Int. J. Appl. Math. Comput. Sci., 2021, Vol. 31, No. 4, 635–645
DOI: 10.34768/amcs-2021-0044

NEURO–ADAPTIVE COOPERATIVE CONTROL FOR HIGH–ORDER NONLINEAR MULTI–AGENT SYSTEMS WITH UNCERTAINTIES

CHENG PENG [a], ANGUO ZHANG [b], JUNYU LI [a,*]

[a] School of Electrical and Mechanical Engineering
Hefei Technology College
Hefei 230012, China
e-mail: 1152475543@qq.com

[b] College of Physics and Information Engineering
Fuzhou University
Fuzhou 350018, China

The consensus problem for a class of high-order nonlinear multi-agent systems (MASs) with external disturbance and system uncertainty is studied. We design an online-update radial basis function (RBF) neural network based distributed adaptive control protocol, where the sliding model control method is also applied to eliminate the influence of the external disturbance and system uncertainty. System consensus is verified by using the Lyapunov stability theorem, and sufficient conditions for cooperative uniform ultimately boundedness (CUUB) are also derived. Two simulation examples demonstrate the effectiveness of the proposed method for both homogeneous and heterogeneous MASs.

Keywords: multi-agent systems, RBF neural network, sliding mode control, cooperative control.

1. Introduction

The problem of leader-following consensus in multi-agent systems (MASs) refers to one or several agents in the system acting as the leader, and the rest being follower agents, where the leader's dynamic behavior is not affected by other nodes. The control goal is to design a distributed protocol based on neighbor information for the follower agents, so that all agents can track the leader's dynamic behavior asymptotically. The leader-following topology is an energy-saving mechanism that exists in many biological systems, and it can strengthen group communication, robustness, flexibility and dispersion (Low, 2000; Parrish *et al*., 2002; Couzin *et al*., 2002; Zhao *et al*., 2021). Therefore, multi-agent systems with a leader-following topology have received increasing attention.

Farrera *et al*. (2020) proposed distributed proportional-integral observer based fault estimation of leader-following linear MASs with actuator faults. In the works of Bechlioulis and Rovithakis (2017) as well as Yang and Li (2020) a robust control based on an observer was designed for a class of high-order nonlinear MASs that satisfies the Lipschitz condition to solve the problem of state enclosing control of MASs. Aryankia and Selmic (2021) proposed an adaptive neural network based backstepping controller that uses rigid graph theory to address the distance based formation control problem and target tracking for nonlinear MASs with bounded time delay and disturbance. A radial basis function (RBF) neural network is used to overcome and compensate for the unknown non-linearity and disturbance in the system dynamics. Qin *et al*. (2019) investigated a class of nonaffine nonlinear MASs with actuator faults of partial loss of effectiveness and a biased fault. A neural network based adaptive consensus protocol is developed, where the neuron input uses both the state and consensus error information. Ni *et al*. (2017) designed a type of sliding mode observer under the condition of input delay, which can send leader information to followers within a limited time.

Bartoszewicz and Adamiak (2019) presented a reference trajectory based sliding mode control strategy

*Corresponding author

for disturbed discrete time dynamical systems. Meng *et al.* (2020) proposed an adaptive neural distributed synchronization scheme with guaranteed performance, which is used to deal with the synchronization control problem in the leader–follower format of a class of high-order nonaffine nonlinear MASs under a directed communication protocol. For linear systems, Ma and Miao (2015) proposed a solution that the leader follows the heterogeneous MAS in the network to achieve a consistent output. When the state information is not easy to measure directly, a solution based on a dynamic regulator and a state observer is used to reconstruct the state. For nonlinear systems, Liu and Jia (2008) aim at a second-order nonlinear system, use a universal approximation function in the form of a of neural network to estimate and approximate unknown functions online, and obtain an adaptive protocol based on neural networks. Shen *et al.* (2020) studied the cooperative control problem with the followers of unmodeled dynamics. The authors also proposed a fully neural network based adaptive control strategy. Lu *et al.* (2021) introduced an adaptive neural control approach to the leader–follower consensus control problem of uncertain MASs.

This paper studies the cooperative control of high-order nonlinear MASs. The motivation is is related to the following two aspects:

First, existing research on MASs mostly focuses on first-order and second-order systems (Zhang *et al.*, 2018; Wu *et al.*, 2018; Li *et al.*, 2018; Duan *et al.*, 2020; Zou *et al.*, 2020; Wang *et al.*, 2020; Yao *et al.*, 2020; Fu *et al.*, 2020). However, in engineering practice, single-link flexible-joint manipulators (Zhang, 2008; Ling *et al.*, 2019), robot formation cooperation (Miao *et al.*, 2018; Zong *et al.*, 2019), and synchronous generator coordination (Fathi *et al.*, 2018; Abdelrahem *et al.*, 2018; Nian and Jiao, 2020) are based on high-order dynamic modeling. Therefore, studying such high-order nonlinear systems not only takes an important theoretical value, but also has a strong engineering practical value (Huang *et al.*, 2015).

Second, due to the interference of the external environment and the uncertainty of its own system parameters, the control objects often have unknown and complex nonlinear dynamics, which makes it difficult to obtain an accurate mathematical model of the control system. The existence of the nonlinear dynamics may transform the system from homogeneous to heterogeneous, which brings about difficulties in the cooperative control of nonlinear systems (Cui *et al.*, 2016).

In view of the above problems, this paper studies the Brunovsky-type high-order nonlinear cooperative control for leader-following MASs. The characteristic of the dynamic system is that each follower node couples the unknown nonlinear dynamics and external disturbance through a high-order integrator, and the dynamics of each node can be completely different (Zhang and Lewis, 2012). The leader agent is a high-order non-autonomous nonlinear system, and its dynamics are unknown to all following agents. This paper designs a distributed adaptive radial basis function (RBF) neural network control algorithm, to ensure that the neural network approximates nonlinear terms online, and eliminate the influence of uncertain items such as continuous bounded disturbance on stability. Finally, an adaptive protocol based on an adaptive RBF neural network is proposed. This method can solve the problem of tracking consensus in high-order nonlinear MASs with uncertainty and ensure the final tracking error.

This paper presents a consensus protocol for high-order nonlinear MASs with uncertainty under the condition of weak connectivity, and provides the corresponding theoretical rationale. The correctness and effectiveness of the proposed method are verified by numerical simulations.

2. Preliminaries and problem statement

2.1. Graph theory and notation. Based on the principles of graph theory, a graph $G = (V, E, A)$ is used to describe the communication topology composed of multi-agents, which represents the information interaction between agents. The vertex $V = \{v_1, v_2, \dots, v_N\}$ is the set of N agents; $E \in V \times V$ represents the directed edge set of information interaction between agents; $\boldsymbol{A} = [a_{ij}]^{N\times N}$ is the weight matrix associated with the information topology; a_{ij} represents the weight of the edge (v_j, v_i). For a directed graph, if $(v_j, v_i) \in E$, then $a_{ij} > 0$, otherwise, $a_{ij} = 0$; the corresponding in-degree matrix and the Laplacian matrix of the graph are respectively defined as $\boldsymbol{D} = \operatorname{diag}\{\deg_1, \deg_2, \dots, \deg_N\}$ and $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}$, where $\deg_i = \sum_{j=1}^{N} a_{ij}$ is the in-degree of vertex i.

This paper uses the following notation: $\mathbb{R}^n$ represents the set of n-dimensional vectors of real numbers; $\mathbf{1}_n$ represents the set of n-dimensional column vectors of ones; $\mathbf{I}_n$ represents an $n \times n$-dimensional identity matrix; $\|\cdot\|$ represents the Euclidean norm of a vector; $\operatorname{diag}\{m_1, m_2, \dots, m_n\}$ is the diagonal matrix with diagonal elements of $m_1, m_2, \dots, m_n$; $\underline{\sigma}(P)$ and $\bar{\sigma}(P)$ denote the smallest and largest singular values of matrix P, respectively; $\operatorname{tr}\{\cdot\}$ denotes the trace of a matrix. $\|\cdot\|_1$ is the 1-norm, and $\|\cdot\|_F$ is the Frobenius norm. Given a matrix $\boldsymbol{A} = [a_{ij}]_{m\times n} \in \mathbb{R}^{m\times n}$, the 1-norm is

$$\|\boldsymbol{A}\|_1 = \max_{1\le j\le n} \sum_{i=1}^{m} |a_{ij}|,$$

and the Frobenius norm is defined as the sum of the squares of the absolute values of the elements of a matrix, that is,

$$\|\boldsymbol{A}\|_F = \sqrt{\mathrm{tr}\big(\boldsymbol{A}^T\boldsymbol{A}\big)} = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2}.$$

2.2. Problem statement. Consider a system composed of $N+1$ agents, where the i-th agent has the following Brunovsky nonlinear dynamic model (Arandabricaire *et al.*, 1995):

$$\begin{cases} \dot{x}_{i,m}(t) = x_{i,m+1}(t), \quad m = 1,2,\dots,M-1, \\ \dot{x}_{i,M}(t) = f_i(t,x_i) + d_i(t) + u_i(t), \quad m = M, \end{cases} \tag{1}$$

where $i = 1,2,\dots,N$; $x_{i,m}(t) \in \mathbb{R}$ represents the m-th state of the i-th agent at time t; $x_i(t) = [x_{i,1}(t), x_{i,2}(t), \dots, x_{i,M}(t)]^{\mathrm{T}} \in \mathbb{R}^M$ represents the state vector of the i-th agent; $d_i(t) \in \mathbb{R}$ represents the uncertainty item of the i-th agent (including the external disturbance or unmodeled dynamics); $u_i(t) \in \mathbb{R}$ represents the control input variable to the i-th agent at time t; $f_i(t,x_i)$ is a continuous function, which represents the inherent nonlinear dynamic behavior of the i-th agent.

The dynamics of the MAS leader (marked with label 0) can be described as

$$\begin{cases} \dot{x}_{0,m}(t) = x_{0,m+1}(t), \quad m = 1,2,\dots,M-1, \\ \dot{x}_{0,M}(t) = f_0(t,x_0), \end{cases} \tag{2}$$

where $x_{0,m}(t) \in \mathbb{R}$ represents the m-th order state of the leader at time t. Here $x_0(t) = [x_{0,1}(t), x_{0,2}(t), \dots, x_{0,M}(t)]^{\mathrm{T}} \in \mathbb{R}^M$ represents the state vector, and $f_0(t,x_i)$ is a continuous function that represents the inherent nonlinear dynamics of the leader.

The m-th order consensus error of agent i is denoted by $\delta_{i,m} = x_{i,m} - x_{0,m}$. Write $\boldsymbol{\delta}_m = \big[\delta_{1,m}, \delta_{2,m}, \dots, \delta_{N,m}\big]^T$; thus we have $\boldsymbol{\delta}_m = \boldsymbol{x}_m - \mathbf{1}_N x_{0,m}$, where $\boldsymbol{x}_m = \big[x_{1,m}, x_{2,m}, \dots, x_{N,m}\big]^{\mathrm{T}} \in \mathbb{R}^N$, $m = 1,2,\dots,M$. The goal of designing a distributed controller in this paper is to approach the zero consensus error and improve the follower tracking performance.

Definition 1. (*Cooperative uniformly ultimate boundedness*) For any $m (m = 1,2,\dots,M)$, if there is a compact set $\Omega_m \subset \mathbb{R}^N$ satisfying the following three conditions, it is said that the high-order traking error δ_m of the followers is *cooperatively uniformly ultimately bounded* (CUUB):

(i) $\{0\} \subset \Omega_m$,

(ii) $\delta_m(t_0) \subset \Omega_m$,

(iii) there exists an upper bound Δ_m and a time T_m, such that $\|\delta_m\| \le \Delta_m$ if $\forall t \ge t_0 + T_m$.

For agent i, when $t \ge t_0 + T_m$, if the tracking error is CUUB, the follower state $x_{i,m}(t)$ converges to the neighborhood state $x_{0,m}(t)$ of the leader.

The local neighborhood error of the i-th agent is defined as

$$\begin{aligned} e_{i,m} &= \sum_{j \in N_i} a_{ij}(x_{j,m} - x_{i,m}) + b_i(x_{0,m} - x_{i,m}) \\ &= \sum_{j=1}^{N} L_{ij} x_{j,m} + b_i(x_{0,m} - x_{i,m}), \end{aligned} \tag{3}$$

where $m = 1,2,\dots,M$. If there is a directed edge (v_0, v_i) between the follower agent i and the leader agent 0, the follower can "perceive" the information of the leader agent, then this weight of this edge is $b_i > 0$, otherwise $b_i = 0$. Define the adjacency matrix $\boldsymbol{B} = \mathrm{diag}\{b_1, b_2, \dots, b_N\}$, and let the m-th order global neighborhood error be $\boldsymbol{e}_m(t) = [e_{1,m}, e_{2,m}, \dots, e_{N,m}]^{\mathrm{T}} \in \mathbb{R}^N$. Let $\boldsymbol{f}(t,x) = [f_1(t,x_1), f_2(t,x_2), \dots, f_N(t,x_N)]^{\mathrm{T}} \in \mathbb{R}^N$, $\boldsymbol{u}(t) = [u_1(t), u_2(t), \dots, u_N(t)]^{\mathrm{T}} \in \mathbb{R}^N$, $\boldsymbol{d}(t) = [d_1(t), d_2(t), \dots, d_N(t)]^{\mathrm{T}} \in \mathbb{R}^N$. Thus, the derivative of (3) can be obtained as

$$\begin{cases} \dot{e}_{i,m}(t) = e_{i,m+1}(t), \quad m = 1,2,\dots,M-1, \\ \dot{e}_{i,M}(t) = -(\boldsymbol{L}+\boldsymbol{B})(\boldsymbol{f} + \boldsymbol{d} + \boldsymbol{u} - f_0\mathbf{1}_N). \end{cases} \tag{4}$$

Define the extended directed network graph composed of the leader and the follower agents as $\bar{G} = (\bar{V}, \bar{E}, \bar{A})$, to realize the information interaction between the network G and the leader agent 0. The tracking consensus needs to make the following assumptions about the network topology.

Assumption 1. There exists a directed spanning tree with leader 0 as the root node in the extended network $\bar{G}$.

Lemma 1. (Das and Lewis, 2010; Cui *et al.*, 2016) *Define*

$$\boldsymbol{q} = [q_1, q_2, \dots, q_N]^T = (\boldsymbol{L}+\boldsymbol{B})^{-1}\mathbf{1}_N,$$

$$\boldsymbol{P} = \mathrm{diag}\{p_i\} = \mathrm{diag}\Big\{\frac{1}{q_i}\Big\}$$

$$\boldsymbol{Q} = \boldsymbol{P}(\boldsymbol{L}+\boldsymbol{B}) + (\boldsymbol{L}+\boldsymbol{B})^T\boldsymbol{P}.$$

Thus both $\boldsymbol{P}$ and $\boldsymbol{Q}$ are positive definite matrices. In view of Assumption 1, in the extended network $\bar{G}$, $\boldsymbol{L}+\boldsymbol{B}$ is a non singular matrix.

Lemma 2. *We have*

$$\|\boldsymbol{\delta}_m\| \le \frac{\|\boldsymbol{e}_m\|}{\underline{\boldsymbol{\sigma}}(\boldsymbol{L}+\boldsymbol{B})}, \quad m = 1,2,\dots,M.$$

Proof. According to (3), define the global error vector

$$\begin{aligned} \boldsymbol{\delta} &= \big[\boldsymbol{\delta}_1, \boldsymbol{\delta}_2, \dots, \boldsymbol{\delta}_M\big]^{\mathrm{T}} \\ &= \big[\boldsymbol{x}_1 - \mathbf{1}_N x_{0,1}, \boldsymbol{x}_2 - \mathbf{1}_N x_{0,2}, \dots, \boldsymbol{x}_M - \mathbf{1}_N x_{0,M}\big]^{\mathrm{T}}, \end{aligned}$$

where $\boldsymbol{x}_m = [x_{1,m}, x_{2,m}, \dots, x_{N,m}]^{\mathrm{T}} \in \mathbb{R}^N$, for $m = 1, 2, \dots, M$.

Then the global error vector of graph $\bar{G}$ can be given by

$$\begin{cases} \boldsymbol{e}_1 = -(\boldsymbol{L}+\boldsymbol{B})(\boldsymbol{x}_1 - \boldsymbol{1}_N x_{0,1}) = -(\boldsymbol{L}+\boldsymbol{B})\boldsymbol{\delta}_1, \\ \vdots \\ \boldsymbol{e}_m = -(\boldsymbol{L}+\boldsymbol{B})(\boldsymbol{x}_m - \boldsymbol{1}_N \boldsymbol{x}_{0,m}) = -(\boldsymbol{L}+\boldsymbol{B})\boldsymbol{\delta}_m. \end{cases} \tag{5}$$

According to Assumption 1, the matrix $\boldsymbol{L}+\boldsymbol{B}$ is non singular. Since $\boldsymbol{e}_m = -(\boldsymbol{L}+\boldsymbol{B})\boldsymbol{\delta}_m$ in (5), we get

$$\boldsymbol{\delta}_m = -(\boldsymbol{L}+\boldsymbol{B})^{-1}\boldsymbol{e}_m. \tag{6}$$

Thus

$$\|\boldsymbol{\delta}_m\| = \left\|(\boldsymbol{L}+\boldsymbol{B})^{-1}\boldsymbol{e}_m\right\| \le \frac{\|\boldsymbol{e}_m\|}{\underline{\sigma}(\boldsymbol{L}+\boldsymbol{B})}. \tag{7}$$

■

3. Design of an adaptive cooperative tracking controller

For MASs with uncertainty, the use of the offline training of neural networks is obviously inappropriate. In order to solve this problem, a distributed neural network controller is designed. An online adaptive RBF neural network control method is adopted to realize the adaptive adjustment of the neural network weight matrix to solve the problem of consensus.

3.1. Sliding mode surface function design. The sliding mode variable structure control is adopted mainly because the control algorithm has the characteristics of rapid response while ensuring the system stability.

The sliding mode surface function of agent i ($i \in N$) is defined as

$$s_i = \alpha_1 e_{i,1} + \alpha_2 e_{i,2} + \dots + \alpha_{M-1} e_{i,M-1} + e_{i,M}, \tag{8}$$

where the $(M-1)$-th-order polynomial with coefficients $\alpha_1, \alpha_2, \dots, \alpha_{M-1}$ is Hurwitz. Then if s_i is bounded, $\boldsymbol{e}_i$ is also bounded. Further, $\boldsymbol{e}_i \to 0$ as $s_i \to 0$.

Thus, the global error vector of the sliding mode surface function is given by $\boldsymbol{s} = [s_1, s_2, \dots, s_N]^{\mathrm{T}} = \alpha_1 \boldsymbol{e}_1 + \dots + \alpha_{M-1}\boldsymbol{e}_{M-1} + \boldsymbol{e}_M$.

Define

$$\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \dots, \alpha_{M-1}]^T,$$

$$\boldsymbol{E}_1 = \left[\boldsymbol{e}_1, \boldsymbol{e}_2, \dots, \boldsymbol{e}_{M-1}\right] \in \mathbb{R}^{N\times M-1},$$

$$\boldsymbol{E}_2 = \dot{\boldsymbol{E}}_1 = \left[\boldsymbol{e}_2, \boldsymbol{e}_3, \dots, \boldsymbol{e}_M\right] \in \mathbb{R}^{N\times M-1},$$

$$\boldsymbol{I} = \left[0, 0, \dots, 1\right]^{\mathrm{T}} \in \mathbb{R}^{M-1},$$

$$\boldsymbol{\Gamma} = \begin{pmatrix} 0 & \boldsymbol{I} \\ -\alpha_1 & (-\alpha_2 - \dots - \alpha_{M-1}) \end{pmatrix}. \tag{9}$$

Thus we have

$$\boldsymbol{E}_2 = \boldsymbol{E}_1\boldsymbol{\Gamma}^{\mathrm{T}} + \boldsymbol{s}\boldsymbol{I}^{\mathrm{T}}. \tag{10}$$

Since $\boldsymbol{\Gamma}$ is a Hurwitz matrix, given any positive number β, there exists a symmetric matrix $\boldsymbol{P}_1 > 0$, which makes the Lyapunov function hold, i.e., the condition on

$$\boldsymbol{\Gamma}^{\mathrm{T}}\boldsymbol{P}_1 + \boldsymbol{P}_1\boldsymbol{\Gamma} = -\beta\boldsymbol{I}_N, \tag{11}$$

the derivative of the dynamic sliding mode error $\boldsymbol{s}$ is

$$\dot{\boldsymbol{s}} = \boldsymbol{\rho} - \big(\boldsymbol{L}+\boldsymbol{B}\big)\big(\boldsymbol{f}+\boldsymbol{d}+\boldsymbol{u}-f_0\boldsymbol{1}_N\big), \tag{12}$$

where $\boldsymbol{\rho} = \alpha_1\boldsymbol{e}_2 + \alpha_2\boldsymbol{e}_3 + \dots + \alpha_{M-1}\boldsymbol{e}_M = \boldsymbol{E}_2\boldsymbol{\alpha}$.

Lemma 3. *For agent i ($i = 1, 2, \dots, N$), assume that*

$$\begin{aligned} |s_i(t)| &\le \psi_i, \quad \forall t \ge t_0, \\ |s_i(t)| &\le \theta_i, \quad \forall t \ge T_{\theta_i}, \end{aligned} \tag{13}$$

where time $T_{\xi_i} > t_0$, t_0 is the initial time, the upper bound $\psi_i > 0, \xi_i > 0$. There exists a time $T_{\theta_i} > t_0$, $\theta_i > 0$, which yields

$$\begin{aligned} \|\boldsymbol{e}_i(t)\| &\le \psi_i, \quad \forall t \ge t_0, \\ \|\boldsymbol{e}_i(t)\| &\le \theta_i, \quad \forall t \ge T_{\theta_i}. \end{aligned} \tag{14}$$

Proof. Let $\boldsymbol{e}_i(t) = \left[e_{i,1}, e_{i,1}, \dots, e_{i,M-1}\right]^{\mathrm{T}} \in \mathbb{R}^{M-1}$. According to (8), we have

$$\dot{\boldsymbol{e}}_i = \boldsymbol{\Gamma}\boldsymbol{e}_i + \boldsymbol{I}s_i. \tag{15}$$

By (15), we get

$$\boldsymbol{e}_i(t) = \mathrm{e}^{\boldsymbol{\Gamma}(t-t_0)}\boldsymbol{e}_i(t_0) + \int_{t_0}^{t} \mathrm{e}^{\boldsymbol{\Gamma}(t-\tau)}\boldsymbol{I}s_i(\tau)\,\mathrm{d}\tau \tag{16}$$

Since $\boldsymbol{\Gamma}$ is a Hurwitz matrix, there exists $\phi > 0, \lambda > 0$, such that

$$\|\mathrm{e}^{\boldsymbol{\Gamma}(t-t_0)}\| \le \phi\mathrm{e}^{-\lambda(t-t_0)}. \tag{17}$$

From (16), it follows that

$$\begin{aligned} \|\boldsymbol{e}_i(t)\| &\le \phi\mathrm{e}^{-\lambda(t-t_0)}\|\boldsymbol{e}_i(t_0)\| \\ &\quad + \int_{t_0}^{t} \phi\mathrm{e}^{-\lambda(t-\tau)}\|\boldsymbol{I}\||s_i(\tau)|\,\mathrm{d}\tau \\ &\le \phi\mathrm{e}^{-\lambda(t-t_0)}\|\boldsymbol{e}_i(t_0)\| + \phi\frac{\|\boldsymbol{B}\|}{\lambda}\sup_{t_1<\tau<t}|s_i(\tau)| \\ &= \phi\mathrm{e}^{-\lambda(t-t_0)}\|\boldsymbol{e}_i(t_0)\| + \frac{\phi}{\lambda}\sup_{t_1<\tau<t}|s_i(\tau)| \end{aligned} \tag{18}$$

It can be seen from (18) that if $s_i(t)$ is bounded, then $\|\boldsymbol{e}_i(t)\| < \infty$, so that, for all $m = 1, 2, \ldots, M$, $e_{i,m}(t)$ is bounded. Moreover,

$$\begin{aligned} e_{i,M}(t) &= s_i - \alpha_1 e_{i,1}(t) - \alpha_2 e_{i,2}(t) - \cdots \\ &\quad - \alpha_{M-1} e_{i,M-1}(t) \end{aligned}$$

is also bounded. Therefore, $\|\boldsymbol{e}_i(t)\| < \infty$ if $s_i(t) < \infty$, i.e., $\|\boldsymbol{e}_i(t)\|$ is bounded.

Then it will be proved that if $s_i(t) \to 0$, $\|e_i\| \to 0$ holds. Since $s_i(\tau) \to 0$, for any given sufficiently small constant $\varepsilon_s > 0$, there is a time step t_1 such that when $\tau \geq t_1$, $(\phi/\lambda)|s_i(\tau)| \leq \varepsilon_s$ holds, and thus $(\phi/\lambda)\sup_{t_1 \leq \tau \leq t} \leq \varepsilon_s$. Similarly, from the exponential stability of $e^{-\lambda(t-t_1)}$ it can be obtained that, for any given sufficiently small constant $\varepsilon_e > 0$, there is a time step t_2 such that when $t - t_1 \geq t_2$, $\phi e^{-\lambda(t-t_1)} \parallel e_i(t_1) \parallel \leq \varepsilon_e$ holds. Substituting the variable t_0 of the inequality (18) by t_1, it can be obtained that when $t \geq t_1 + t_2$, we have

$$\begin{aligned} \|\boldsymbol{e}_i(t)\| &\leq \phi \mathrm{e}^{-\lambda(t-t_1)} \|\boldsymbol{e}_i(t_1)\| + \frac{\phi}{\lambda} \sup_{t_1 \leq \tau \leq t} |s_i(\tau)| \\ &\leq \phi \mathrm{e}^{-\lambda(t-t_1)} \|\boldsymbol{e}_i(t_1)\| + \frac{\phi}{\lambda} \sup_{\tau \geq t_1} |s_i(\tau)| \\ &\leq \varepsilon_e + \varepsilon_s. \end{aligned} \tag{19}$$

According to the arbitrariness of ε_e and ε_s, when $t \to \infty$, $\|e_i(t)\| \to 0$ holds. Thus we get $e_{i,m}(t) \to 0$ for all $m = 1, 2, \ldots, M-1$. Therefore, $e_{i,M}(t) = s_i - \alpha_1 e_{i,1}(t) - \alpha_2 e_{i,2}(t) - \cdots - \alpha_{M-1} e_{i,M-1}(t) \to 0$ can be obtained by (8). In summary, as $s_i(t) \to 0$, $\boldsymbol{e}_i \to 0$ holds, and the Lemma 3 is proved.

■

3.2. Distributed control protocol design. While designing for each agent i a distributed control law u_i and a weight adaptation law of the RBF neural network, we shall make the following assumptions.

Assumption 2. There is a positive number $\bar{X} > 0$, such that the leader state satisfies $\|x_0(t)\| \leq \bar{X}$.

Assumption 3. There is a continuous function $g(\cdot) : \mathbb{R}^m \to \mathbb{R}$, satisfying $|f_0(t, x_0(t))| \leq |g(x_0(t))|$.

Assumption 4. The external disturbance $d_i(t)$ of each agent is unknown and bounded, namely $\sup\{\|d_1(t)\|; \|d_2(t)\|, \ldots, \|d_N(t)\|, \|d_0(t)\|\} \leq d$, d can be unknown.

It should be noted that Assumption 3 shows that the leader's nonlinear term $f_0(t, x_0(t))$ has an upper limit $F, \forall t \geq t_0$. Assuming there exits parameters $\bar{X}, \bar{F}$ and $\bar{d}$, the designer does not need to know them, that is, these bounds are not directly used for controller design, but are used for the Lyapunov method to analyze system stability.

This paper proposes a distributed neuro-adaptive (radial basis function, RBF neural network) control algorithm to ensure that the neural network approximates nonlinear terms online, and eliminate the influence of uncertain terms such as continuous bounded disturbances on stability.

The kernel function of the RBF neural network used in this paper is a Gaussian function,

$$\boldsymbol{\varphi}_j = \exp\left(\frac{\|\boldsymbol{x}_i - \boldsymbol{c}_j\|^2}{2\hat{b}_j^2}\right), \tag{20}$$

where $j = 1, 2, \ldots, z$, z is the number of nodes in the hidden layer, $\hat{b}_j$ is the variance.

The network output is calculated by

$$o_i(x_i) = \boldsymbol{W}_i^{\mathrm{T}} \boldsymbol{\varphi}_i(x_i) + \varepsilon_i,$$

where $\boldsymbol{W_i} = [w_{i,1}, w_{i,2}, \ldots, w_{i,z}]$, $\boldsymbol{\varphi}_i(x_i) = [\varphi_{i,1}(x_i), \varphi_{i,2}(x_i), \ldots, \varphi_{i,z}(x_i)]^{\mathrm{T}}$. Thus the global output vector can be formulated as

$$\boldsymbol{o}(x_i) = \boldsymbol{W}^{\mathrm{T}} \boldsymbol{\varphi}(x) + \boldsymbol{\varepsilon}. \tag{21}$$

According to the Stone–Weierstrass theorem (Joshi, 1983), given a compact set Ω, for any positive number ε_h, there exists a sufficiently large positive integer z^*, so that for any $z > z^*$, an ideal weight vector $\boldsymbol{W}^i$ and a suitable radial basis function vector φ_i can always be found such that $\|\varepsilon_i\| \leq \varepsilon_h$.

The system state $x_{i,1}(t), x_{i,2}(t), \ldots, x_{i,M}(t)$ is used as the input of the network thus the network actual output is

$$\hat{o}_i(x_i) = \hat{\boldsymbol{W}}_i^{\mathrm{T}}(t) \boldsymbol{\varphi}_i(x_i), \tag{22}$$

where $\hat{\boldsymbol{W}}_i(t) \in \mathbb{R}^z$ is the vector of estimated weigths of agent i. Define the global network output vector as

$$\hat{\boldsymbol{o}}(x) = \hat{\boldsymbol{W}}^{\mathrm{T}} \boldsymbol{\varphi}(x) \tag{23}$$

with the approxmiation error $\boldsymbol{\varepsilon} = [\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_N]$.

Define the maximal value of the Gaussian function output $\bar{\varphi}_i = \max_{x_i \in \Omega} \|\boldsymbol{\varphi}_i(x_i)\|$, and the maximal value of the ideal weight $\bar{W}_i = \max \|\boldsymbol{W}_i\|$. Thus there exist positive numbers $\bar{\varphi}$, $\bar{W}$ and $\bar{\varepsilon}$, satisfying $\|\boldsymbol{\varphi}\| \leq \bar{\varphi}$, $\|\boldsymbol{W}\| \leq \bar{W}$ and $\|\boldsymbol{\varepsilon}\| \leq \bar{\varepsilon}$.

Design the weight adaptation law of the RBF neural network as

$$\dot{\hat{\boldsymbol{W}}}_i = -\boldsymbol{F}_i \boldsymbol{\varphi}_i \boldsymbol{s}_i p_i (\deg_i + b_i) - \zeta \boldsymbol{F}_i \hat{\boldsymbol{W}}_i. \tag{24}$$

It can be written in the following concise vector form:

$$\dot{\hat{\boldsymbol{W}}} = -\boldsymbol{F} \boldsymbol{\varphi} \boldsymbol{s} \boldsymbol{P} (\boldsymbol{D} + \boldsymbol{B}) - \zeta \boldsymbol{F} \hat{\boldsymbol{W}}, \tag{25}$$

where $\boldsymbol{F}_i = \boldsymbol{F}_i^T \in \mathbb{R}^{z \times z}$ is any positive definite matrix, the positive number ζ is an adjustable scalar, the matrix $\boldsymbol{P}$

Algorithm 1. Algorithm of online weight adaptation of the RBF neural network.

Input: Dynamic sliding mode error $\mathbf{s}$.
Output: Network output weights $\hat{\mathbf{W}}$.
Setting the scalar ζ, matrix $\mathbf{F}$, initialize matrix $\mathbf{P}$ according to Lemma 1.

for time t **in** $[1, 2, \ldots, T]$ **do**
  foreach output weight $\hat{\mathbf{W}}_i$ **do**
    Calculate the sliding mode error $\mathbf{s}$ according to Eqn. (12);
    Calculate the network activation φ_i according to Eqn. (20);
    Calculate the variation of $\dot{\hat{\mathbf{W}}}_{\mathbf{i}}$ by Eqn. (24);
    $\hat{\mathbf{W}}_i \leftarrow \dot{\hat{\mathbf{W}}}_{\mathbf{i}} + \hat{\mathbf{W}}_i$.
  end
end

has been defined in Lemma 1. A detailed description of the online adaptation is presented in Algorithm 1.

We propose the distributed control protocol for each agent i by

$$\begin{aligned} u_i =& \frac{1}{\deg_i + b_i}\Big(\alpha_1 e_{i,2} + \alpha_2 e_{i,3} + \cdots + \alpha_{M-1} e_{i,M}\Big) \\ & - \hat{o}_i(x_i) + k s_i. \end{aligned} \tag{26}$$

Thus the vector form for all the agents can be written as

$$\boldsymbol{u} = -\big(\boldsymbol{D} + \boldsymbol{B}\big)^{-1}\boldsymbol{\rho} - \hat{\boldsymbol{o}}(x) + k\boldsymbol{s}, \tag{27}$$

where the control gain satisfies

$$k > \frac{2}{\sigma(\boldsymbol{Q})}\Big(\frac{r^2}{\zeta} + \frac{2}{\beta}\boldsymbol{g}^2 + h\Big), \tag{28}$$

and

$$r = -\frac{1}{2}\bar{\varphi}\bar{\sigma}(\boldsymbol{P})\bar{\sigma}(\boldsymbol{A}),$$

$$g = -\frac{1}{2}\Big(\frac{\overline{\sigma}\left(\boldsymbol{P}\right)\overline{\sigma}\left(\boldsymbol{A}\right)}{\underline{\sigma}\left(\boldsymbol{D}+\boldsymbol{B}\right)}\left\|\boldsymbol{\Gamma}\right\|_F\left\|\overline{\boldsymbol{\alpha}}\right\| + \overline{\sigma}\left(\boldsymbol{P}_1\right)\Big),$$

$$h = \frac{\overline{\sigma}\left(\boldsymbol{P}\right)\overline{\sigma}\left(\boldsymbol{A}\right)}{\underline{\sigma}\left(\boldsymbol{D}+\boldsymbol{B}\right)}\left\|\overline{\alpha}\right\|,$$

$\boldsymbol{P}_1$ has been defined in (11), $\beta > 0$, $\boldsymbol{Q}$ has been explained in Lemma 1, and the coefficient ζ has been explained in (25).

4. Main results

Define $\tilde{\boldsymbol{W}}$ as the error between the ideal weight $\boldsymbol{W}$ and the estimated weights $\hat{\boldsymbol{W}}$ of the RBF neural network. Construct the Lyapunov function

$$\boldsymbol{V}(t) = \boldsymbol{V}_1(t) + \boldsymbol{V}_2(t) + \boldsymbol{V}_3(t), \tag{29}$$

where $\boldsymbol{P} = \boldsymbol{P}^{\mathrm{T}} > 0$; $\boldsymbol{F}^{-1} = \boldsymbol{F}^{-T} > 0$, and

$$\boldsymbol{V}_1(t) = \frac{1}{2}\boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{s}, \tag{30}$$

$$\boldsymbol{V}_2(t) = \frac{1}{2}\mathrm{tr}\left\{\tilde{\boldsymbol{W}}^{\mathrm{T}}\boldsymbol{F}^{-1}\tilde{\boldsymbol{W}}\right\}, \tag{31}$$

$$\boldsymbol{V}_3(t) = \frac{1}{2}\mathrm{tr}\left\{\boldsymbol{E}_1\boldsymbol{P}_1\boldsymbol{E}_1^{\mathrm{T}}\right\}. \tag{32}$$

We have the derivative of $\boldsymbol{V}_1(t)$,

$$\begin{aligned} \dot{\boldsymbol{V}}_1(t) &= \boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\dot{\boldsymbol{s}} \\ &= \boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\big[\boldsymbol{\rho} - (\boldsymbol{L}+\boldsymbol{B})(\boldsymbol{f}+\boldsymbol{d}+\boldsymbol{u}-f_0\mathbf{1}_N)\big]. \end{aligned} \tag{33}$$

Setting $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}$ and substituting (27) into (33), we have

$$\begin{aligned} \dot{\boldsymbol{V}}_1(t) &= \boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\big[\boldsymbol{\rho} - (\boldsymbol{L}+\boldsymbol{B})(\boldsymbol{f}+\boldsymbol{d} - \big(\boldsymbol{D}+\boldsymbol{B}\big)^{-1}\boldsymbol{\rho} \\ &\quad - \hat{\boldsymbol{o}}(x) + k\boldsymbol{s} - f_0\mathbf{1}_N)\big] \\ &= \boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{\rho} - \boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\big(\boldsymbol{L}+\boldsymbol{B}\big)\big(\boldsymbol{\varepsilon}+\boldsymbol{d}-f_0\mathbf{1}_N\big) \\ &\quad - k\boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\big(\boldsymbol{L}+\boldsymbol{B}\big)\boldsymbol{s} \\ &\quad - \boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\big[(\boldsymbol{D}+\boldsymbol{B})-\boldsymbol{A}\big]\big[(\boldsymbol{D}+\boldsymbol{B})^{-1}\boldsymbol{\rho} + \tilde{\boldsymbol{W}}^{\mathrm{T}}\boldsymbol{\varphi}\big] \\ &= \boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\big(\boldsymbol{L}+\boldsymbol{B}\big)\big(\boldsymbol{\varepsilon}+\boldsymbol{d}-f_0\mathbf{1}_N\big) \qquad (34) \\ &\quad - k\boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\big(\boldsymbol{L}+\boldsymbol{B}\big)\boldsymbol{s} \\ &\quad - \boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\big(\boldsymbol{D}+\boldsymbol{B}\big)\tilde{\boldsymbol{W}}^{\mathrm{T}}\boldsymbol{\varphi} + \boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{A}\tilde{\boldsymbol{W}}^{\mathrm{T}}\boldsymbol{\varphi} \\ &\quad + \boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{A}\big(\boldsymbol{D}+\boldsymbol{B}\big)^{-1}\boldsymbol{\rho}. \end{aligned} \tag{35}$$

According to Lemma 1 and $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y} = \mathrm{tr}\{\boldsymbol{y}\boldsymbol{x}^{\mathrm{T}}\}$, (34) can be rewritten as

$$\begin{aligned} \dot{V}_1(t) =& -\boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}(\boldsymbol{L}+\boldsymbol{B})(\boldsymbol{\varepsilon}+\boldsymbol{d}-f_0\mathbf{1}_N) \\ & - \frac{1}{2}k\boldsymbol{s}^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{s} - \mathrm{tr}\{\tilde{\boldsymbol{W}}^{\mathrm{T}}\boldsymbol{\varphi}\boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}(\boldsymbol{D}+\boldsymbol{B})\} \\ & + \mathrm{tr}\{\tilde{\boldsymbol{W}}^{\mathrm{T}}\boldsymbol{\varphi}\boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{A}\} \qquad (36) \\ & + \boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{A}(\boldsymbol{D}+\boldsymbol{B})^{-1}\boldsymbol{\rho}. \end{aligned} \tag{37}$$

As $\dot{\tilde{\boldsymbol{W}}} = \dot{\boldsymbol{W}} - \dot{\hat{\boldsymbol{W}}} = -\dot{\hat{\boldsymbol{W}}}$, by substituting it into (25), we can get the derivative of $V_2(t)$ with respect to time,

$$\begin{aligned} \dot{V}_2(t) &= \frac{1}{2}\mathrm{tr}\{\tilde{\boldsymbol{W}}^{\mathrm{T}}\boldsymbol{F}^{-1}\tilde{\boldsymbol{W}}\} \\ &= \mathrm{tr}\{\tilde{\boldsymbol{W}}^{\mathrm{T}}\boldsymbol{F}^{-1}[\boldsymbol{F}\boldsymbol{\phi}\boldsymbol{r}\boldsymbol{P}(\boldsymbol{D}+\boldsymbol{B}) + \zeta\boldsymbol{F}\hat{\boldsymbol{W}}]\} \\ &= \mathrm{tr}\{\tilde{\boldsymbol{W}}^{\mathrm{T}}\boldsymbol{\varphi}\boldsymbol{s}\boldsymbol{P}(\boldsymbol{D}+\boldsymbol{B})\} + \zeta\mathrm{tr}\{\tilde{\boldsymbol{W}}^{\mathrm{T}}\hat{\boldsymbol{W}}\} \\ &= \mathrm{tr}\{\tilde{\boldsymbol{W}}^{\mathrm{T}}\boldsymbol{\varphi}\boldsymbol{s}\boldsymbol{P}(\boldsymbol{D}+\boldsymbol{B})\} + \zeta\mathrm{tr}\{\tilde{\boldsymbol{W}}^{\mathrm{T}}\boldsymbol{W}\} \\ &\quad - \zeta\mathrm{tr}\{\tilde{\boldsymbol{W}}^{\mathrm{T}}\tilde{\boldsymbol{W}}\} \end{aligned} \tag{38}$$

As for the derivative of $V_3(t)$ with respect to time, we get

$$\dot{V}_3(t) = \mathrm{tr}\{\boldsymbol{E}_2\boldsymbol{P}_1\boldsymbol{E}_1^{\mathrm{T}}\}. \tag{39}$$

Substitute (10) into (39) and use (11),to get

$$\begin{aligned}\dot{V}_3(t) &= -\frac{\beta}{2}\mathrm{tr}\{\boldsymbol{E}_1\boldsymbol{E}_1^{\mathrm{T}}\} + \mathrm{tr}\{\boldsymbol{s}\boldsymbol{\Gamma}^{\mathrm{T}}\boldsymbol{P}_1\boldsymbol{E}_1^{\mathrm{T}}\}\\ &\le -\frac{\beta}{2}\left\|\boldsymbol{E}_1\right\|_{\mathrm{F}}^2 + \bar{\sigma}(\boldsymbol{P}_1)\left\|\boldsymbol{s}\right\|\left\|\boldsymbol{E}_1\right\|_{\mathrm{F}}. \end{aligned} \tag{40}$$

Overall, the derivative of Lyapunov function $V(t)$ is

$$\begin{aligned}\dot{V}(t) =&\dot{V}_1(t) + \dot{V}_2(t) + \dot{V}_3(t)\\ =&-\boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\big(\boldsymbol{L}+\boldsymbol{B}\big)\big(\boldsymbol{\varepsilon}+\boldsymbol{d}-f_0\boldsymbol{1}_N\big)\\ &-\frac{1}{2}k\boldsymbol{s}^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{s} + \mathrm{tr}\{\tilde{\boldsymbol{W}}^{\mathrm{T}}\varphi\boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{A}\}\\ &+\zeta\mathrm{tr}\{\tilde{\boldsymbol{W}}^{\mathrm{T}}\boldsymbol{W}\} - \zeta\mathrm{tr}\{\tilde{\boldsymbol{W}}^{\mathrm{T}}\tilde{\boldsymbol{W}}\}\\ &+\boldsymbol{s}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{A}\big(\boldsymbol{D}+\boldsymbol{B}\big)^{-1}\boldsymbol{\rho} - \frac{\beta}{2}\mathrm{tr}\{\boldsymbol{E}_1\boldsymbol{E}_1^{\mathrm{T}}\}\\ &+\mathrm{tr}\{\boldsymbol{s}\boldsymbol{\Gamma}^{\mathrm{T}}\boldsymbol{P}_1\boldsymbol{E}_1^{\mathrm{T}}\}\\ \le&\bar{\sigma}(\boldsymbol{P})\bar{\sigma}\big(\boldsymbol{L}+\boldsymbol{B}\big)\bar{T}\|\boldsymbol{s}\| - \frac{1}{2}k\underline{\sigma}(\boldsymbol{Q})\|\boldsymbol{s}\|^2\\ &+\bar{\varphi}\bar{\sigma}(\boldsymbol{P})\bar{\sigma}(\boldsymbol{A})\|\tilde{\boldsymbol{W}}\|_F\|s\|\\ &+\zeta\bar{\boldsymbol{W}}\|\tilde{\boldsymbol{W}}\|_F - \zeta\|\tilde{\boldsymbol{W}}\|_F^2\\ &+\frac{\bar{\sigma}(\boldsymbol{P})\bar{\sigma}(\boldsymbol{A})}{\underline{\sigma}(\boldsymbol{D}+\boldsymbol{B})}\big(\|s\|\|\boldsymbol{E}_1\|_F\|\boldsymbol{\Gamma}\|_F\|\bar{\boldsymbol{\alpha}}\|\\ &\quad+\|s\|^2\|\boldsymbol{\Gamma}\|\|\bar{\boldsymbol{\alpha}}\|\big) - \frac{\beta}{2}\|\boldsymbol{E}_1\|_F^2\\ &+\bar{\sigma}(P_1)\|\boldsymbol{s}\|\|\boldsymbol{E}_1\|_F, \end{aligned} \tag{41}$$

where $\bar{T} = \bar{\varepsilon} + \bar{d} + \bar{F}$.

Note that the definition of r, g and h in (28), (41) can be rewritten as

$$\begin{aligned}&\dot{V}(t)\\ &\le -\left(\frac{1}{2}k\underline{\sigma}(\boldsymbol{Q}) - \frac{\bar{\sigma}(\boldsymbol{P})\bar{\sigma}(\boldsymbol{A})}{\underline{\sigma}(\boldsymbol{D}+\boldsymbol{B})}\left\|\bar{\boldsymbol{\alpha}}\right\|\right)\cdot\left\|\boldsymbol{s}\right\|^2\\ &\quad-\zeta\|\tilde{\boldsymbol{W}}\|_{\mathrm{F}}^2 - \frac{\beta}{2}\left\|\boldsymbol{E}_1\right\|_{\mathrm{F}}^2\\ &\quad+\bar{\boldsymbol{\varphi}}\bar{\sigma}(\boldsymbol{P})\bar{\sigma}(A)\cdot\|\tilde{\boldsymbol{W}}\|_{\mathrm{F}}\left\|\boldsymbol{s}\right\|\\ &\quad+\left(\frac{\bar{\sigma}(\boldsymbol{P})\bar{\sigma}(\boldsymbol{A})}{\bar{\sigma}(\boldsymbol{D}+\boldsymbol{B})}\|\boldsymbol{\Lambda}\|_{\mathrm{F}}\left\|\bar{\boldsymbol{\alpha}}\right\| + \bar{\sigma}(\boldsymbol{P}_1)\right)\left\|\boldsymbol{s}\right\|\left\|\boldsymbol{E}_1\right\|_{\mathrm{F}}\\ &\quad+\bar{\sigma}(\boldsymbol{P})\bar{\sigma}(\boldsymbol{L}+\boldsymbol{B})\bar{T}\left\|\boldsymbol{s}\right\| + \zeta\bar{W}\left\|\tilde{\boldsymbol{W}}\right\|_{\mathrm{F}}\\ &= -\left(\frac{1}{2}k\underline{\sigma}(Q) - h\right)\|\boldsymbol{s}\|^2 - \zeta\left\|\tilde{\boldsymbol{W}}\right\|_F^2\\ &\quad-\frac{\beta}{2}\left\|\boldsymbol{E}_1\right\|_{\mathrm{F}}^2 - 2r\|\tilde{\boldsymbol{W}}\|_{\mathrm{F}}\left\|\boldsymbol{s}\right\| - 2g\left\|\boldsymbol{s}\right\|\left\|\boldsymbol{E}_1\right\|_{\mathrm{F}}\\ &\quad+\bar{\sigma}(P)\bar{\sigma}(L+B)\bar{T}\left\|\boldsymbol{s}\right\| + \zeta\bar{W}\|\tilde{\boldsymbol{W}}\|_{\mathrm{F}}\end{aligned}$$

$$\begin{aligned}&= -\begin{bmatrix}\|\boldsymbol{E}_1\|_{\mathrm{F}}\\ \left\|\tilde{\boldsymbol{W}}\right\|_{\mathrm{F}}\\ \|\boldsymbol{s}\|\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}\frac{\beta}{2} & 0 & g\\ 0 & \zeta & r\\ g & r & \frac{1}{2}k\bar{\sigma}(Q)-h\end{bmatrix}\\ &\quad\times\begin{bmatrix}\|\boldsymbol{E}_1\|_{\mathrm{F}}\\ \left\|\tilde{\boldsymbol{W}}\right\|_{\mathrm{F}}\\ \|\boldsymbol{s}\|\end{bmatrix}\\ &\quad+\begin{bmatrix}0 & \zeta\bar{W} & \bar{\sigma}(\boldsymbol{P})\bar{\sigma}(\boldsymbol{L}+\boldsymbol{B})\bar{T}\end{bmatrix}\begin{bmatrix}\|\boldsymbol{E}_1\|_{\mathrm{F}}\\ \|\boldsymbol{W}\|_{\mathrm{F}}\\ \|\boldsymbol{s}\|\end{bmatrix}.\end{aligned} \tag{42}$$

Let

$$\begin{aligned}\boldsymbol{z} =&\Big[\|\boldsymbol{E}_1\|_{\mathrm{F}} \quad \|\tilde{\boldsymbol{W}}\|_{\mathrm{F}} \quad \|\boldsymbol{s}\|\Big]^{\mathrm{T}},\\ \boldsymbol{\Xi} =&\begin{bmatrix}\frac{\beta}{2} & 0 & g\\ 0 & \zeta & r\\ g & r & \frac{1}{2}k\underline{\sigma}(Q)-h\end{bmatrix},\\ \boldsymbol{\omega} =&\Big[0 \;\; \zeta\overline{W} \;\; \bar{\sigma}(\boldsymbol{P})\bar{\sigma}(\boldsymbol{L}+\boldsymbol{B})\overline{T}\Big]^{\mathrm{T}}.\end{aligned} \tag{43}$$

Accordingly,

$$\dot{V}(t) \le -\boldsymbol{z}^{\mathrm{T}}\boldsymbol{\Xi}z + \boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{z} = -V_z(\boldsymbol{z}). \tag{44}$$

The condition for the system to achieve asymptotic stability is that $V_z(z)$ is a positive definite function, that is, the following two conditions are met:

(i) the matrix Ξ is positive definite;

(ii) $\|\boldsymbol{z}\| > \|\boldsymbol{\omega}\|/\overline{\sigma}(\boldsymbol{\Xi})$.

In order to validate the positive definiteness of the matrix $\boldsymbol{\Xi}$, we can check whether all main sub-determinants are positive, i.e.,

$$\begin{aligned}\beta &> 0,\\ \beta\zeta &> 0,\\ \zeta\Big[\beta\big(\frac{1}{2}k\underline{\sigma}(Q)-h\big)-2g^2\Big]-\beta r^2 &> 0.\end{aligned} \tag{45}$$

It is easy to see that $\|\omega\|_1 > \|\omega\|$, if $\|\boldsymbol{z}\| \ge Bd$, the condition (ii) above holds, we get

$$Bd = \frac{\|\boldsymbol{\omega}\|_1}{\underline{\sigma}(\boldsymbol{\Xi})} = \frac{\bar{\sigma}(\boldsymbol{P})\bar{\sigma}(\boldsymbol{L}+\boldsymbol{B})\bar{T}+\zeta\bar{W}}{\underline{\sigma}(\boldsymbol{\Xi})}. \tag{46}$$

Thus, both conditions (i) and (ii) are met, and we can get

$$\dot{V}(t) \le -V_z(\boldsymbol{z}), \quad \forall\,\|\boldsymbol{z}\| \ge Bd, \tag{47}$$

and $V_z(\boldsymbol{z})$ is a continuous positive definite function.

5. Simulation

In this section, we consider the consensus verification experiments of two different MASs, i.e., homogeneous and heterogeneous MASs, where the homogeneous MAS means that the dynamic models of all agents are the same, while the heterogeneous MAS means that they are different. The network communication topology of both the two experiments is shown in Fig. 1, which includes 5 follower agents and a leader. To simplify the simulation design, assume that the weight of the edges communicating with each other are both equal to 1.

We assume the dynamic models of the five followers are third-order uncertain nonlinear systems:

$$\begin{cases} \dot{x}_{i,1}(t) = x_{i,2}(t), \\ \dot{x}_{i,2}(t) = x_{i,3}(t), \\ \dot{x}_{i,3}(t) = f_i(t, x_i) + d_i(t) + u_i(t). \end{cases}$$

5.1. Consensus of homogeneous multi-agent systems.

The parameters of the RBF neural network are randomly initialized in advance, and the initialized state of the leader and follower agents is

$$\begin{aligned} \boldsymbol{x}_0 &= [x_{0,1}, x_{0,2}, x_{0,3}]^{\mathrm{T}} = [6, \quad 0.38, \quad 0]^{\mathrm{T}}, \\ \boldsymbol{x}_1 &= [x_{1,1}, x_{1,2}, x_{1,3}]^{\mathrm{T}} = [20, \quad -0.76, \quad 0]^{\mathrm{T}}, \\ \boldsymbol{x}_2 &= [x_{2,1}, x_{2,2}, x_{2,3}]^{\mathrm{T}} = [0, \quad 0.91, \quad 0]^{\mathrm{T}}, \\ \boldsymbol{x}_3 &= [x_{3,1}, x_{3,2}, x_{3,3}]^{\mathrm{T}} = [-5, \quad 0.77, \quad 0]^{\mathrm{T}}, \\ \boldsymbol{x}_4 &= [x_{4,1}, x_{4,2}, x_{4,3}]^{\mathrm{T}} = [-2, \quad 0.88, \quad 0]^{\mathrm{T}}, \\ \boldsymbol{x}_5 &= [x_{5,1}, x_{5,2}, x_{5,3}]^{\mathrm{T}} = [9, \quad 0.33, \quad 0]^{\mathrm{T}}. \end{aligned}$$

Let the nonlinear function $f_i, i = 0, 1, \dots, 5$ of the homogeneous system be

$$\begin{aligned} f_i(t, x_i(t)) = {} & 20\sin(2t + 0.2) + 10\sin(2x_{i,2}) \\ & - 0.2x_{i,1} + 0.5x_{i,2}, \end{aligned}$$

where f_0 is the term of the leader agent.

For the i-th agent, it is assumed that it contains uncertainties such as external disturbances and sensor noise, which are uniformly modeled as $0.05\sin(t)$. The state trajectories of all the agents of the third-order system are shown in Fig. 2. The position states of the five homogeneous followers are gradually approaching the position state of leader 0, that is, the tracking consensus of the MAS is achieved.

5.2. Consensus of heterogeneous multi-agent systems.

To illustrate that the proposed protocol can be applied to heterogeneous MASs, reconsider the third-order MAS of with 5 followers, where the nonlinear function f_i for each

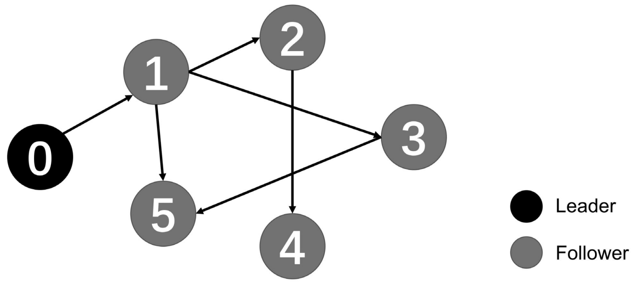


Fig. 1. Directed communication topology with 6 followers and 1 leader.

agent i is unique,

$$\begin{aligned} f_0(t, x_0(t)) &= 0.5\sin x_{0,1} - 0.1x_{0,2} + \cos(1.2t), \\ f_1(t, x_1(t)) &= -0.2\sin x_{1,1} - 0.5x_{1,2}, \\ f_2(t, x_2(t)) &= -1.5\sin x_{2,1} - x_{2,2} + 0.5\cos(t), \\ f_3(t, x_3(t)) &= 0.8\sin x_{3,1} - \cos x_{3,3}, \end{aligned}$$

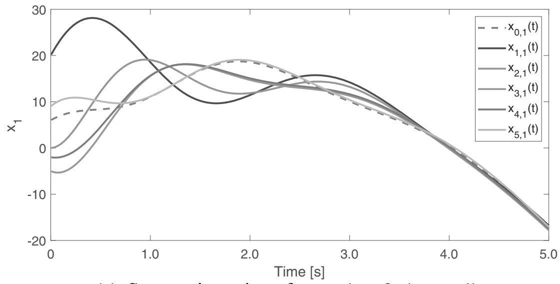
$$\begin{aligned} f_4(t, x_4(t)) &= -\sin x_{4,1} - 0.1\cos(0.1t), \\ f_5(t, x_5(t)) &= -1.8\sin x_{5,1} + 0.1\sin x_{5,2} + \cos(2t). \end{aligned}$$

The controller and the parameters of the neural network, the network topology, the initial state of the system, the uncertain term $d_i(t)$, and the dynamics model of the leader are the same as those in Section 5.1. The state trajectories of all agents in the third-order system are shown in Fig. 3. The position states of the 6 heterogeneous followers are gradually approaching the position state of leader 0, which means that they finally achieve state consensus, i.e.,

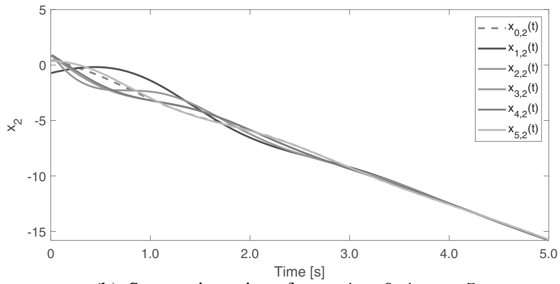
$$\lim_{t\to\infty} \|\boldsymbol{x}_i(t) - \boldsymbol{x}_j(t)\| = 0, \quad \forall i, j = 0, 1, \dots, N.$$

Through simulations, it is verified that the proposed adaptive consensus control protocol is not only applicable to homogeneous MASs, but also to heterogeneous MASs.

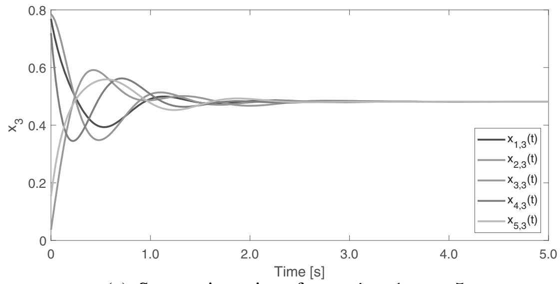
It should be noted that, in view of the problems of external disturbances and uncertain items in the system, currently there are mainly robust control methods and state observer methods. Robust control has a certain inhibitory effect on external disturbances, but it cannot effectively eliminate the influence of external disturbances on consistency. At present, the more commonly used method is to use a state observer to estimate the uncertain items of the MAS to compensate for the unknown items, to achieve the consistency of the multi-agent system, and the design is for linear systems. In this paper, a high-order nonlinear uncertain MAS have been investigated. The nonlinear term of the system is approximated by an RBF neural network. A sliding mode controller is designed



(a) State trajectories of $x_{i,1}, i = 0, 1, \ldots, 5$.

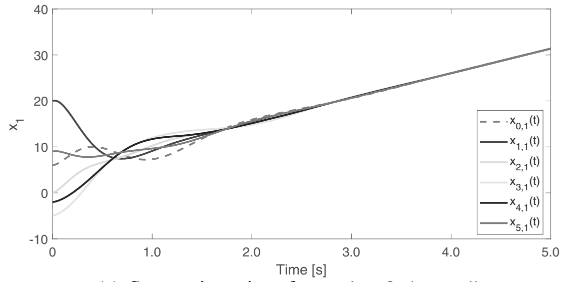


(b) State trajectories of $x_{i,2}, i = 0, 1, \ldots, 5$.

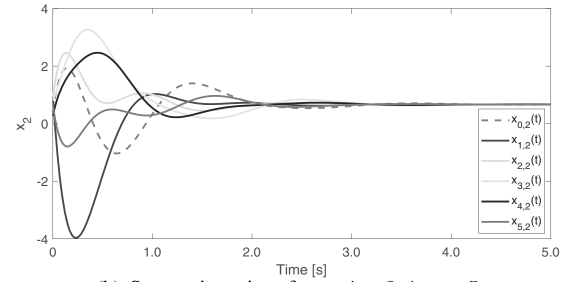


(c) State trajectories of $x_{i,3}, i = 1, \ldots, 5$.

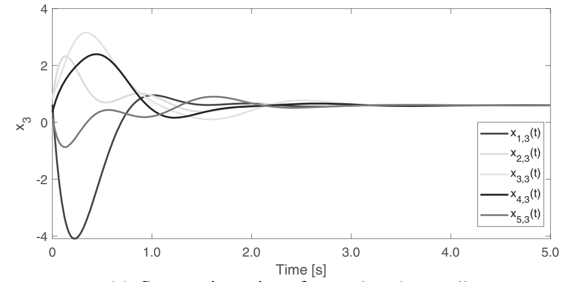
Fig. 2. State trajectories of homogeneous systems.



(a) State trajectories of $x_{i,1}, i = 0, 1, \ldots, 5$.



(b) State trajectories of $x_{i,2}, i = 0, 1, \ldots, 5$.



(c) State trajectories of $x_{i,3}, i = 1, \ldots, 5$.

Fig. 3. State trajectories of heterogeneous systems.

to compensate for external disturbances, so that the multi-agent system can be stabilized, thereby realizing the leader–follower consistency problem.

6. Conclusion

This paper proposed a neuro-adaptive cooperative control method for leader-following MASs, where an adaptive RBF neural network is adopted to approximate the nonlinear terms. The proposed method does not need to know the upper bounds of nonlinear terms and uncertain terms. Theoretical results showed that the finite time required for a Brunovsky-type high-order nonlinear agent system to reach consensus depends not only on the relevant control parameters and the information topology of the designed algorithm, but also on the initial state of the MASs. The simulation results proved that the proposed adaptive sliding mode control algorithm of the distributed RBF neural network approximation can not only effectively deal with unknown or even heterogeneous nonlinear dynamics, but also has a good anti-interference ability to ensure the convergence of system tracking errors.

Acknowledgment

This work is supported by the Natural Science Foundation of the Anhui Higher Education Institutions of China under the grant no. Kj2020A0987, and the Key Natural Science Foundation of Hefei Technology College under the grant no. 202014KJA004.

References

Abdelrahem, M., Hackl, C.M. and Kennel, R. (2018). Finite position set-phase locked loop for sensorless control of direct-driven permanent-magnet synchronous generators, *IEEE Transactions on Power Electronics* **33**(4): 3097–3105.

Arandabricaire, E., Moog, C. and Pomet, J. (1995). Infinitesimal Brunovsky form for nonlinear systems with applications to dynamic linearization, *Geometry in Nonlinear Control and Differential Inclusions* **13**(32): 19–33.

Aryankia, K. and Selmic, R.R. (2021). Neuro-adaptive formation control and target tracking for nonlinear multi-agent systems with time-delay, *IEEE Control Systems Letters* **5**(3): 791–796.

Bartoszewicz, A. and Adamiak, K. (2019). A reference trajectory based discrete time sliding mode control strategy, *International Journal of Applied Mathematics and Computer Science* **29**(3): 517–525, DOI: 10.2478/amcs-2019-0038.

Bechlioulis, C.P. and Rovithakis, G.A. (2017). Decentralized robust synchronization of unknown high order nonlinear multi-agent systems with prescribed transient and steady state performance, *IEEE Transactions on Automatic Control* **62**(1): 123–134.

Couzin, I.D., Karuse, J., James, R., Ruxton, G.D. and Franks, N.R. (2002). Collective memory and spatial sorting in animal groups, *Journals of Theoretical Biology* **218**(1): 1–11.

Cui, B., Zhao, C., Ma, T. and Feng, C. (2016). Leader-following consensus of nonlinear multi-agent systems with switching topologies and unreliable communications, *Neural Computing and Applications* **27**(4): 909–915.

Das, A. and Lewis, F.L. (2010). Cooperative adaptive control for synchronization of second-order systems with unknown nonlinearities, *International Journal of Robust and Nonlinear Control* **21**(13): 1509–1524.

Duan, P., Liu, K., Huang, N. and Duan, Z. (2020). Event-based distributed tracking control for second-order multiagent systems with switching networks, *IEEE Transactions on Systems, Man, and Cybernetics: Systems* **50**(9): 3220–3230.

Farrera, B., López-Estrada, F.R., Chadli, M., Valencia-Palomo, G. and Gómez-Peñate, S. (2020). Distributed fault estimation of multi-agent systems using a proportional-integral observer: A leader-following application, *International Journal of Applied Mathematics and Computer Science* **30**(3): 551–560, DOI: 10.34768/amcs-2020-0040.

Fathi, A., Shafiee, Q. and Bevrani, H. (2018). Robust frequency control of microgrids using an extended virtual synchronous generator, *IEEE Transactions on Power Systems* **33**(6): 6289–6297.

Fu, J., Lv, Y., Wen, G., Yu, X. and Huang, T. (2020). Velocity and input constrained coordination of second-order multi-agent systems with relative output information, *IEEE Transactions on Network Science and Engineering* **7**(3): 1925–1938.

Huang, J., Dou, L., Fang, H., Chen, J. and Yang, Q. (2015). Distributed backstepping-based adaptive fuzzy control of multiple high-order nonlinear dynamics, *Nonlinear Dynamics* **81**(1–2): 63–75.

Joshi, K.D. (1983). *Introduction to General Topology*, John Wiley & Sons, New York.

Li, B., Yang, H., Chen, Z. and Liu, Z. (2018). Distributed containment control of multi-agent systems with general linear dynamics and time-delays, *International Journal of Control, Automation and Systems* **16**(6): 2718–2726.

Ling, S., Wang, H. and Liu, P.X. (2019). Adaptive fuzzy dynamic surface control of flexible-joint robot systems with input saturation, *IEEE/CAA Journal of Automatica Sinica* **6**(1): 97–107.

Liu, Y. and Jia, Y.M. (2008). Leader-following consensus protocol for second-order multi-agent systems using neural networks, *Proceedings of the 27th Chinese Control Conference, Kunming, China*, pp. 535–539.

Low, D. J. (2000). Following the crowd, *Nature* **407**(6803): 465–466.

Lu, K., Liu, Z., Lai, G., Chen, C.L. and Zhang, Y. (2021). Adaptive consensus tracking control of uncertain nonlinear multiagent systems with predefined accuracy, *IEEE Transactions on Cybernetics* **51**(1): 405–415.

Ma, Q. and Miao, G.Y. (2015). Output consensus for heterogeneous multi-agent systems with linear dynamics, *Applied Mathematics and Computation* **271**(15): 548–555.

Meng, W., Liu, P.X., Yang, Q. and Sun, Y. (2020). Distributed synchronization control of nonaffine multiagent systems with guaranteed performance, *IEEE Transactions on Neural Networks and Learning Systems* **31**(5): 1571–1580.

Miao, Z., Liu, Y., Wang, Y., Yi, G. and Fierro, R. (2018). Distributed estimation and control for leader-following formations of nonholonomic mobile robots, *IEEE Transactions on Automation Science and Engineering* **15**(4): 1946–1954.

Ni, J., Liu, L., Liu, C. and Liu, J. (2017). Fixed-time leader-following consensus for second-order multiagent systems with input delay, *IEEE Transactions on Industrial Electronics* **64**(11): 8635–8646.

Nian, H. and Jiao, Y. (2020). Improved virtual synchronous generator control of DFIG to ride-through symmetrical voltage fault, *IEEE Transactions on Energy Conversion* **35**(2): 672–683.

Parrish, J.K., Viscido, S.V., and Grunbaum, D. (2002). Self-organized fish schools: An examination of emergent properties, *Biological Bulletin* **202**(3): 296–305.

Qin, J., Zhang, G., Zheng, W.X. and Kang, Y. (2019). Neural network-based adaptive consensus control for a class of nonaffine nonlinear multiagent systems with actuator faults, *IEEE Transactions on Neural Networks and Learning Systems* **30**(12): 3633–3644.

Shen, Q., Shi, P., Zhu, J., Wang, S. and Shi, Y. (2020). Neural networks-based distributed adaptive control of nonlinear multiagent systems, *IEEE Transactions on Neural Networks and Learning Systems* **31**(3): 1010–1021.

Wang, Q., Wang, J.L., Wu, H.N. and Huang, T. (2020). Consensus and h_∞ consensus of nonlinear second-order multi-agent systems, *IEEE Transactions on Network Science and Engineering* **7**(3): 1251–1264.

Wu, Z.-G., Xu, Y., Pan, Y.-J., Su, H. and Tang, Y. (2018). Event-triggered control for consensus problem in multi-agent systems with quantized relative state measurements and external disturbance, *IEEE Transactions on Circuits and Systems I: Regular Papers* **65**(7): 2232–2242.

Yang, N. and Li, J. (2020). Distributed robust adaptive learning coordination control for high-order nonlinear multi-agent systems with input saturation, *IEEE Access* **8**: 9953–9964.

Yao, D., Li, H., Lu, R. and Shi, Y. (2020). Distributed sliding-mode tracking control of second-order nonlinear multiagent systems: An event-triggered approach, *IEEE Transactions on Cybernetics* **50**(9): 3892–3902.

Zhang, A., Zhang, S., Chen, Q. and Zhu, W. (2018). Consensus control of a class of second-order nonlinear multi-agent systems based on distributed adaptive protocol, *Proceedings of the 30th Chinese Control and Decision Conference, CCDC 2018, Shenyang, China*, pp. 4668–4672.

Zhang, H. and Lewis, F.L. (2012). Adaptive cooperative tracking control of higher-order nonlinear systems with unknown dynamics, *Automatica* **48**(7): 1432–1439.

Zhao, K., Song, Y., Meng, W., Chen, C.P. and Chen, L. (2021). Low-cost approximation-based adaptive tracking control of output-constrained nonlinear systems, *IEEE Transactions on Neural Networks and Learning Systems* **32**(11): 4890–4900.

Zhang. Z (2008). Predictive function control of the single-link manipulator with flexible joint, *2008 IEEE International Conference on Automation and Logistics, Qingdao, China*, pp. 1184–1188.

Zong, C., Ji, Z., Tian, L. and Zhang, Y. (2019). Distributed multi-robot formation control based on bipartite consensus with time-varying delays, *IEEE Access* **7**: 144790–144798.

Zou, W., Shi, P., Xiang, Z. and Shi, Y. (2020). Finite-time consensus of second-order switched nonlinear multi-agent systems, *IEEE Transactions on Neural Networks and Learning Systems* **31**(5): 1757–1762.

Cheng Peng was born in Chaohu, Anhui Province, PR China, in 1986. He holds an MSc degree from the Hefei University of Technology, PR China. Now he works in the School of Mechanical and Electrical Engineering, Hefei Technology College. His research interests include mechanical design, hydraulic and pneumatic technology, and control engineering.

Anguo Zhang was born in Hefei City, Anhui Province, PR China, in 1990. He received his BSc and MSc degrees in control engineering from Chongqing University in 2012 and 2016, respectively. Since 2018, he has been a researcher and a senior engineer in the Research Institute of Ruijie, Ruijie Networks Co., Ltd. He is also pursuing a PhD in communication and information systems at Fuzhou University. His research interests include machine learning, artificial neural networks, control theory and applications.

Junyu Li graduated from the Wuhan Institute of Technology in 2013 with a major in mechanical engineering. She currently works as a lecturer at Hefei Technology College. Her main research interests include mechanical engineering, machine learning and artificial neural networks.

Received: 14 June 2021
Revised: 3 July 2021
Accepted: 5 August 2021