Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Containment control for heterogeneous nonlinear multi-agent systems under distributed event-triggered schemes*

Ya-ni SUN, Wen-cheng ZOU, Jian GUO, Zheng-rong XIANG†‡

School of Automation, Nanjing University of Science and Technology, Nanjing 210094, China

†E-mail: xiangzr@njust.edu.cn

Received Jan. 18, 2020; Revision accepted Aug. 5, 2020; Crosschecked Sept. 11, 2020; Published online Oct. 29, 2020

Abstract: We study the containment control problem for high-order heterogeneous nonlinear multi-agent systems under distributed event-triggered schemes. To achieve the containment control objective and reduce communication consumption among agents, a distributed event-triggered control scheme is proposed by applying the backstepping method, Lyapunov functional approach, and neural networks. Then, the results are extended to the self-triggered control case to avoid continuous monitoring of state errors. The developed protocols and triggered rules ensure that the output for each follower converges to the convex hull spanned by multi-leader signals within a bounded error. In addition, no agent exhibits Zeno behavior. Two numerical simulations are finally presented to verify the correctness of the obtained results.

Key words: Multi-agent systems; Distributed event-triggered control; Containment control; Heterogeneous nonlinear systems; Zeno behavior

https://doi.org/10.1631/FITEE.2000034 **CLC number:** TP13

1 Introduction

To date, a great deal of work has been put into cooperative control for multi-agent systems (MASs) based on the behaviors of animal groups in nature and their wide practical applications such as consensus control (Wang ZD et al., 2013; Li HL et al., 2017; Lin and Zheng, 2017; Rehan et al., 2018; Zheng YS et al., 2018), rendezvous control (Fan et al., 2011), formation control (Ma CQ and Zhang, 2012; Li WX et al., 2014; Liu YF and Geng, 2015), and containment control (Liu KE et al., 2014; Ma Q and Miao, 2014; Qin et al., 2017). In the study of cooperative control for MASs, consensus is one of the most fundamental problems and has been extensively investigated in the past few decades. Consensus means that by developing appropriate protocols, the states of all agents in the system converge to a common value. Generally speaking, MASs can be divided into two categories: leaderless and leader-following MASs. For an MAS with only one leader, the consensus problem is called leader-following consensus (Guo, 2016; Qin et al., 2016; Lu and Liu, 2019). In some applications, there might be multiple leaders in an MAS; the problem in this case is called containment control, in which all the followers are expected to converge to the convex hull spanned by multiple leaders under appropriate control laws (Ji et al., 2008).

In recent years, containment control has attracted enormous concern because of its pratical applications (Xu and Zhu, 2011). Liu KE et al.

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (Nos. 61873128 and 61673219) and the Jiangsu Key Research and Development Plan, China (No. BE2018004-3)

ORCID: Ya-ni SUN, https://orcid.org/0000-0003-3670-6673; Zheng-rong XIANG, https://orcid.org/0000-0002-0869-5471

(2014) addressed the containment control problem for second-order linear MASs with time-varying delays, where both dynamic and stationary leaders were considered. Zhou and Wang (2015) studied the containment control for linear MASs, where the graph is directed and the control inputs of the leaders may be time-varying. Ma Q and Miao (2014) proposed distributed dynamic and static control methods for linear MASs with time-varying delays. Haghshenas et al. (2015) and Chu et al. (2016) proposed containment control schemes for heterogeneous linear systems. Fu and Wang (2016) solved the finite-time containment control problem by designing a distributed observer-based controller. Obviously, there have been many achievements in containment control of linear MASs. However, it is universally known that most real systems are nonlinear, and thus it is absolutely essential to study nonlinear MASs (Yoo, 2013; He et al., 2015; Liu ZX et al., 2015; Wang P and Jia, 2015; Wang W et al., 2015; Li WQ et al., 2016). In the cooperative control of nonlinear MASs, the major challenge lies in how to deal with the nonlinearities. He et al. (2015) and Li YF et al. (2017) designed nonlinear functions to satisfy some growth conditions such as the Lipschitz condition. Wang W et al. (2015) applied a fuzzy logic system to approximate the nonlinearities.

In the view of continuous communication among agents which may cause excessive data and waste embedded processor resources, the event-triggered method is more favorable when designing protocols (Dimarogonas et al., 2012; Yan et al., 2014; Hu et al., 2015; Wei and Xiao, 2016; Dolk and Heemels, 2017; Wu et al., 2018a, 2018b; Zou and Xiang, 2019a). Tabuada (2007) proposed an event-triggered scheme, which was applied into MASs for both centralized and distributed cases in Dimarogonas et al. (2012). There have been some studies on distributed event-triggered containment control of MASs with multiple leaders. Miao et al. (2017) studied containment control for first- and second-order systems with single time delay and multiple time delays in event-triggered schemes, respectively. Zou and Xiang (2019b) designed event-triggered control strategies with the improved triggered condition for second-order MASs. The problem of event-triggered containment control for second-order MASs with sampled position data was addressed in Xia et al. (2018). Zou and Xiang (2017) presented an event-triggered control method for heterogeneous linear MASs by an output regulation approach. Zhang WB et al. (2017) addressed the containment control problem using event-triggered strategies for MASs, where communication may exist among the leaders and the Zeno behavior is excluded. Li JZ et al. (2018) designed event-triggered protocols for high-order MASs with and without input delays.

Note that there have been few studies involving event-triggered containment control for high-order heterogeneous nonlinear MASs. In consideration of this, we focus on high-order heterogeneous nonlinear MASs. The main contributions of this study are summarized as follows:

1. By applying the backstepping method, Lyapunov stability approach, and neural network technique, both distributed event-triggered and self-triggered containment control schemes are proposed. Compared with the MASs in Miao et al. (2017), Zhang WB et al. (2017), Zou and Xiang (2017), Li JZ et al. (2018), and Xia et al. (2018), the system in this study is more general. In addition, in most literature on nonlinear MASs, like Li YF et al. (2017) and Zou and Xiang (2019b), nonlinear functions are usually required to satisfy specific conditions such as the global Lipschitz condition. However, the restrictions of the nonlinear terms in this study are relaxed and the nonlinearities are totally unknown and heterogeneous. The control schemes in this study can achieve the containment control objective in the presence of more general nonlinearities.

2. Due to the existence of unknown and heterogeneous nonlinear terms, the lower bound of the interval between two adjacent triggers may not exist, which makes many existing control methods invalid as the agents may exhibit Zeno behaviors. Under the event-triggered and self-triggered schemes proposed in this study, Zeno behavior can be excluded strictly despite the MAS complexity.

Notations used in this paper are summarized as follows: $\boldsymbol{P} > \boldsymbol{0}$ (or $\boldsymbol{P} \geq \boldsymbol{0}$) indicates that matrix $\boldsymbol{P} \in \mathbb{R}^{n\times n}$ is positive definite (or positive semi-definite). The maximum and minimum eigenvalues of $\boldsymbol{P}$ are denoted by $\lambda_{\max}(\boldsymbol{P})$ and $\lambda_{\min}(\boldsymbol{P})$, respectively. "$\otimes$" represents the Kronecker product. The convex hull of X is denoted as $\mathrm{Co}\,(X)$, where $X = \{x_1, x_2, \ldots, x_n\}$. $\mathrm{dis}\,(\boldsymbol{x}, \mathcal{Z})$ represents the distance from $\boldsymbol{x} \in \mathbb{R}^n$ to the set $\mathcal{Z} \subseteq \mathbb{R}^n$, and $\mathrm{dis}\,(\boldsymbol{x}, \mathcal{Z}) = \inf\limits_{\boldsymbol{z}\in\mathcal{Z}} \|\boldsymbol{x} - \boldsymbol{z}\|$, where $\|\cdot\|$ means the Euclidean norm of a vector or a matrix.

2 Preliminaries and problem formulation

2.1 Graph theory

Suppose that graph $G=(V,E)$ with vertex set $V=\{1,2,\ldots,N+M\}$ and edge set $E\subseteq V\times V$ stands for the communication topology of MASs consisting of M leaders and N followers. $\boldsymbol{A}=[a_{ij}]_{(N+M)\times(N+M)}$ is the adjacency matrix, where each element is nonnegative. In detail, $(j,i)\in E\Leftrightarrow a_{ij}>0$, and $(j,i)\notin E\Leftrightarrow a_{ij}=0$. $(i,j)\in E$ means that agent j can receive information from agent i, and the neighbor set of agent i is $\mathcal{N}_i=\{j:j\in V,(j,i)\in E\}$. $\boldsymbol{D}=\mathrm{diag}(d_1,d_2,\ldots,d_{N+M})$ implies the degree matrix of G, where $d_i=\sum\limits_{j\in\mathcal{N}_i}a_{ij}$ is the degree of agent i. The Laplacian matrix can be denoted as

$$\boldsymbol{L}=\boldsymbol{D}-\boldsymbol{A}=\begin{pmatrix}\boldsymbol{L}_1 & \boldsymbol{L}_2\\ \boldsymbol{0}_{M\times N} & \boldsymbol{0}_{M\times M}\end{pmatrix},$$

where $\boldsymbol{L}_1\in\mathbb{R}^{N\times N}$ and $\boldsymbol{L}_2\in\mathbb{R}^{N\times M}$.

2.2 Radial basis function neural networks

The following radial basis function neural networks (RBFNNs) are applied to approximate an unknown continuous function (Zhang T et al., 2000)

$$\Phi_{nn}(\boldsymbol{y})=\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{H}(\boldsymbol{y}),\tag{1}$$

where $\boldsymbol{Z}=(z_1,z_2,\ldots,z_l)^{\mathrm{T}}\in\mathbb{R}^l$ refers to the weight vector and $\boldsymbol{H}(\boldsymbol{y})=[h_1(\boldsymbol{y}),h_2(\boldsymbol{y}),\ldots,h_l(\boldsymbol{y})]^{\mathrm{T}}\in\mathbb{R}^l$ is the basis function vector ($l>1$ represents the number of nodes in the neural networks).

Denoting μ_i as the width of $h_i(\boldsymbol{y})$ and $\boldsymbol{w}_i$ as the center, select $h_i(\boldsymbol{y})$ as the Gaussian function:

$$h_i(\boldsymbol{y})=\exp\left[-\frac{(\boldsymbol{y}-\boldsymbol{w}_i)^{\mathrm{T}}(\boldsymbol{y}-\boldsymbol{w}_i)}{\mu_i^2}\right],\ i=1,2,\ldots,l.\tag{2}$$

Any unknown function $\phi(\boldsymbol{y})$ that is continuous, nonlinear, and defined on a compact set $\Omega\in\mathbb{R}^q$ can be approximated by a neural network $(\boldsymbol{Z}^*)^{\mathrm{T}}\boldsymbol{H}(\boldsymbol{y})$; that is, for any arbitrary accuracy $\delta_0>0$, one has

$$\phi(\boldsymbol{y})=(\boldsymbol{Z}^*)^{\mathrm{T}}\boldsymbol{H}(\boldsymbol{y})+\delta(\boldsymbol{y}),\ \boldsymbol{y}\in\Omega,\tag{3}$$

where $\boldsymbol{Z}^*\triangleq\arg\limits_{\boldsymbol{Z}\in\mathbb{R}^l}\left\{\sup\limits_{\boldsymbol{y}\in\Omega}\left|\phi(\boldsymbol{y})-\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{H}(\boldsymbol{y})\right|\right\}$ means the appropriate constant weight vector, and $|\delta(\boldsymbol{y})|<\delta_0$.

2.3 Problem formulation

Considering an n^{th}-order nonlinear MAS which consists of M leaders and N followers, the i^{th} follower is modeled by

$$\begin{cases}\dot{x}_{i,1}=x_{i,2}+f_{i,1}(x_{i,1}),\\ \dot{x}_{i,2}=x_{i,3}+f_{i,2}(x_{i,1},x_{i,2}),\\ \qquad\vdots\\ \dot{x}_{i,n-1}=x_{i,n}+f_{i,n-1}(x_{i,1},x_{i,2},\ldots,x_{i,n-1}),\\ \dot{x}_{i,n}=u_i+f_{i,n}(x_{i,1},x_{i,2},\ldots,x_{i,n}),\\ y_i=x_{i,1},\ i\in\Gamma_1,\end{cases}\tag{4}$$

where $\Gamma_1=\{1,2,\ldots,N\}$, $(x_{i,1},x_{i,2},\ldots,x_{i,n})^{\mathrm{T}}\in\mathbb{R}^n$ is the state of agent i, $u_i\in\mathbb{R}$ and $y_i\in\mathbb{R}$ are the control input and output of the i^{th} follower, respectively, and $f_{i,p}(\cdot)\in\mathbb{R}$ $(p=1,2,\ldots,n)$ is the unknown nonlinear function.

Denote w_k as the k^{th} leader signal, $k\in\Gamma_2=\{N+1,N+2,\ldots,N+M\}$.

Remark 1 If $M=1$, the containment control problem for MAS (4) will degrade into the leader-following consensus. Compared with the MASs in the existing studies on event-triggered containment control, MAS (4) is more general. In addition, the dynamics of the followers is heterogeneous and of higher order, and the nonlinearities are not required to satisfy specific conditions, such as the global Lipschitz condition, challenging the achievement of event-triggered containment control, and the exclusion of Zeno behavior.

Assumption 1 The leader signal w_k $(k\in\Gamma_2)$ is an n-times continuously differentiable function. Besides, there exists $R_0>0$ such that $\left|w_k^{(\bar{p})}\right|\leq R_0$, $\bar{p}=0,1,\ldots,n$.

Assumption 2 The communication graph among all the followers is undirected and none of them could send information to leaders. What is more, for each follower, there is at least one leader that has a directed path to it.

Lemma 1 (Meng et al., 2010) Suppose that Assumption 2 holds, and that $\boldsymbol{L}_1$ is positive definite and symmetric. Each entry of $-\boldsymbol{L}_1^{-1}\boldsymbol{L}_2$ is nonnegative and the element sum of each row equals 1.

Lemma 2 (Hardy et al., 1952) For any $u,v\in\mathbb{R}$,

$$|u|^{\nu}|v|^{\pi}\leq\frac{\nu}{\nu+\pi}\rho|u|^{\nu+\pi}+\frac{\pi}{\nu+\pi}\rho^{-\frac{\nu}{\pi}}|v|^{\nu+\pi},\tag{5}$$

where ν, π, and ρ are arbitrary positive real numbers.

Definition 1 (Ji et al., 2008) MAS (4) can achieve the containment control objective if the designed protocol makes the output of each follower converge to the convex hull spanned by leader signals, which means $\forall i \in \Gamma_1$, $\lim_{t\to\infty} \operatorname{dis}(y_i, \operatorname{Co}(w_{N+1}, w_{N+2}, \ldots, w_{N+M})) = 0$.

Let $[\tilde{\boldsymbol{w}}_1^{\mathrm{T}}, \tilde{\boldsymbol{w}}_2^{\mathrm{T}}, \ldots, \tilde{\boldsymbol{w}}_N^{\mathrm{T}}]^{\mathrm{T}} = (-\boldsymbol{L}_1^{-1}\boldsymbol{L}_2 \otimes \boldsymbol{I}_n) \cdot [\bar{\boldsymbol{w}}_{N+1}^{\mathrm{T}}, \bar{\boldsymbol{w}}_{N+2}^{\mathrm{T}}, \ldots, \bar{\boldsymbol{w}}_{N+M}^{\mathrm{T}}]^{\mathrm{T}}$, $\tilde{\boldsymbol{w}}_i = [\tilde{w}_{i,1}, \tilde{w}_{i,2}, \ldots, \tilde{w}_{i,n}]^{\mathrm{T}} \in \mathbb{R}^n$ $(i \in \Gamma_1)$, and $\bar{\boldsymbol{w}}_k = [w_k, w_k^{(1)}, \ldots, w_k^{(n-1)}]^{\mathrm{T}} \in \mathbb{R}^n$ $(k \in \Gamma_2)$.

In this study, the practical containment control objective can be achieved if the convergence state error $y_i - \tilde{w}_{i,1}$ satisfies $\lim_{t\to\infty} |y_i - \tilde{w}_{i,1}| \le \varepsilon_0, \forall i \in \Gamma_1$, where ε_0 is a positive constant.

3 Event-triggered containment control

3.1 Event-triggered protocol design

For simplification, we first define

$$\begin{cases} \rho_{i,1} = \sum_{j=1}^{N} a_{ij}\left(x_{j,1}^r - x_{i,1}^r\right) + \sum_{k=N+1}^{N+M} a_{ik}\left(w_k - x_{i,1}^r\right), \\ \rho_{i,2} = \sum_{j=1}^{N} a_{ij}\left(x_{j,2}^r - x_{i,2}^r\right) + \sum_{k=N+1}^{N+M} a_{ik}\left(w_k^{(1)} - x_{i,2}^r\right), \\ \quad\vdots \\ \rho_{i,n} = \sum_{j=1}^{N} a_{ij}\left(x_{j,n}^r - x_{i,n}^r\right) \\ \qquad + \sum_{k=N+1}^{N+M} a_{ik}\left(w_k^{(n-1)} - x_{i,n}^r\right), \ i \in \Gamma_1, \end{cases} \tag{6}$$

where $\boldsymbol{x}_i^r \triangleq \left[x_{i,1}^r, x_{i,2}^r, \ldots, x_{i,n}^r\right]^{\mathrm{T}} \in \mathbb{R}^n$ is called the objective route of agent i.

The dynamics of $\boldsymbol{x}_i^r$ is given as follows:

$$\dot{\boldsymbol{x}}_i^r = \boldsymbol{A}\boldsymbol{x}_i^r + \boldsymbol{B}\boldsymbol{K}\boldsymbol{\rho}_i\left(t_k^i\right), \quad t \in \left[t_k^i, t_{k+1}^i\right), \tag{7}$$

where $\boldsymbol{A} = \begin{bmatrix} \boldsymbol{0}_{(n-1)\times 1} & \boldsymbol{I}_{(n-1)\times(n-1)} \\ 0 & \boldsymbol{0}_{1\times(n-1)} \end{bmatrix} \in \mathbb{R}^{n\times n}$, $\boldsymbol{B} = \begin{bmatrix} \boldsymbol{0}_{1\times(n-1)} & 1 \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^n$, $\boldsymbol{K} = [K_1, K_2, \ldots, K_n] \in \mathbb{R}^{1\times n}$ ($\boldsymbol{K}$ needs to be designed later), $\boldsymbol{\rho}_i = [\rho_{i,1}, \rho_{i,2}, \ldots, \rho_{i,n}]^{\mathrm{T}} \in \mathbb{R}^n$, and t_k^i represents the triggered instant of the i^{th} follower. In distributed event-triggered control, each follower has its own triggered time.

Then, the triggered condition is given as

$$t_{k+1}^i = \inf\left\{t > t_k^i \,|\, \|\boldsymbol{\omega}_i\| \ge \Delta_i\right\}, \ \Delta_i > 0, \tag{8}$$

where $\boldsymbol{\omega}_i = \boldsymbol{\rho}_i(t) - \boldsymbol{\rho}_i\left(t_k^i\right)$, $i \in \Gamma_1$.

Let $\boldsymbol{e}_i = \boldsymbol{x}_i^r - \tilde{\boldsymbol{w}}_i$, $i \in \Gamma_1$. The Lyapunov function is chosen as

$$V_0 = \boldsymbol{e}^{\mathrm{T}}\left(\boldsymbol{I}_N \otimes \boldsymbol{P}\right)\boldsymbol{e}, \tag{9}$$

where $\boldsymbol{e} = \left[\boldsymbol{e}_1^{\mathrm{T}}, \boldsymbol{e}_2^{\mathrm{T}}, \ldots, \boldsymbol{e}_N^{\mathrm{T}}\right]^{\mathrm{T}}$ and $\boldsymbol{P} > \boldsymbol{0}$ such that

$$\boldsymbol{A}^{\mathrm{T}}\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - 2\boldsymbol{P}\boldsymbol{B}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P} + \sigma\boldsymbol{I} \le \boldsymbol{0}, \ \sigma > 0. \tag{10}$$

Then the derivative of V_0 along system (7) is

$$\begin{aligned} \dot{V}_0 = & \boldsymbol{e}^{\mathrm{T}}[\boldsymbol{I}_N \otimes (\boldsymbol{A}^{\mathrm{T}}\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A}) - \boldsymbol{L}_1 \otimes 2\boldsymbol{P}\boldsymbol{B}\boldsymbol{K}]\boldsymbol{e} \\ & + \boldsymbol{e}^{\mathrm{T}}\left\{\boldsymbol{L}_1^{-1}\boldsymbol{L}_2[w_{N+1}^n, w_{N+2}^n, \ldots, w_{N+M}^n]^{\mathrm{T}} \right. \\ & \left. \otimes 2\boldsymbol{P}\boldsymbol{B}\right\} - \boldsymbol{e}^{\mathrm{T}}(\boldsymbol{I}_N \otimes 2\boldsymbol{P}\boldsymbol{B}\boldsymbol{K})\boldsymbol{\omega}, \end{aligned} \tag{11}$$

where $\boldsymbol{\omega} = \left[\boldsymbol{\omega}_1^{\mathrm{T}}, \boldsymbol{\omega}_2^{\mathrm{T}}, \ldots, \boldsymbol{\omega}_N^{\mathrm{T}}\right]^{\mathrm{T}}$.

Choosing $\boldsymbol{K} = \alpha\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P}$, $\alpha > 0$, and $\alpha\lambda_1 \ge 1$ $(\lambda_1 = \lambda_{\min}(\boldsymbol{L}_1))$, then from inequality (10), we have

$$\begin{aligned} & \boldsymbol{e}^{\mathrm{T}}[\boldsymbol{I}_N \otimes (\boldsymbol{A}^{\mathrm{T}}\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A}) - \boldsymbol{L}_1 \otimes 2\boldsymbol{P}\boldsymbol{B}\boldsymbol{K}]\boldsymbol{e} \\ & \le -\sigma\boldsymbol{e}^{\mathrm{T}}\boldsymbol{e}. \end{aligned} \tag{12}$$

In addition, from triggered condition (8) and using Lemma 2, one has

$$\begin{aligned} & \boldsymbol{e}^{\mathrm{T}}\left\{\boldsymbol{L}_1^{-1}\boldsymbol{L}_2\left[w_{N+1}^{(n)}, w_{N+2}^{(n)}, \ldots, w_{N+M}^{(n)}\right]^{\mathrm{T}} \otimes 2\boldsymbol{P}\boldsymbol{B}\right\} \\ & \le 2\|\boldsymbol{e}\|\left\|\boldsymbol{L}_1^{-1}\right\|\|\boldsymbol{L}_2\|\|\boldsymbol{P}\|\sqrt{M}R_0 \\ & = 2\varsigma\|\boldsymbol{e}\|\sqrt{M} \\ & \le \varsigma\left(\frac{M\boldsymbol{e}^{\mathrm{T}}\boldsymbol{e}}{\beta_1} + \beta_1\right), \end{aligned} \tag{13}$$

$$\begin{aligned} -\boldsymbol{e}^{\mathrm{T}}\left(\boldsymbol{I}_N \otimes 2\boldsymbol{P}\boldsymbol{B}\boldsymbol{K}\right)\boldsymbol{\omega} & \le \gamma\|\boldsymbol{e}\|\|\boldsymbol{\omega}\| \\ & \le \frac{\gamma\boldsymbol{e}^{\mathrm{T}}\boldsymbol{e}}{2\beta_2} + \frac{\beta_2 N\Delta_{\max}^2}{2}, \end{aligned} \tag{14}$$

where $\varsigma = R_0\left\|\boldsymbol{L}_1^{-1}\right\|\|\boldsymbol{L}_2\|\|\boldsymbol{P}\|$, $\gamma = \lambda_{\max}\left(2\alpha\boldsymbol{P}\boldsymbol{B}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P}\right)$, $\Delta_{\max} = \max_i\{\Delta_i\}$, and β_1 and β_2 are positive constants to be designed.

According to inequalities (12)–(14), it can be easily determined that

$$\dot{V}_0 \le -q_1\boldsymbol{e}^{\mathrm{T}}\boldsymbol{e} + q_2, \tag{15}$$

where $q_1 = \sigma - \frac{\varsigma M}{\beta_1} - \frac{\gamma^2}{2\beta_2}$, $q_2 = \beta_1\varsigma + \frac{\beta_2 N\Delta_{\max}^2}{2}$, and β_1 and β_2 are positive constants making $q_1 > 0$.

Next, the design process of the control input u_i $(i \in \Gamma_1)$ will be given using the neural network technique and the backstepping method.

Define the following coordinate transformation:

$$\begin{cases} \xi_{i,1} = x_{i,1} - x_{i,1}^r, \\ \xi_{i,l} = x_{i,l} - \xi_{i,l-1}^*, \quad l = 2, 3, \ldots, n, \end{cases} \tag{16}$$

where $\xi_{i,l}^*$ $(l = 1, 2, \ldots, n)$ is the virtual controller to be designed later.

Step 1: Choose the Lyapunov function as follows:

$$V_{i,1} = \frac{1}{2}\xi_{i,1}^2 + \frac{1}{2}\tilde{\vartheta}_{i,1}^2, \tag{17}$$

where $\tilde{\vartheta}_{i,1} = \vartheta_{i,1} - \hat{\vartheta}_{i,1}$. Here, $\hat{\vartheta}_{i,1}$ is a dynamic auxiliary variable, and $\vartheta_{i,1}$ is a constant that needs to be designed later.

From Eq. (16), one has

$$\dot{\xi}_{i,1} = \xi_{i,2} + \xi_{i,1}^* + f_{i,1} - x_{i,2}^r. \tag{18}$$

Then it is easy to determine that

$$\dot{V}_{i,1} = \xi_{i,1}\left(\xi_{i,2} + \xi_{i,1}^* + f_{i,1} - x_{i,2}^r\right) - \dot{\hat{\vartheta}}_{i,1}\tilde{\vartheta}_{i,1}. \tag{19}$$

By Lemma 2, we have $\xi_{i,1}\xi_{i,2} \leq \frac{1}{2}\xi_{i,1}^2 + \frac{1}{2}\xi_{i,2}^2$, so Eq. (19) yields

$$\begin{aligned} \dot{V}_{i,1} \leq\ & \xi_{i,1}\left(f_{i,1} + \frac{1}{2}\xi_{i,1} - x_{i,2}^r\right) + \xi_{i,1}\xi_{i,1}^* \\ & + \frac{1}{2}\xi_{i,2}^2 - \dot{\hat{\vartheta}}_{i,1}\tilde{\vartheta}_{i,1}. \end{aligned} \tag{20}$$

Define

$$\Phi_{i,1}\left(\boldsymbol{X}_{i,1}\right) = f_{i,1} + \frac{1}{2}\xi_{i,1} - x_{i,2}^r, \tag{21}$$

where $\boldsymbol{X}_{i,1} = \left(x_{i,1}, x_{i,1}^r, x_{i,2}^r\right)^{\mathrm{T}}$.

RBFNNs are applied to approximate $\Phi_{i,1}(\cdot)$ on compact set Ω_i, because the nonlinear function $\Phi_{i,1}(\cdot)$ is totally unknown and cannot be determined to design protocols. There exists an RBFNN $(\boldsymbol{Z}_{i,1}^*)^{\mathrm{T}}\boldsymbol{H}_{i,1}\left(\boldsymbol{X}_{i,1}\right)$ for any given constant $\delta_1 > 0$ such that

$$\Phi_{i,1}\left(\boldsymbol{X}_{i,1}\right) = (\boldsymbol{Z}_{i,1}^*)^{\mathrm{T}}\boldsymbol{H}_{i,1}\left(\boldsymbol{X}_{i,1}\right) + \delta_{i,1}\left(\boldsymbol{X}_{i,1}\right), \tag{22}$$

where $\boldsymbol{Z}_{i,1}^*$ is an adjustable parameter vector, $\boldsymbol{H}_{i,1}\left(\boldsymbol{X}_{i,1}\right)$ the basis function vector satisfying $\boldsymbol{H}_{i,1}^{\mathrm{T}}\left(\boldsymbol{X}_{i,1}\right)\boldsymbol{H}_{i,1}\left(\boldsymbol{X}_{i,1}\right) \leq L_{i,1}$ ($L_{i,1}$ is the dimension of $\boldsymbol{H}_{i,1}(\boldsymbol{X}_{i,1})$), and $\delta_{i,1}\left(\boldsymbol{X}_{i,1}\right)$ the approximation error satisfying $\left|\delta_{i,1}\left(\boldsymbol{X}_{i,1}\right)\right| \leq \delta_1$.

According to Eq. (22), we can determine

$$\xi_{i,1}\Phi_{i,1}\left(\boldsymbol{X}_{i,1}\right) \leq \frac{\left\|\boldsymbol{Z}_{i,1}^*\right\|^2 L_{i,1}}{2\upsilon_1}\xi_{i,1}^2 + \frac{1}{2\mu_1}\xi_{i,1}^2 + c_1, \tag{23}$$

where $c_1 = \frac{\upsilon_1}{2} + \frac{\mu_1}{2}\delta_1^2$, and υ_1 and μ_1 are both positive constants.

Then we choose the virtual input as

$$\begin{cases} \xi_{i,1}^* = -\left(k_1 + \frac{1}{2\mu_1} + \hat{\vartheta}_{i,1}\right)\xi_{i,1}, \\ \dot{\hat{\vartheta}}_{i,1} = \xi_{i,1}^2 - \eta_1\hat{\vartheta}_{i,1}, \end{cases} \tag{24}$$

where $k_1 > 0$ and $\eta_1 > 0$.

Letting $\vartheta_{i,1} = \frac{\left\|\boldsymbol{Z}_{i,1}^*\right\|^2 L_{i,1}}{2\upsilon_1}$, one has

$$\dot{V}_{i,1} \leq -k_1\xi_{i,1}^2 + \eta_1\hat{\vartheta}_{i,1}\tilde{\vartheta}_{i,1} + \frac{1}{2}\xi_{i,2}^2 + c_1. \tag{25}$$

Step 2: Select the Lyapunov function with the following form:

$$V_{i,2} = V_{i,1} + \frac{1}{2}\xi_{i,2}^2 + \frac{1}{2}\tilde{\vartheta}_{i,2}^2, \tag{26}$$

where $\tilde{\vartheta}_{i,2} = \vartheta_{i,2} - \hat{\vartheta}_{i,2}$. Here, $\hat{\vartheta}_{i,2}$ is a dynamic auxiliary variable, and $\vartheta_{i,2}$ is a constant to be determined.

Based on the definition of $\xi_{i,2}$, we obtain

$$\dot{\xi}_{i,2} = \xi_{i,3} + \xi_{i,2}^* + f_{i,2} - \dot{\xi}_{i,1}^*, \tag{27}$$

where $\dot{\xi}_{i,1}^* = \frac{\partial\xi_{i,1}^*}{\partial x_{i,1}}\left(x_{i,2} + f_{i,1}\right) + \frac{\partial\xi_{i,1}^*}{\partial x_{i,1}^r}x_{i,2}^r + \frac{\partial\xi_{i,1}^*}{\partial\hat{\vartheta}_{i,1}}\dot{\hat{\vartheta}}_{i,1}$.

By Lemma 2, it can be found that

$$\begin{aligned} \dot{V}_{i,2} \leq\ & -k_1\xi_{i,1}^2 + \eta_1\hat{\vartheta}_{i,1}\tilde{\vartheta}_{i,1} + c_1 + \xi_{i,2}\xi_{i,2}^* + \xi_{i,2} \\ & \cdot\left(f_{i,2} + \xi_{i,2} - \dot{\xi}_{i,1}^*\right) - \dot{\hat{\vartheta}}_{i,2}\tilde{\vartheta}_{i,2} + \frac{1}{2}\xi_{i,3}^2. \end{aligned} \tag{28}$$

Define

$$\Phi_{i,2}\left(\boldsymbol{X}_{i,2}\right) = f_{i,2} + \xi_{i,2} - \dot{\xi}_{i,1}^*, \tag{29}$$

where $\boldsymbol{X}_{i,2} = (x_{i,1}, x_{i,2}, x_{i,1}^r, x_{i,2}^r, \hat{\vartheta}_{i,1})^{\mathrm{T}}$.

There exists a neural network $(\boldsymbol{Z}_{i,2}^*)^{\mathrm{T}}\boldsymbol{H}_{i,2}\left(\boldsymbol{X}_{i,2}\right)$ for any given constant $\delta_2 > 0$ such that

$$\Phi_{i,2}\left(\boldsymbol{X}_{i,2}\right) = (\boldsymbol{Z}_{i,2}^*)^{\mathrm{T}}\boldsymbol{H}_{i,2}\left(\boldsymbol{X}_{i,2}\right) + \delta_{i,2}\left(\boldsymbol{X}_{i,2}\right), \tag{30}$$

where $\boldsymbol{H}_{i,2}^{\mathrm{T}}\left(\boldsymbol{X}_{i,2}\right)\boldsymbol{H}_{i,2}\left(\boldsymbol{X}_{i,2}\right) \leq L_{i,2}$ and $\left|\delta_{i,2}\left(\boldsymbol{X}_{i,2}\right)\right| \leq \delta_2$.

In view of Eq. (30), we can determine that

$$\xi_{i,2}\Phi_{i,2}\left(\boldsymbol{X}_{i,2}\right) \leq \frac{\left\|\boldsymbol{Z}_{i,2}^*\right\|^2 L_{i,2}}{2\upsilon_2}\xi_{i,2}^2 + \frac{1}{2\mu_2}\xi_{i,2}^2 + c_2, \tag{31}$$

where $c_2 = \frac{\upsilon_2}{2} + \frac{\mu_2}{2}\delta_2^2$, and υ_2 and μ_2 are both positive constants.

Then we choose the virtual input as

$$\begin{cases} \xi_{i,2}^* = -\left(k_2 + \frac{1}{2\mu_2} + \hat{\vartheta}_{i,2}\right)\xi_{i,2}, \\ \dot{\hat{\vartheta}}_{i,2} = \xi_{i,2}^2 - \eta_2\hat{\vartheta}_{i,2}, \end{cases} \tag{32}$$

where $k_2 > 0$ and $\eta_2 > 0$.

Letting $\vartheta_{i,2} = \frac{\left\|\boldsymbol{Z}_{i,2}^*\right\|^2 L_{i,2}}{2\upsilon_2}$, we have

$$\dot{V}_{i,2} \leq \sum_{j=1}^{2}\left(-k_j\xi_{i,j}^2 + \eta_j\hat{\vartheta}_{i,j}\tilde{\vartheta}_{i,j} + c_j\right) + \frac{1}{2}\xi_{i,3}^2. \tag{33}$$

Step l ($l = 3, 4, \ldots, n-1$): Supposing that under the Lyapunov function

$$V_{i,l-1} = \sum_{j=1}^{l-1}\left(\tfrac{1}{2}\xi_{i,j}^2 + \tfrac{1}{2}\tilde{\vartheta}_{i,j}^2\right) \tag{34}$$

where

$$\tilde{\vartheta}_{i,l-1} = \vartheta_{i,l-1} - \hat{\vartheta}_{i,l-1} \tag{35}$$

and the virtual input

$$\begin{cases} \xi_{i,l-1}^* = -\left(k_{l-1} + \frac{1}{2\mu_{l-1}} + \hat{\vartheta}_{i,l-1}\right)\xi_{i,l-1} \\ \dot{\hat{\vartheta}}_{i,l-1} = \xi_{i,l-1}^2 - \eta_{l-1}\hat{\vartheta}_{i,l-1} \end{cases} \tag{36}$$

at step $l-1$, we can determine that

$$\dot{V}_{i,l-1} \leq \sum_{j=1}^{l-1}\left(-k_1\xi_{i,j}^2 + \eta_j\hat{\vartheta}_{i,j}\tilde{\vartheta}_{i,j} + c_j\right) + \frac{1}{2}\xi_{i,l}^2, \tag{37}$$

where $\vartheta_{i,l-1} = \frac{\left\|\boldsymbol{Z}_{i,l-1}^*\right\|^2 L_{i,l-1}}{2\upsilon_{l-1}}$ and $c_{l-1} = \frac{\upsilon_{l-1}}{2} + \frac{\mu_{l-1}}{2}\delta_{l-1}^2$.

Based on the statement above, the Lyapunov function is chosen as

$$V_{i,l} = V_{i,l-1} + \frac{1}{2}\xi_{i,l}^2 + \frac{1}{2}\tilde{\vartheta}_{i,l}^2, \tag{38}$$

where $\tilde{\vartheta}_{i,l} = \vartheta_{i,l} - \hat{\vartheta}_{i,l}$. Here, $\hat{\vartheta}_{i,l}$ is a dynamic auxiliary variable, and $\vartheta_{i,l}$ is a constant to be determined.

In view of the definition of $\xi_{i,l}$, one has

$$\dot{\xi}_{i,l} = \xi_{i,l+1} + \xi_{i,l}^* + f_{i,l} - \dot{\xi}_{i,l-1}^*, \tag{39}$$

where $\dot{\xi}_{i,l-1}^* = \sum_{j=1}^{l-1}\frac{\partial\xi_{i,l-1}^*}{\partial x_{i,j}^r}x_{i,j+1}^r + \sum_{j=1}^{l-1}\frac{\partial\xi_{i,l-1}^*}{\partial\hat{\theta}_{i,j}}\dot{\hat{\vartheta}}_{i,j} + \sum_{j=1}^{l-1}\frac{\partial\xi_{i,l-1}^*}{\partial x_{i,j}}\left(x_{i,j+1} + f_{i,j}\right)$.

Substituting Eq. (39) into Eq. (38) yields

$$\begin{aligned}\dot{V}_{i,l} \leq& \sum_{j=1}^{l-1}\left(-k_j\xi_{i,j}^2 + \eta_j\hat{\vartheta}_{i,j}\tilde{\vartheta}_{i,j} + c_j\right) + \xi_{i,l}\xi_{i,l}^* + \xi_{i,l} \\ &\cdot\left(f_{i,l} + \xi_{i,l} - \dot{\xi}_{i,l-1}^*\right) - \dot{\hat{\vartheta}}_{i,l}\tilde{\vartheta}_{i,l} + \frac{1}{2}\xi_{i,l+1}^2.\end{aligned} \tag{40}$$

Define

$$\varPhi_{i,l}\left(\boldsymbol{X}_{i,l}\right) = f_{i,l} + \xi_{i,l} - \dot{\xi}_{i,l-1}^*, \tag{41}$$

where $\boldsymbol{X}_{i,l} = \left(x_{i,1},\ldots,x_{i,l},x_{i,1}^r,\ldots,x_{i,l}^r,\hat{\vartheta}_{i,1},\ldots,\hat{\vartheta}_{i,l}\right)^{\mathrm{T}}$.

There exists a neural network $(\boldsymbol{Z}_{i,l}^*)^{\mathrm{T}}\boldsymbol{H}_{i,l}\left(\boldsymbol{X}_{i,l}\right)$ for any given constant $\delta_l > 0$ such that

$$\varPhi_{i,l}\left(\boldsymbol{X}_{i,l}\right) = (\boldsymbol{Z}_{i,l}^*)^{\mathrm{T}}\boldsymbol{H}_{i,l}\left(\boldsymbol{X}_{i,l}\right) + \delta_{i,l}\left(\boldsymbol{X}_{i,l}\right), \tag{42}$$

where $\boldsymbol{H}_{i,l}^{\mathrm{T}}\left(\boldsymbol{X}_{i,l}\right)\boldsymbol{H}_{i,l}\left(\boldsymbol{X}_{i,l}\right) \leq L_{i,l}$ and $\left|\delta_{i,l}\left(\boldsymbol{X}_{i,l}\right)\right| \leq \delta_l$.

In view of Eq. (42), we can determine that

$$\xi_{i,l}\varPhi_{i,l}\left(\boldsymbol{X}_{i,l}\right) \leq \frac{\left\|\boldsymbol{Z}_{i,l}^*\right\|^2 L_{i,l}}{2\upsilon_l}\xi_{i,l}^2 + \frac{1}{2\mu_l}\xi_{i,l}^2 + c_l, \tag{43}$$

where $c_l = \frac{\upsilon_l}{l} + \frac{\mu_l}{2}\delta_l^2$, and υ_l and μ_l are both positive constants.

Then we choose the virtual input as

$$\begin{cases} \xi_{i,l}^* = -\left(k_l + \frac{1}{2\mu_l} + \hat{\vartheta}_{i,l}\right)\xi_{i,l}, \\ \dot{\hat{\vartheta}}_{i,l} = \xi_{i,l}^2 - \eta_l\hat{\vartheta}_{i,l}, \end{cases} \tag{44}$$

where $k_l > 0$ and $\eta_l > 0$.

Letting $\vartheta_{i,l} = \frac{\left\|\boldsymbol{Z}_{i,l}^*\right\|^2 L_{i,l}}{2\upsilon_l}$, we have

$$\dot{V}_{i,l} \leq \sum_{j=1}^{l}\left(-k_j\xi_{i,j}^2 + \eta_j\hat{\vartheta}_{i,j}\tilde{\vartheta}_{i,j} + c_j\right) + \frac{1}{2}\xi_{i,l+1}^2. \tag{45}$$

Step n: Determine the Lyapunov function as

$$V_{i,n} = \sum_{j=1}^{n}\left(\frac{1}{2}\xi_{i,j}^2 + \frac{1}{2}\tilde{\vartheta}_{i,j}^2\right), \tag{46}$$

where $\tilde{\vartheta}_{i,n} = \vartheta_{i,n} - \hat{\vartheta}_{i,n}$. Here, $\hat{\vartheta}_{i,n}$ is a dynamic auxiliary variable, and $\vartheta_{i,n}$ is unknown and needs to be determined later.

Since $\dot{\xi}_{i,n} = u_i + f_{i,n} - \dot{\xi}_{i,n-1}^*$, it can be easily determined that

$$\begin{aligned}\dot{V}_{i,n} \leq& \sum_{j=1}^{n-1}\left(-k_j\xi_{i,j}^2 + \eta_j\hat{\vartheta}_{i,j}\tilde{\vartheta}_{i,j} + c_j\right) + \frac{1}{2}\xi_{i,n}^2 \\ &+ \xi_{i,n}\left(u_i + f_{i,n} - \dot{\xi}_{i,n-1}^*\right) - \dot{\hat{\vartheta}}_{i,n}\tilde{\vartheta}_{i,n}.\end{aligned} \tag{47}$$

Define

$$\Phi_{i,n}\left(\boldsymbol{X}_{i,n}\right)=f_{i,n}+\frac{1}{2}\xi_{i,n}-\dot{\xi}_{i,n-1}^{*}, \tag{48}$$

where $\boldsymbol{X}_{i,n}=\Big(x_{i,1},\ldots,x_{i,n},x_{i,1}^{r},\ldots,x_{i,n}^{r},\hat{\vartheta}_{i,1},\ldots,\hat{\vartheta}_{i,n}\Big)^{\mathrm{T}}$.

There exists a neural network $(\boldsymbol{Z}_{i,n}^{*})^{\mathrm{T}}\boldsymbol{H}_{i,n}\left(\boldsymbol{X}_{i,n}\right)$ for any given constant $\delta_n>0$ such that

$$\Phi_{i,n}\left(\boldsymbol{X}_{i,n}\right)=(\boldsymbol{Z}_{i,n}^{*})^{\mathrm{T}}\boldsymbol{H}_{i,n}\left(\boldsymbol{X}_{i,n}\right)+\delta_{i,n}\left(\boldsymbol{X}_{i,n}\right), \tag{49}$$

where $\boldsymbol{H}_{i,n}^{\mathrm{T}}\left(\boldsymbol{X}_{i,n}\right)\boldsymbol{H}_{i,n}\left(\boldsymbol{X}_{i,n}\right)\leq L_{i,n}$ and $|\delta_{i,n}\left(\boldsymbol{X}_{i,n}\right)|\leq\delta_n$.

In view of Eq. (49), we can determine that

$$\xi_{i,n}\Phi_{i,n}\left(\boldsymbol{X}_{i,n}\right)\leq\frac{\left\|\boldsymbol{Z}_{i,n}^{*}\right\|^{2}L_{i,n}}{2\upsilon_n}\xi_{i,n}^{2}+\frac{1}{2\mu_n}\xi_{i,n}^{2}+c_n, \tag{50}$$

where $c_n=\dfrac{\upsilon_n}{n}+\dfrac{\mu_n}{2}\delta_n^2$, and υ_n and μ_n are both positive constants.

Then, we choose the input control signal as

$$u_i=-\left(k_n+\frac{1}{2\mu_n}+\hat{\vartheta}_{i,n}\right)\xi_{i,n}, \tag{51}$$

$$\dot{\hat{\vartheta}}_{i,n}=\xi_{i,n}^{2}-\eta_n\hat{\vartheta}_{i,n}, \tag{52}$$

where $k_n>0$ and $\eta_n>0$.

Letting $\vartheta_{i,n}=\dfrac{\left\|\boldsymbol{Z}_{i,n}^{*}\right\|^{2}L_{i,n}}{2\upsilon_n}$, we have

$$\dot{V}_{i,n}\leq\sum_{j=1}^{n}\Big(-k_j\xi_{i,j}^{2}+\eta_j\hat{\vartheta}_{i,j}\tilde{\vartheta}_{i,j}+c_j\Big). \tag{53}$$

The controller design procedure can be summarized as follows:

1. According to the order of MAS (4), $\boldsymbol{A}$ and $\boldsymbol{B}$ can be determined. Then select σ and solve $\boldsymbol{P}$ to satisfy inequality (10).

2. Calculate $\lambda_1=\lambda_{\min}\left(\boldsymbol{L}_1\right)$ and choose $\alpha>0$ so that $\alpha\lambda_1\geq1$. Then $\boldsymbol{K}$ can be obtained by $\boldsymbol{K}=\alpha\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P}$, and the dynamics of $\boldsymbol{x}_i^r$ can be determined by Eq. (7).

3. Choose k_l, μ_l, and η_l to design the virtual input $\xi_{i,l}^{*}$ at step l $(l=1,2,\ldots,n-1)$.

4. Select k_n, μ_n, and η_n to determine the control input u_i at step n.

3.2 Convergence analysis

Theorem 1 Consider MAS (4) under the protocols given in Eqs. (7), (51), and (52), and the triggered condition is designed as Eq. (8). If Assumptions 1 and 2 hold, then the practical containment control objective can be achieved.

Proof From inequality (15), we have

$$\dot{V}_0\leq-\frac{q_1}{\|\boldsymbol{P}\|}V_0+q_2. \tag{54}$$

Then from inequality (54), it can be determined that $\forall i\in\Gamma_1$,

$$\lim_{t\to\infty}\left|x_{i,1}^{r}-\tilde{w}_{i,1}\right|\leq\varepsilon_1\triangleq\sqrt{\frac{q_2\left\|\boldsymbol{P}\right\|}{q_1\lambda_{\min}\left(\boldsymbol{P}\right)}}. \tag{55}$$

By applying Lemma 2, for $j=1,2,\ldots,n$, we have

$$\hat{\vartheta}_{i,j}\tilde{\vartheta}_{i,j}=-\tilde{\vartheta}_{i,j}^{2}+\vartheta_{i,j}\tilde{\vartheta}_{i,j}\leq-\frac{1}{2}\tilde{\vartheta}_{i,j}^{2}+\frac{1}{2}\vartheta_{i,j}^{2}. \tag{56}$$

Then inequality (53) can be rewritten as

$$\dot{V}_{i,n}\leq-\varpi V_{i,n}+C, \tag{57}$$

where $\varpi=\min\left\{2k_1,2k_2,\ldots,2k_n,\eta_1,\eta_2,\ldots,\eta_n\right\}$ and $C=\sum\limits_{j=1}^{n}\left(c_j+\dfrac{1}{2}\eta_j\vartheta_{i,j}^{2}\right)$.

Thus, we can obtain

$$\lim_{t\to\infty}\left|y_i-x_{i,1}^{r}\right|\leq\varepsilon_2\triangleq\sqrt{\frac{2C}{\varpi}}. \tag{58}$$

Combining inequalities (55) and (58), it can be concluded that

$$\lim_{t\to\infty}\left|y_i-\tilde{w}_{i,1}\right|\leq\varepsilon_0\triangleq\varepsilon_1+\varepsilon_2,\ \forall i\in\Gamma_1. \tag{59}$$

Remark 2 On account of the triggered condition (8) and inequality (54), it is easy to determine that the frequency of the events and the convergence error for each follower are affected by Δ_i $(i\in\Gamma_1)$. Specifically, as Δ_i increases, the convergence error becomes larger. However, a relatively large Δ_i will contribute to a relatively low frequency of the events.

3.3 Exclusion of Zeno behavior

Theorem 2 Consider MAS (4) under the protocols given in Eqs. (7), (51), and (52), and the triggered condition is designed as Eq. (8). If Assumptions 1 and 2 hold, no agent exhibits Zeno behavior.

Proof According to Eqs. (6), (8), and (11), we can easily find

$$\begin{aligned}\frac{\mathrm{d}\left\|\boldsymbol{\omega}_i\right\|}{\mathrm{d}t} &\leq \left\|\dot{\boldsymbol{\omega}}_i\right\| \leq \|\dot{\boldsymbol{\omega}}\| = \|\dot{\boldsymbol{\rho}}\| \leq \left\|\boldsymbol{L}_1\right\|\|\dot{\boldsymbol{e}}\| \\ &\leq \left\|\boldsymbol{L}_1\right\|\|\boldsymbol{Q}\|\|\boldsymbol{e}\| + \left\|\boldsymbol{L}_1\right\| \varPi_0, \end{aligned} \tag{60}$$

where $\boldsymbol{Q} = \boldsymbol{I}_N \otimes \boldsymbol{A} - \boldsymbol{L}_1 \otimes \alpha \boldsymbol{B}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P}$ and $\varPi_0 = \left\|\boldsymbol{L}_1^{-1}\right\|\left\|\boldsymbol{L}_2\right\| \sqrt{M} R_0 + \alpha\|\boldsymbol{P}\| \sqrt{N} \varDelta_{\max}$.

From inequality (15), we can determine that

$$\|\boldsymbol{e}\| \leq \sqrt{\frac{\max\left\{V_0(0), \dfrac{q_2\|\boldsymbol{P}\|}{q_1}\right\}}{\lambda_{\min}(\boldsymbol{P})}} \triangleq \varPi_1. \tag{61}$$

Thus, $\forall t \in \left[t_k^i, t_{k+1}^i\right)$ $(i \in \varGamma_1)$, inequality (60) can be transformed to

$$\frac{\mathrm{d}\left\|\boldsymbol{\omega}_i\right\|}{\mathrm{d}t} \leq \left\|\boldsymbol{L}_1\right\|\|\boldsymbol{Q}\| \varPi_1 + \left\|\boldsymbol{L}_1\right\| \varPi_0 \triangleq \varPi, \tag{62}$$

and it is obvious that under the event-triggered condition (8), we can determine that $t_{k+1}^i - t_k^i \geq \tau_i \triangleq \dfrac{\varDelta_i}{\varPi}$.

3.4 Self-triggered control scheme design

Under the event-triggered control scheme, continuous state measurement is required to check condition (8) in $\left[t_k^i,\ t_{k+1}^i\right)$ $(i \in \varGamma_1, k = 0, 1, \ldots)$. To avoid such continuous communication and state monitoring among agents, a self-triggered control scheme is designed.

Define

$$\begin{aligned}\varphi\left(t_k^i\right) = &\sum_{j=2}^{n}\left|\rho_{i,j}(t_k^i)\right| \\ &+ k_1\left|\sum_{j=1}^{N} a_{ij}\rho_{j,1}(t_{k_j}^j) - \sum_{l=1}^{N+M} a_{il}\rho_{i,1}\left(t_k^i\right)\right| \\ &+ k_2\left|\sum_{j=1}^{N} a_{ij}\rho_{j,2}(t_{k_j}^j) - \sum_{l=1}^{N+M} a_{il}\rho_{i,2}(t_k^i)\right| \\ &+ \ldots \\ &+ k_n\left|\sum_{j=1}^{N} a_{ij}\rho_{j,n}(t_{k_j}^j) - \sum_{l=1}^{N+M} a_{il}\rho_{i,n}(t_k^i)\right|, \end{aligned} \tag{63}$$

where $t_{k_j}^j$ represents the last triggered instant of agent j before t_k^i.

Suppose that the last triggered instant of agent i $(i \in \varGamma_1)$ is t_m^i, and define $t^* = t_m^i + \ln\left(\frac{\varDelta_i}{\varphi(t_k^i)+g_i h} + 1\right)$, where $\varDelta_i > 0$ and $g_i = \sum_{k=N+1}^{N+M} a_{ik}$. If no neighbor is triggered before t^*, then $t_{m+1}^i = t^*$. If at t_{p_1} $(t_{p_1} < t^*)$ one neighbor p_1 is triggered, let $\varDelta_i' = \varDelta_i - [\varphi\left(t_k^i\right) + g_i h](\mathrm{e}^{t_{p_1} - t_m^i} - 1)$. If $\varDelta_i' \leq 0$, the next triggered instant is t_{p_1}; otherwise, recalculate $\varphi\left(t_k^i\right)$ by substituting $\rho_{p_1,1}\left(t_{p_1}\right), \rho_{p_1,2}\left(t_{p_1}\right), \ldots, \rho_{p_1,n}\left(t_{p_1}\right)$ into Eq. (63). Then we can determine that $t^* = t_{p_1} + \ln\left(\frac{\varDelta_i'}{\varphi(t_k^i)+g_i h} + 1\right)$. If at t_{p_2} $(t_{p_2} < t^*)$ another neighbor p_2 is triggered, let $\varDelta_i'' = \varDelta_i' - [\varphi\left(t_k^i\right) + g_i h]\left(\mathrm{e}^{t_{p_2} - t_{p_1}} - 1\right)$. If $\varDelta_i'' \leq 0$, $t_{m+1}^i = t_{p_2}$; otherwise, update $\varphi\left(t_k^i\right)$ by substituting $\rho_{p_2,1}\left(t_{p_2}\right), \rho_{p_2,2}\left(t_{p_2}\right), \ldots, \rho_{p_2,n}\left(t_{p_2}\right)$ into Eq. (63), which makes $t^* = t_{p_2} + \ln\left(\frac{\varDelta_i''}{\varphi(t_k^i)+g_i h} + 1\right)$. This process will be repeated until no triggering exists among the neighbors of agent i before t^*. Then, the next triggered instant t_{m+1}^i can be finally determined as $t_{m+1}^i = t^*$.

Theorem 3 Considering MAS (4) under the self-triggered rule, if Assumptions 1 and 2 hold and the protocols are given in Eqs. (7), (51), and (52), practical containment control can be achieved and no agent exhibits Zeno behavior.

Proof First, it is easy to determine that for any $t \in \left[t_k^i, t_{k+1}^i\right)$,

$$\begin{aligned} &\frac{\mathrm{d}\left\|\boldsymbol{\rho}_i(t) - \boldsymbol{\rho}_i\left(t_k^i\right)\right\|}{\mathrm{d}t} \\ \leq &\left\|\dot{\boldsymbol{\rho}}_i(t)\right\| \leq \left\|\left(\dot{\rho}_{i,1}, \dot{\rho}_{i,2}, \ldots, \dot{\rho}_{i,n-1}\right)^{\mathrm{T}}\right\| + \left|\dot{\rho}_{i,n}\right| \\ \leq &\left\|\begin{array}{c}\rho_{i,2}(t) - \rho_{i,2}\left(t_k^i\right) \\ \rho_{i,3}(t) - \rho_{i,3}\left(t_k^i\right) \\ \vdots \\ \rho_{i,n}(t) - \rho_{i,n}\left(t_k^i\right)\end{array}\right\| + \left\|\begin{array}{c}\rho_{i,2}\left(t_k^i\right) \\ \rho_{i,3}\left(t_k^i\right) \\ \vdots \\ \rho_{i,n}\left(t_k^i\right)\end{array}\right\| \\ &+ \left|\sum_{j=1}^{N} a_{ij}\left(u_j^r - u_i^r\right) + \sum_{k=N+1}^{N+M} a_{ik}\left(w_k^{(n)} - u_i^r\right)\right| \\ \leq &\left\|\boldsymbol{\rho}_i(t) - \boldsymbol{\rho}_i\left(t_k^i\right)\right\| + \varphi\left(t_k^i\right) + g_i h, \end{aligned} \tag{64}$$

where $u_i^r = k_1\rho_{i,1}\left(t_k^i\right) + k_2\rho_{i,2}\left(t_k^i\right) + \ldots + k_n\rho_{i,n}\left(t_k^i\right)$.

Then it can be obtained that $\left\|\boldsymbol{\rho}_i(t) - \boldsymbol{\rho}_i\left(t_k\right)\right\| \leq \left(\mathrm{e}^{t - t_k^i} - 1\right)\left[\varphi\left(t_k\right) + g_i h\right]$. From the self-triggered rule, we can further determine that $\left\|\boldsymbol{\omega}_i\right\| = \left\|\boldsymbol{\rho}_i(t) - \boldsymbol{\rho}_i\left(t_k\right)\right\| \leq \varDelta_i$.

To construct the Lyapunov function (9), by taking its derivative, we have

$$\dot{V}_0 \leq -\chi_1 \boldsymbol{e}^{\mathrm{T}} \boldsymbol{e} + \chi_2, \tag{65}$$

where $\chi_1 = \sigma - \frac{\varsigma M}{b_1} - \frac{\gamma^2}{2b_2}$, $\chi_2 = b_1\varsigma + \frac{b_2 N \Delta_{\max}^2}{2}$, $\varsigma = R_0 \left\|\boldsymbol{L}_1^{-1}\right\| \left\|\boldsymbol{L}_2\right\| \left\|\boldsymbol{P}\right\|$, $\gamma = \lambda_{\max}\left(2\alpha \boldsymbol{PBB}^{\mathrm{T}}\boldsymbol{P}\right)$, $\Delta_{\max} = \max_i \{\Delta_i\}$, and b_1 and b_2 are positive constants making $\chi_1 > 0$.

Following the proof procedure for Theorem 1, we can prove that the practical containment control objective can be achieved.

In addition, according to Eq. (63) we can determine that

$$\begin{aligned} \varphi\left(t_k^i\right) \leq & \sum_{j=2}^{n} \|\boldsymbol{\rho}\| + k_1\left(2\iota_i + g_i\right)\|\boldsymbol{\rho}\| + k_2\left(2\iota_i + g_i\right)\|\boldsymbol{\rho}\| \\ & + \ldots + k_n\left(2\iota_i + g_i\right)\|\boldsymbol{\rho}\| \\ \leq & \left(n - 1 + \left(2\iota_i + g_i\right)\sum_{j=1}^{n} k_j\right)\|\boldsymbol{\rho}\|. \end{aligned} \tag{66}$$

Then from inequality (65), we have

$$\|\boldsymbol{e}\| \leq \sqrt{\frac{\max\left\{V_0(0), \frac{\chi_2\|\boldsymbol{P}\|}{\chi_1}\right\}}{\lambda_{\min}(\boldsymbol{P})}} \triangleq \Pi_2. \tag{67}$$

In view of inequality (67), we can determine from inequality (66) that

$$\varphi\left(t_k^i\right) \leq \left(n - 1 + \left(2\iota_i + g_i\right)\sum_{j=1}^{n} k_j\right)\|\boldsymbol{L}_1\|\|\boldsymbol{e}\| \triangleq \Omega, \tag{68}$$

where $\iota_i = \sum_{j=1}^{N} a_{ij}$, $i \in \Gamma_1$.

Thus, from inequality (68) and the self-triggered rule, it can be easily determined that Zeno behavior can be excluded.

Remark 3 It is worth mentioning that under the event-triggered scheme, continuous monitoring of the triggered condition (8) is still required for follower i while $t_k^i + \tau_i \leq t \leq t_{k+1}^i$ $(i \in \Gamma_1)$, which means that follower i needs to continuously communicate with its neighbors during this period. In contrast, the self-triggered rule overcomes this problem because the next triggered time for follower i is decided by updating only Eq. (63) and computing t^* until there is no neighbor triggered before the latest recalculated t^*.

4 Simulations

In this section, we will provide two numerical simulation examples using MATLAB/Simulink to illustrate the validity of the obtained results. Consider a third-order MAS consisting of four followers (labeled 1, 2, 3, and 4) and two leaders (labeled 5 and 6), which means that $N = 4$, $M = 2$, and $n = 3$. The communication topology among agents is shown in Fig. 1, where $a_{ij} = 1$ if $(j, i) \in E$ and $a_{ij} = 0$ otherwise.

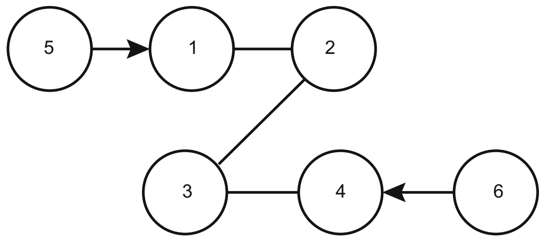

Fig. 1 Topology of the multi-agent system

First, the dynamics of the followers is given as

$$\begin{cases} \dot{x}_{i,1} = x_{i,2} + 0.02x_{i,1}^{2/3}, \\ \dot{x}_{i,2} = x_{i,3} + 0.1\left(\cos x_{i,1} + \sin x_{i,2}\right), \\ \dot{x}_{i,3} = u_i + 0.2\left(\sin x_{i,2} + \cos x_{i,3}\right), \\ y_i = x_{i,1}, \ i = 1, 2, \end{cases} \tag{69}$$

$$\begin{cases} \dot{x}_{i,1} = x_{i,2} + 0.1\sin x_{i,1}, \\ \dot{x}_{i,2} = x_{i,3} + 0.08\left(\cos x_{i,1} + \sin x_{i,2}^2\right), \\ \dot{x}_{i,3} = u_i + 0.1\left(\cos x_{i,2} \sin x_{i,3}\right), \\ y_i = x_{i,1}, \ i = 3, 4. \end{cases} \tag{70}$$

Then, the initial states of the followers are given as

$$\begin{cases} \left[x_{1,1}(0), x_{1,2}(0), x_{1,3}(0)\right] = [-2, 1, 0.1], \\ \left[x_{2,1}(0), x_{2,2}(0), x_{2,3}(0)\right] = [4, -0.5, 1], \\ \left[x_{3,1}(0), x_{3,2}(0), x_{3,3}(0)\right] = [1, -1, 0.2], \\ \left[x_{4,1}(0), x_{4,2}(0), x_{4,3}(0)\right] = [-3, 0.5, -0.2], \end{cases} \tag{71}$$

and the initial states of the objective route for each follower are shown as

$$\begin{cases} \left[x_{1,1}^r(0), x_{1,2}^r(0), x_{1,3}^r(0)\right] = [5, -1, 0.1], \\ \left[x_{2,1}^r(0), x_{2,2}^r(0), x_{2,3}^r(0)\right] = [-8, 1, -0.2], \\ \left[x_{3,1}^r(0), x_{3,2}^r(0), x_{3,3}^r(0)\right] = [4, 0.3, 0.5], \\ \left[x_{4,1}^r(0), x_{4,2}^r(0), {x_{4,3}}^r(0)\right] = [3, 2, 0.1]. \end{cases} \tag{72}$$

In addition, two leader signals are determined as $w_5^{(3)} = 0.1\left(w_5 \sin t + \dot{w}_5 \cos t + \ddot{w}_5 \sin t\right)$ and $w_6^{(3)} =$

$0.1\left(w_6 \cos t + \dot{w}_6 \cos t + \ddot{w}_6 \sin t\right)$. The initial states for leaders are $\left[w_5\left(0\right), \dot{w}_5\left(0\right), \ddot{w}_5\left(0\right)\right] = [1, -0.5, 0.1]$ and $\left[w_6\left(0\right), \dot{w}_6\left(0\right), \ddot{w}_6\left(0\right)\right] = [-1, 0.5, 0.2]$.

Example 1 In this example, the containment control protocol consisting of Eqs. (7), (51), and (52) with the event-triggered condition (8) is adopted.

Let $\sigma = 4$. We can determine that $\boldsymbol{P} = \begin{bmatrix} 20 & 14 & 6 \\ 14 & 25 & 9 \\ 6 & 9 & 12 \end{bmatrix}$. Because $\lambda_1 = 0.382$, we choose $\alpha = 3$ so that $\boldsymbol{K} = [18, 27, 36]$. Other parameters are given as $k_1 = k_2 = k_3 = 8$, $\mu_1 = \mu_2 = \mu_3 = 2$, $\eta_1 = \eta_2 = \eta_3 = 12$, and $\Delta_i = 0.1$ $(i \in \Gamma_1)$ in the triggered condition.

Simulation results are shown in Figs. 2–5. Fig. 2 shows that the objective route of each follower can converge to the convex hull spanned by the two leader signals with a bounded error. Fig. 3 shows that the practical containment control objective is achieved. Fig. 4 presents the followers' control inputs. Triggered instants are given in Fig. 5.

Example 2 In this example, the protocol consisting of Eqs. (7), (51), and (52) with the self-triggered rule is considered. Parameters are the same as those in Example 1. Simulation results are shown in Figs. 6–9. Fig. 6 indicates that the objective routes achieve practical containment control. Fig. 7 shows that the outputs converge to the convex hull spanned by the leaders. Fig. 8 presents the control inputs of the followers. Triggered instants are given in Fig. 9.

According to the simulation results, we can determine that both event-triggered and self-triggered control schemes can make followers achieve practical containment control. Under the proposed control method, each follower communicates with only its neighbors at the triggered instant, which reduces the consumption of communication resources.

In addition, from Figs. 5 and 9, we can see that the followers' triggered instants are different under the event-triggered and self-triggered schemes. Continuous monitoring of the error state is not required between two adjacent triggered instants under the self-triggered scheme, but more triggers exist for the followers.

5 Conclusions

In this study, we have discussed the containment control problem for a class of high-order

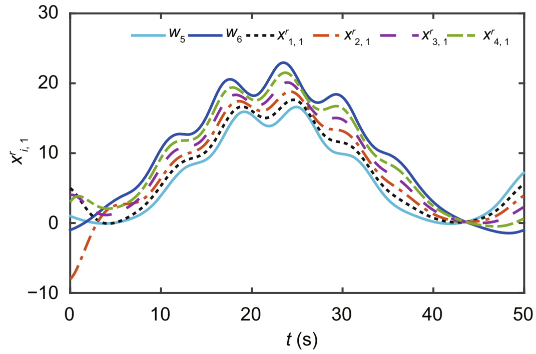

Fig. 2 Objective routes of the followers under the event-triggered scheme

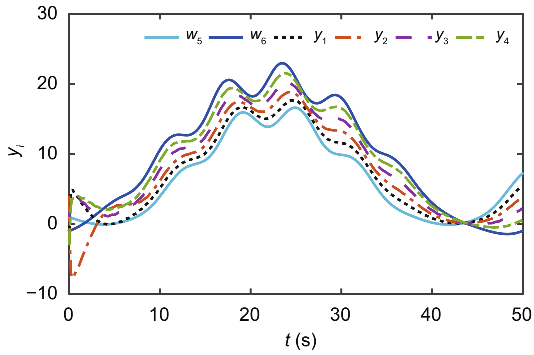

Fig. 3 Outputs of the followers under the event-triggered scheme

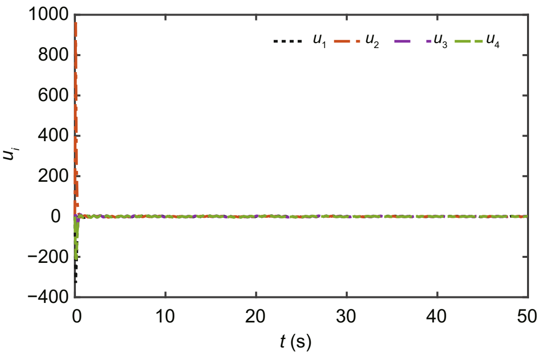

Fig. 4 Control inputs of the followers under the event-triggered scheme

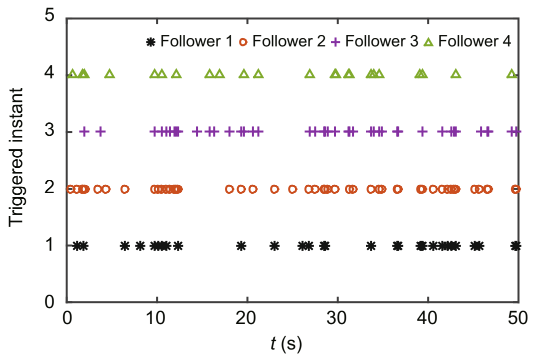

Fig. 5 Triggered instants of the followers under the event-triggered scheme

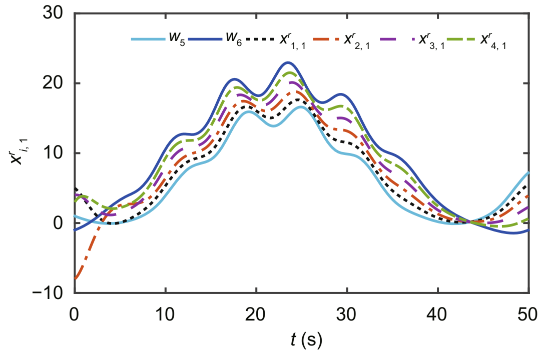

Fig. 6 Objective routes of the followers under the self-triggered scheme

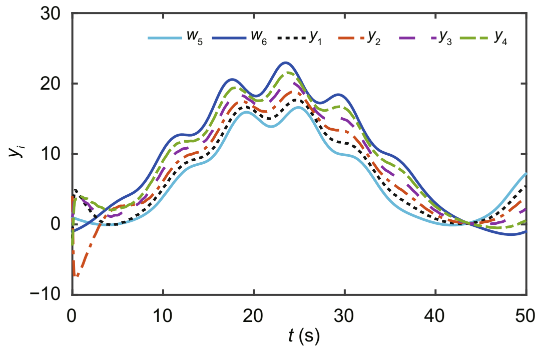

Fig. 7 Outputs of the followers under the self-triggered scheme

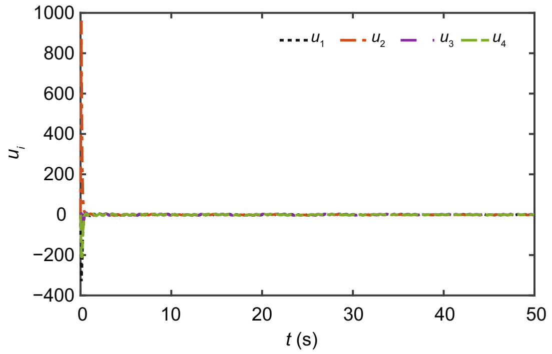

Fig. 8 Control inputs of the followers under the self-triggered scheme

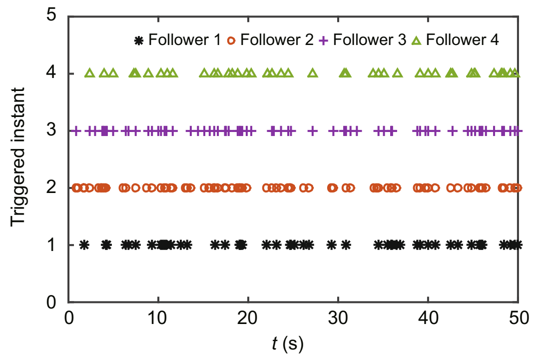

Fig. 9 Triggered instants of the followers under the self-triggered scheme

heterogeneous nonlinear multi-agent systems (MASs) with the distributed event-triggered mechanism. By developing an appropriate distributed event-triggered control scheme, the practical containment control objective has been achieved and Zeno behavior has been avoided. Then a distributed self-triggered control scheme has been proposed. Finally, we have provided two simulation examples to validate the correctness of the main results. In future study, we will focus on event-triggered containment control for nonlinear MASs with switched topologies and finite-time event-triggered containment control for MASs.

Contributors

Zheng-rong XIANG provided the idea. Ya-ni SUN designed the research. Ya-ni SUN and Wen-cheng ZOU conducted the simulations. Ya-ni SUN drafted the manuscript. Jian GUO and Zheng-rong XIANG helped organize the manuscript. Ya-ni SUN and Zheng-rong XIANG revised and finalized the paper.

Compliance with ethics guidelines

Ya-ni SUN, Wen-cheng ZOU, Jian GUO, and Zheng-rong XIANG declare that they have no conflict of interest.

References

Chu HJ, Gao LX, Zhang WD, 2016. Distributed adaptive containment control of heterogeneous linear multi-agent systems: an output regulation approach. *IET Contr Theory Appl*, 10(1):95-102. https://doi.org/10.1049/iet-cta.2015.0398

Dimarogonas DV, Frazzoli E, Johansson KH, 2012. Distributed event-triggered control for multi-agent systems. *IEEE Trans Autom Contr*, 57(5):1291-1297. https://doi.org/10.1109/TAC.2011.2174666

Dolk V, Heemels M, 2017. Event-triggered control systems under packet losses. *Automatica*, 80:143-155. https://doi.org/10.1016/j.automatica.2017.02.029

Fan Y, Feng G, Wang Y, 2011. Combination framework of rendezvous algorithm for multi-agent systems with limited sensing ranges. *Asian J Contr*, 13(2):283-294. https://doi.org/10.1002/asjc.178

Fu JJ, Wang JZ, 2016. Robust finite-time containment control of general linear multi-agent systems under directed communication graphs. *J Franklin Inst*, 353(12):2670-2689. https://doi.org/10.1016/j.jfranklin.2016.05.015

Guo WL, 2016. Leader-following consensus of the second-order multi-agent systems under directed topology. *ISA Trans*, 65:116-124. https://doi.org/10.1016/j.isatra.2016.07.011

Haghshenas H, Badamchizadeh MA, Baradarannia M, 2015. Containment control of heterogeneous linear multi-agent systems. *Automatica*, 54:210-216. https://doi.org/10.1016/j.automatica.2015.02.002

Hardy GH, Littlewood JE, Pólya G, 1952. Inequalities. Cambridge University Press, UK.

He XY, Wang QY, Yu WW, 2015. Distributed finite-time containment control for second-order nonlinear multi-agent systems. *Appl Math Comput*, 268:509-521. https://doi.org/10.1016/j.amc.2015.06.101

Hu AH, Cao JD, Hu MF, et al., 2015. Event-triggered consensus of multi-agent systems with noises. *J Franklin Inst*, 352(9):3489-3503. https://doi.org/10.1016/j.jfranklin.2014.08.005

Ji M, Ferrari-Trecate G, Egerstedt M, et al., 2008. Containment control in mobile networks. *IEEE Trans Autom Contr*, 53(8):1972-1975. https://doi.org/10.1109/TAC.2008.930098

Li HL, Yang RN, Li QN, 2017. Designing a novel consensus protocol for multiagent systems with general dynamics under directed networks. *Front Inform Technol Electron Eng*, 18(8):1071-1081. https://doi.org/10.1631/FITEE.1601422

Li JZ, Li CL, Yang XF, et al., 2018. Event-triggered containment control of multi-agent systems with high-order dynamics and input delay. *Electronics*, 7(12):343. https://doi.org/10.3390/electronics7120343

Li WQ, Liu L, Feng G, 2016. Distributed containment tracking of multiple stochastic nonlinear systems. *Automatica*, 69:214-221. https://doi.org/10.1016/j.automatica.2016.02.021

Li WX, Chen ZQ, Liu ZX, 2014. Formation control for nonlinear multi-agent systems by robust output regulation. *Neurocomputing*, 140:114-120. https://doi.org/10.1016/j.neucom.2014.03.030

Li YF, Hua CC, Wu SS, et al., 2017. Output feedback distributed containment control for high-order nonlinear multiagent systems. *IEEE Trans Cybern*, 47(8):2032-2043. https://doi.org/10.1109/TCYB.2017.2655054

Lin X, Zheng YS, 2017. Finite-time consensus of switched multiagent systems. *IEEE Trans Syst Man Cybern Syst*, 47(7):1535-1545. https://doi.org/10.1109/TSMC.2016.2631659

Liu KE, Xie GM, Wang L, 2014. Containment control for second-order multi-agent systems with time-varying delays. *Syst Contr Lett*, 67:24-31. https://doi.org/10.1016/j.sysconle.2013.12.013

Liu YF, Geng ZY, 2015. Finite-time formation control for linear multi-agent systems: a motion planning approach. *Syst Contr Lett*, 85:54-60. https://doi.org/10.1016/j.sysconle.2015.08.009

Liu ZX, Jin QT, Chen ZQ, 2015. Distributed containment control for bounded unknown second-order nonlinear multi-agent systems with dynamic leaders. *Neurocomputing*, 168:1138-1143. https://doi.org/10.1016/j.neucom.2015.05.009

Lu MB, Liu L, 2019. Leader-following consensus of second-order nonlinear multi-agent systems subject to disturbances. *Front Inform Technol Electron Eng*, 20(1):88-94. https://doi.org/10.1631/FITEE.1800611

Ma CQ, Zhang JF, 2012. On formability of linear continuous-time multi-agent systems. *J Syst Sci Compl*, 25(1):13-29. https://doi.org/10.1007/s11424-012-0108-3

Ma Q, Miao GY, 2014. Distributed containment control of linear multi-agent systems. *Neurocomputing*, 133:399-403. https://doi.org/10.1016/j.neucom.2013.12.034

Meng ZY, Ren W, You Z, 2010. Distributed finite-time attitude containment control for multiple rigid bodies. *Automatica*, 46(12):2092-2099. https://doi.org/10.1016/j.automatica.2010.09.005

Miao GY, Cao JD, Alsaedi A, et al., 2017. Event-triggered containment control for multi-agent systems with constant time delays. *J Franklin Inst*, 354(15):6956-6977. https://doi.org/10.1016/j.jfranklin.2017.08.010

Qin JH, Yu CB, Anderson BDO, 2016. On leaderless and leader-following consensus for interacting clusters of second-order multi-agent systems. *Automatica*, 74:214-221. https://doi.org/10.1016/j.automatica.2016.07.008

Qin JH, Zheng WX, Gao HJ, et al., 2017. Containment control for second-order multiagent systems communicating over heterogeneous networks. *IEEE Trans Neur Netw Learn Syst*, 28(9):2143-2155. https://doi.org/10.1109/TNNLS.2016.2574830

Rehan M, Jameel A, Ahn CK, 2018. Distributed consensus control of one-sided Lipschitz nonlinear multiagent systems. *IEEE Trans Syst Man Cybern Syst*, 48(8):1297-1308. https://doi.org/10.1109/TSMC.2017.2667701

Tabuada P, 2007. Event-triggered real-time scheduling of stabilizing control tasks. *IEEE Trans Autom Contr*, 52(9):1680-1685. https://doi.org/10.1109/TAC.2007.904277

Wang P, Jia Y, 2015. Robust H_∞ containment control for second-order multi-agent systems with nonlinear dynamics in directed networks. *Neurocomputing*, 153:235-241. https://doi.org/10.1016/j.neucom.2014.11.031

Wang W, Wang D, Peng Z, 2015. Distributed containment control for uncertain nonlinear multi-agent systems in non-affine pure-feedback form under switching topologies. *Neurocomputing*, 152:1-10. https://doi.org/10.1016/j.neucom.2014.11.035

Wang ZD, Ding DR, Dong HL, et al., 2013. H_∞ consensus control for multi-agent systems with missing measurements: the finite-horizon case. *Syst Contr Lett*, 62(10):827-836. https://doi.org/10.1016/j.sysconle.2013.06.004

Wei B, Xiao F, 2016. Event-triggered control for synchronization of coupled harmonic oscillators. *Syst Contr Lett*, 97:163-168. https://doi.org/10.1016/j.sysconle.2016.09.006

Wu Z, Xu Y, Lu R, et al., 2018a. Event-triggered control for consensus of multiagent systems with fixed/switching topologies. *IEEE Trans Syst Man Cybern Syst*, 48(10):1736-1746. https://doi.org/10.1109/TSMC.2017.2744671

Wu Z, Xu Y, Pan Y, et al., 2018b. Event-triggered control for consensus problem in multi-agent systems with quantized relative state measurements and external disturbance. *IEEE Trans Circ Syst I*, 65(7):2232-2242. https://doi.org/10.1109/TCSI.2017.2777504

Xia H, Zheng WX, Shao JL, 2018. Event-triggered containment control for second-order multi-agent systems with sampled position data. *ISA Trans*, 73:91-99. https://doi.org/10.1016/j.isatra.2017.11.001

Xu HH, Zhu J, 2011. An iterative approach to Bayes risk decoding and system combination. *J Zhejiang Univ-Sci C (Comput & Electron)*, 12(3):204-212. https://doi.org/10.1631/jzus.C1000045

Yan HC, Shen YC, Zhang H, et al., 2014. Decentralized event-triggered consensus control for second-order multi-agent systems. *Neurocomputing*, 133:18-24. https://doi.org/10.1016/j.neucom.2013.11.036

Yoo SJ, 2013. Distributed adaptive containment control of uncertain nonlinear multi-agent systems in strict-feedback form. *Automatica*, 49(7):2145-2153. https://doi.org/10.1016/j.automatica.2013.03.007

Zhang T, Ge SS, Hang CC, 2000. Adaptive neural network control for strict-feedback nonlinear systems using backstepping design. *Automatica*, 36(12):1835-1846. https://doi.org/10.1016/S0005-1098(00)00116-3

Zhang WB, Tang Y, Liu YR, et al., 2017. Event-triggering containment control for a class of multi-agent networks with fixed and switching topologies. *IEEE Trans Circ Syst I*, 64(3):619-629. https://doi.org/10.1109/TCSI.2016.2618944

Zheng YS, Ma JY, Wang L, 2018. Consensus of hybrid multi-agent systems. *IEEE Trans Neur Netw Learn Syst*, 29(4):1359-1365. https://doi.org/10.1109/TNNLS.2017.2651402

Zhou F, Wang ZJ, 2015. Containment control of linear multi-agent systems with directed graphs and multiple leaders of time-varying bounded inputs. *IET Contr Theory Appl*, 9(16):2466-2473. https://doi.org/10.1049/iet-cta.2014.0634

Zou WC, Xiang ZR, 2017. Event-triggered distributed containment control of heterogeneous linear multi-agent systems by an output regulation approach. *Int J Syst Sci*, 48(10):2041-2054. https://doi.org/10.1080/00207721.2017.1309595

Zou WC, Xiang ZR, 2019a. Event-triggered containment control of second-order nonlinear multi-agent systems. *J Franklin Inst*, 356(17):10421-10438. https://doi.org/10.1016/j.jfranklin.2018.05.060

Zou WC, Xiang ZR, 2019b. Event-triggered leader-following consensus of non-linear multi-agent systems with switched dynamics. *IET Contr Theory Appl*, 13(9):1222-1228. https://doi.org/10.1049/iet-cta.2018.5126