & 0 \end{bmatrix}$$

$$K = \begin{bmatrix} -0.0068 & 0 & 0 & 0 & 1.0068 & 0 & 0 & 0 \\ 0 & -0.0068 & 0 & 0 & 0 & 1.0068 & 0 & 0 \\ 0 & 0 & -0.0068 & 0 & 0 & 0 & 1.0068 & 0 \\ 0 & 0 & 0 & -0.0068 & 0 & 0 & 0 & 1.0068 \end{bmatrix}$$

During the process of coordinate transformation, N_0 and Q_0 are chosen as follows and then some important parameters are also obtained.

$$Q_0 = N_0 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T$$

$$T = \begin{bmatrix} 0.4892 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0.0557 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0.0749 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.5448 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0.4892 & 0 & 0 & 0.4422 & 0.1761 & 0 & 1 \\ 0 & 0.0557 & 0 & 0 & 3.5446 & -7.5922 & 0 & 1 \\ 0 & 0.0749 & 0 & 0 & -5.5200 & 4.4900 & 0 & 0 \\ 0 & 0.5448 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0.4892 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0.0557 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0.0749 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5448 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0.4892 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.0557 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.0749 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.5448 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$
\begin{bmatrix}
0 & 0 & 0.4422 & 0.1761 & 0 & 0 & 0 & 0 \\
0 & 0 & 3.5446 & -7.5922 & 0 & 0 & 0 & 0 \\
0 & 0 & -5.5200 & 4.4900 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.4422 & 0.1761 & 0 & 0 & 0 \\
0 & 0 & 0 & 3.5446 & -7.5922 & 0 & 0 & 0 \\
0 & 0 & 0 & -5.5200 & 4.4900 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0.4422 & 0.1761 \\
0 & 1 & 0 & 0 & 0 & 0 & 3.5446 & -7.5922 \\
0 & 0 & 0 & 0 & 0 & 0 & -5.5200 & 4.4900 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

$$
U =
\begin{bmatrix}
0.4422 & 0.1761 & 0 & 1 & 0 & 0 & 0 & 0 \\
3.5446 & -7.5922 & 0 & 1 & 0 & 0 & 0 & 0 \\
-5.5200 & 4.4900 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 1 & 0.4422 & 0.1761 \\
0 & 0 & 1 & 0 & 1 & 1 & 3.5446 & -7.5922 \\
0 & 0 & 0 & 0 & 0 & 0 & -5.5200 & 4.4900 \\
0 & 0 & -1 & 0 & -1 & -1 & 0 & 0 \\
0.4422 & 0.1761 & 0 & 1 & 0 & 1 & 0 & 0 \\
3.5446 & -7.5922 & 0 & 1 & 0 & 1 & 0 & 0 \\
-5.5200 & 4.4900 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 0 & -1 & 0 & 0 \\
0.4422 & 0.1761 & 0 & 1 & 1 & 0 & 0 & 0 \\
3.5446 & -7.5922 & 0 & 1 & 1 & 0 & 0 & 0 \\
-5.5200 & 4.4900 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & -1 & 0 & 0 & 0
\end{bmatrix}
$$

$$
\begin{bmatrix}
0 & 0 & 0 & 0 & 0.4892 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.0557 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.0749 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.5448 & 0 & 0 & 0 \\
0.4422 & 0.1761 & 0.4422 & 0.1761 & 0 & 0.4892 & 0 & 0 \\
3.5446 & -7.5922 & 3.5446 & -7.5922 & 0 & 0.0557 & 0 & 0 \\
-5.5200 & 4.4900 & -5.5200 & 4.4900 & 0 & 0.0749 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0.5448 & 0 & 0 \\
0 & 0 & 0.4422 & 0.1761 & 0 & 0 & 0.4892 & 0 \\
0 & 0 & 3.5446 & -7.5922 & 0 & 0 & 0.0557 & 0 \\
0 & 0 & -5.5200 & 4.4900 & 0 & 0 & 0.0749 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0.5448 & 0 \\
0.4422 & 0.1761 & 0 & 0 & 0 & 0 & 0 & 0.4892 \\
3.5446 & -7.5922 & 0 & 0 & 0 & 0 & 0 & 0.0557 \\
-5.5200 & 4.4900 & 0 & 0 & 0 & 0 & 0 & 0.0749 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5448
\end{bmatrix}
$$

To validate the effect of the FD methods, after constructing a FT-UIO in agent 2, the following two simulation cases are used.

Case I Assume that agents 1 , 2 and 4 all occur faults and there is no fault in agent 3. It should be noted that the agent 2 is the agent constructed the FT-UIO and its fault should theoretically not affect the FD of its neighbor agents.

For the agent 1: $f_1(t) = \begin{bmatrix} f_{11}(t) \\ f_{12}(t) \end{bmatrix}$:

$$
f_{11}(t) = \begin{cases} 0 & 0s \leq t < 2s \\ 0.05e^{-0.3(t-4)} & 2 \leq t < 7s \end{cases}, \quad f_{12}(t) = 0
$$

For the agent 3: $f_2(t) = \begin{bmatrix} f_{21}(t) \\ f_{22}(t) \end{bmatrix}$:

$$
f_{21}(t) = \begin{cases} 0 & 0s \leq t < 1s \\ 0.5\sin(2(t-4)) & 1s \leq t < 7s \end{cases}, \quad f_{22}(t) = 0
$$

For the agent 4: $f_4(t) = \begin{bmatrix} f_{41}(t) \\ f_{42}(t) \end{bmatrix}$:

$$
f_{41}(t) = \begin{cases} 0 & 0s \leq t < 3s \\ 0.6\sin(-1.5(t-4)) & 3s \leq t < 7s \end{cases}, \quad f_{42}(t) = 0
$$

The obtained simulation results are shown as follows (Fig. 2):

Case II Assume that faults are occurred in agents 2, 3 and 4. In accordance with the above, the agent 2 is the agent constructed the FT-UIO and its fault should theoretically
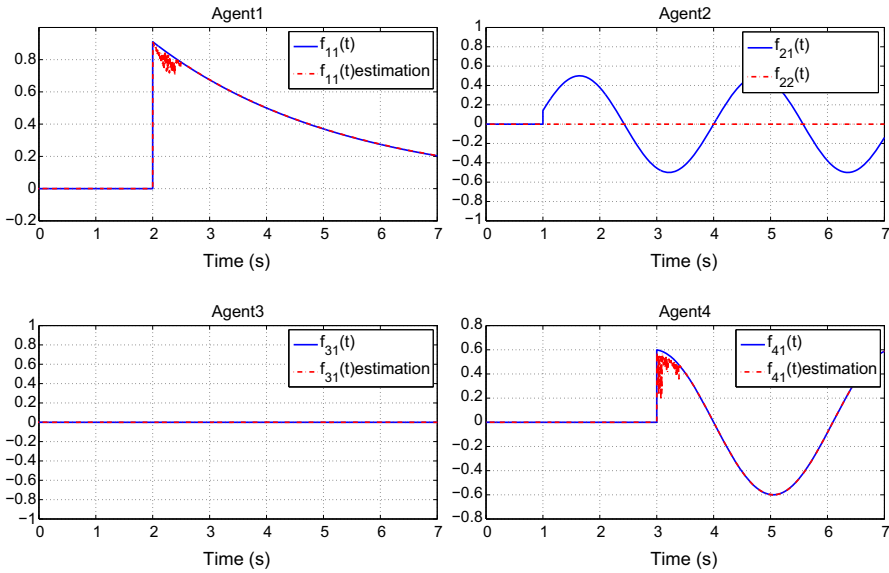
Fig. 2 Simulation results of case I

not affect the FD of its neighbor agents. And we assume that the fault occurred in agent 2 remains the same.

For the agent 3: $f_3(t) = \begin{bmatrix} f_{31}(t) \\ f_{32}(t) \end{bmatrix}$:

$$f_{31}(t) = \begin{cases} 0 & 0s \leq t < 3s \\ -0.7e^{0.2(t-5)} & 3s \leq t < 7s \end{cases}, \quad f_{32}(t) = 0$$

For the agent 4: $f_4(t) = \begin{bmatrix} f_{41}(t) \\ f_{42}(t) \end{bmatrix}$:

$$f_{41}(t) = \begin{cases} 0 & 0s \leq t < 3s \\ 0.65\sin(t-4) & 3s \leq t < 7s \end{cases}, \quad f_{42}(t) = 0$$

The obtained simulation results are shown as follows (Fig. 3):

As shown in these figures, when the faults occur in the neighbor agents, the FT-UIO in agent 2 can diagnose the faults in finite time $\tau = 1s$. Furthermore, the FD of each neighbor agent will not be affected by the faults occurred in other agents including the agent 2, which is selected to construct the FT-UIO. Simulation results show the effectiveness of the proposed design technique.

5 Conclusion

In this paper, a FT-UIO-based distributed FD scheme for MAS with actuator faults and disturbances is proposed. Under the communication topology , the global multi-agent
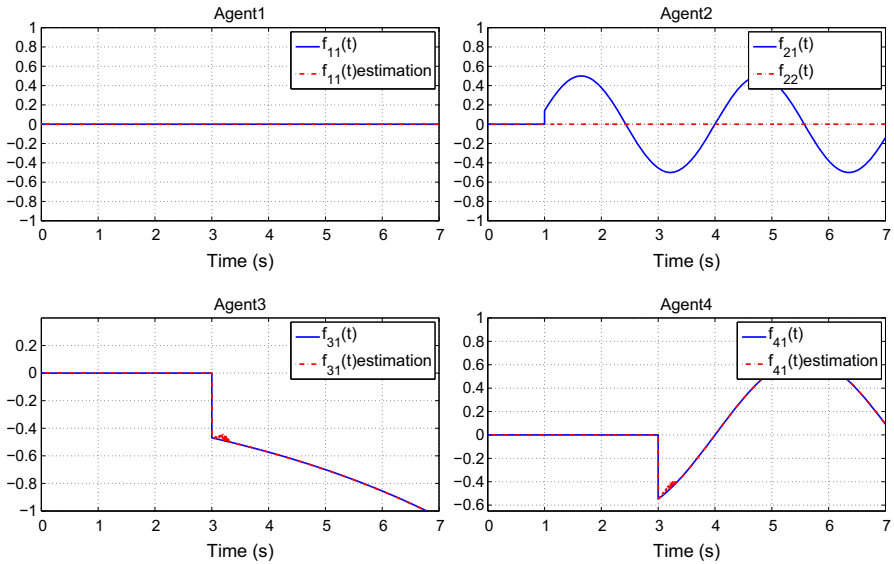
Fig. 3 Simulation results of case II

system dynamics are represented with respect to a certain agent and the output of the system becomes the local information which can be used by the selected agent. Then, the system will be decomposed into three subsystems. Next, constituted by two distinct observers, the FT-UIO is constructed at the selected agent and is designed to converge in a pre-assigned finite time, which is chosen as the time delay of the observers. With the state estimation obtained from the FT-UIO in the selected agent, a FD algorithm is also proposed to diagnose the faults occurred in its neighbor agents, and then not only the location of the faults but also the detailed information of the faults can be obtained in a pre-assigned finite time. Finally, the effectiveness and advantages of the proposed distributed FD method are shown through simulation studies performed on the control of a VTOL aircraft in the vertical plane.

References

1. J. Chen, R.J. Patton, *Robust Model-Based Fault Diagnosis for Dynamic Systems* (Kluwer Academic Publishers, Norwell, 1999)
2. S.X. Ding, *Model-Based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools* (Springer, Berlin, 2013)
3. P. Dorato, Short time stability in linear time-varying systems. PTOCI. RE Int. Conv. **4**, 83–87 (1961)
4. Z.H. Guan, F.L. Sun, Y.W. Wang et al., Finite-Time consensus for leader-following second-order multi-agent networks. IEEE Trans. Circuits Syst. I Regul. Pap. **59**(11), 2646–2654 (2012)
5. Y. Guo, S.M. Song, Finite-time control for formation flying spacecraft with coupled attitude and translational dynamics. IEEE Int. Conf. Control Autom. **45**(5), 89–94 (2013)
6. B. Jiang, K. Zhang, P. Shi, Integrated fault estimation and accommodation design for discrete-time Takagi–Sugeno fuzzy systems with actuator faults. IEEE Trans. Fuzzy Syst. **19**(2), 291–304 (2011)
7. K.S. Lee, T.G. Park, New results on fault reconstruction using a finite-time converging unknown input observer. IET Control Theory Appl. **6**(9), 1258–1265 (2012)

8. F.L. Lewis, H. Zhang, K. Hengster-Movric, A. Das, *Cooperative Control of Multi-agent Systems: Optimal and Adaptive Design Approaches* (Springer, New York, 2014), pp. 822–831

9. T. Liu, B. Wu, L. Liu et al., Finite-time stability of discrete switched singular positive systems. Circuits Syst. Signal Process. **36**(6), 2243–2255 (2017)

10. X. Lu, R. Lu, S. Chen et al., Finite-time distributed tracking control for multi-agent systems with a virtual leader. IEEE Trans. Circuits Syst. I Regul. Pap. **60**(2), 352–362 (2013)

11. B. Marx, D. Koenig, J. Ragot, Design of observers for Takagi–Sugeno descriptor systems with unknown inputs and application to fault diagnosis. IET Control Theory Appl. **1**(5), 1487–1495 (2007)

12. E. Moulay, W. Perruquetti, Finite time stability conditions for non-autonomous continuous systems. Int. J. Control **81**(5), 797–803 (2008)

13. K.K. Oh, M.C. Park, H.S. Ahn, A survey of multi-agent formation control. Automatica **53**, 424–440 (2015)

14. R. Olfati-Saber, Flocking for multi-agent dynamic systems: algorithms and theory. IEEE Trans. Autom. Control **51**(3), 401–420 (2006)

15. L. Qin, X. He, D.H. Zhou, A survey of fault diagnosis for swarm systems. Syst. Sci. Control Eng. Open Access J. **2**(1), 13–23 (2014)

16. X. Qin, H. Fang, X. Liu, Strong tracking filter-based fault diagnosis of networked control system with multiple packet dropouts and parameter perturbations. Circuits Syst. Signal Process. **35**(7), 1–20 (2016)

17. K. Reif et al., Networked control systems in motor vehicles. ATZelektronik Worldw. **3**(4), 18–23 (2008)

18. W. Ren, R.W. Beard, E.M. Atkins, Information consensus in multivehicle cooperative control. IEEE Control Syst. **129**(2), 571–583 (2007)

19. T. Schetter, M. Campbell, D. Surka, Multiple agent-based autonomy for satellite constellations. Artif. Intell. **145**(1–2), 147–80 (2000)

20. J.Y. Shi, L. Chen, W.W. Cui, A new hybrid fault diagnostic method for combining dependency matrix diagnosis and fuzzy diagnosis based on an enhanced inference operator. Circuits Syst. Signal Process. **35**(1), 1–28 (2016)

21. A. Valdes, K. Khorasani, L. Ma. Dynamic neural network-based fault detection and isolation for thrusters in formation flying of satellites, in *International Symposium on Neural Networks: Advances in Neural Networks*, (Springer, Berlin, 2009), pp. 780–793

22. X. Wang, H. Su, X. Wang et al., Second-order consensus of multi-agent systems via periodically intermittent pinning control. Circuits Syst. Signal Process. **35**(7), 2413–2431 (2016)

23. L. Weiss, E.F. Infante, Finite time stability under perturbing forces and on product spaces. IEEE Trans. Autom. Control **12**(1), 54–59 (1967)

24. X. Zhang, Q. Zhang et al., Distributed fault diagnosis in a class of interconnected nonlinear uncertain systems. Int. J. Control **37**(11), 170–179 (2012)

25. K. Zhang, B. Jiang, V. Cocquempot, Adaptive technique-based distributed fault estimation observer design for multi-agent systems with directed graphs. IET Control Theory Appl. **9**(18), 2619–2625 (2015)

26. K. Zhang, B. Jiang, P. Shi, V. Cocquempot, *Observer-Based Fault Estimation Techniques* (Springer, Berlin, 2017)

27. Y. Zhao, Z. Duan, G. Wen, Finite-time consensus for second-order multi-agent systems with saturated control protocols. IET Control Theory Appl. **9**(3), 312–319 (2015)

28. M. Zhou, Y. Shen, Q. Wang. Robust UIO-based fault estimation for sampled-data systems: an LMI approach. In *IEEE International Conference on Information and Automation*, (2013), pp. 1308–1313