

Research Article

Neural-Network-Based Collaborative Control for Continuous Unknown Nonlinear Systems

Siyu Gao and Xin Wang 

Chongqing Key Laboratory Nonlinear Circuits & Intelligent, Southwest University, Chongqing 400715, China

Correspondence should be addressed to Xin Wang; xinwangswu@163.com

Received 11 January 2021; Revised 1 March 2021; Accepted 10 March 2021; Published 24 March 2021

Academic Editor: Shiping Wen

Copyright © 2021 Siyu Gao and Xin Wang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes an NN-based cooperative control scheme for a type of continuous nonlinear system. The model studied in this paper is designed as an interconnection topology, and the main consideration is the connection mode of the undirected graph. In order to ensure the online sharing of learning knowledge, this paper proposes a novel weight update scheme. In the proposed update scheme, the weights of the neural network are discrete, and these discrete weights can gradually approach the optimal value through cooperative learning, thereby realizing the control of the unknown nonlinear system. Through the trained neural network, it is proved if the interconnection topology is undirected and connected, the state of the unknown nonlinear system can converge to the target trajectory after a finite time, and the error of the system can converge to a small neighbourhood around the origin. It is also guaranteed that all closed-loop signals in the system are bounded. A simulation example is provided to more intuitively prove the effectiveness of the proposed distributed cooperative learning control scheme at the end of the article.

1. Introduction

The control of the system is a big problem for humans to solve the mechanical operations, and in real life, most systems are nonlinear systems. This means that we cannot solve the control problem for this type of system through a unified constant equation. For the control of complex nonlinear systems, many researchers have made corresponding research and established a lot of different solution models, most of which are based on adaptive neural networks.

The adaptive neural network usually acts as a “perceptron” in the control of the system, which means that we can fit some unknown functions through the neural network. Due to the global estimation characteristics, learning ability, and tolerance of fuzzy input of neural networks, the control and learning method based on neural network has aroused great interest [1–15]. Because the adaptive method in the neural network can automatically adjust the processing method and processing order, so as to find the potential structural features to obtain the best processing method. The

study of using adaptive method to control the system has also been widely concerned [16–19]. Most of these control methods are based on the characteristics of adaptive neural networks, so as to realize the identification process of unknown functions and then use Lyapunov functions for stability analysis [20–23]. In this type of control process, the selection and training of neural networks often have a large impact on system performance. This is because the stability analysis method based on Lyapunov cannot guarantee that the weight of the neural network reaches the optimal value, which means that even for the same task, the weight of the neural network still needs to be adjusted repeatedly. In other words, the neural network control method using only Lyapunov’s stability theory cannot fit a given nonlinear system well.

In order to solve this problem, the radial RBF neural network is considered to be used in the recognition of nonlinear systems. By using the persistently exciting feature of the RBF neural network [24], the convergence of the neural network weights can be guaranteed to a certain extent. The PE condition further ensures that the fitting error

of the nonlinear system can be controlled within a smaller range. At the same time, a deterministic learning mechanism is proposed on the basis of the PE condition [25], and this kind of mechanism ensures that the past dynamics are stored in the form of “pairs,” so that the new control process can be adjusted accordingly.

In addition, a cooperative learning scheme is often mentioned in neural network-based control systems. This is because cooperative learning can divide the task into several parts, which can be distributed to multiple agents, and the learning efficiency can be improved through the communication between the agents [26].

In [27], a cooperative recognition technology based on the neural network is introduced, but this type of technology can only identify unknown nonlinear systems and cannot control them.

Benefiting from previous research, this paper proposes an adaptive cooperative control scheme for continuous unknown nonlinear systems based on the radial basis function neural network. Using this control scheme, the continuous unknown nonlinear function can be controlled with arbitrary precision under the given reference trajectory, and it is guaranteed that all signals are ultimately bounded. The tracking error also converges in a small neighbourhood centred at the origin.

The writing steps of this article are as follows: In the first part, the basic knowledge used in this article will be briefly introduced, and relevant lemmas that will be used in the proof phase will be derived from different knowledge. In the second part, the problem discussed in this article will be described, and a model of the problem to be solved will be established. Then, it is clarified to what extent this problem can be solved using the scheme proposed in this article. This part will get specific results, so it is an important part of this article. In the third part, the questions raised in the second part will be proved accordingly. In the fourth part, simulation will be conducted through examples to prove the correctness and effectiveness of the results. The last part will make a conclusion for this article.

2. Preliminaries

2.1. Kronecker Product [28]

Definition 1. Let $X \in R^{m \times n}$, $Y \in R^{p \times q}$; then, the Kronecker product between X and Y has the following form:

$$X \otimes Y = \begin{bmatrix} x_{11}Y & \cdots & x_{1n}Y \\ \vdots & \ddots & \vdots \\ x_{m1}Y & \cdots & x_{mn}Y \end{bmatrix}. \quad (1)$$

2.2. Algebraic Graph Theory. Generally speaking, we can express the connections between systems in the form of graphs. For an undirected graph $G = (V, E)$, it is composed of a set of vertices V where $V = \{v_1, v_2, \dots, v_n\}$ and a set of edges E , where $E = \{e_1, e_2, \dots, e_n\}$. Each side in E is connected with the vertices in V . $A = [a_{ij}]_{n \times n}$ is the adjacency

matrix of this graph, where each element has $a_{ij} \geq 0$. This adjacency matrix can be expressed as a square matrix: $|V| \times |V|$. Assuming that there is no self-edge in the figure, that is, $a_{ii} = 0$, it is defined that the exchange of information from node v_i to v_j can be expressed as $e_{i,j} = (v_i, v_j)$. The Laplacian matrix in the graph can be defined as $\mathcal{L} = [l_{ij}] \in R^{n \times n}$. This matrix is related to the adjacency matrix A and satisfies the condition of $i \neq j$, $l_{ii} = \sum_{j=1}^n a_{ij}$, where $l_{ij} = -a_{ij}$.

Lemma 1 (See [29]). *We think that the symmetric Laplacian matrix in an undirected graph G can be expressed as \mathcal{L} . So, for this undirected graph, \mathcal{L} has at least one zero eigenvalue, and the other nonzero eigenvalues of \mathcal{L} are all positive numbers. In addition, if and only if G is connected, \mathcal{L} has only a simple zero eigenvalue, the eigenvector corresponding to this eigenvalue is $(1/\sqrt{N})\mathbf{1}_n$, and all other eigenvalues are positive numbers.*

2.3. PE Condition and Uniformly Globally Exponentially Stable. We consider a system as shown below:

$$\begin{aligned} \dot{x} &= f(x, t), \\ x(t_0) &= x_0, \end{aligned} \quad (2)$$

where f denotes a nonlinear function with t as a variable. It is assumed that this nonlinear function is continuous and satisfies the following Lipschitz condition:

$$|f(x_1) - f(x_2)| \leq L(x_1 - x_2), \quad (3)$$

where $x_1, x_2 \in R$. The abovementioned condition ensures that the solution in system (2) exists and is unique. $x(t)$ represents the solution to (2) from the specific initial situation (t_0, x_0) .

Definition 2. In system (2), if there is a uniform and local exponentially stable equilibrium point when $x = 0$, then there are two positive definite constants c_1, c_2 , and the following condition is met when $r > 0$:

$$\|x(t)\| \leq c_1 \|x_0\| e^{-c_2(t-t_0)}. \quad (4)$$

If formula (4) is satisfied in any initial condition $x(t_0)$, then the system in (2) is said to be uniformly globally exponentially stable.

Definition 3. (Cooperative PE) (See [30]). We think that $Q(\tau)$ meets the PE condition when $Q(\tau)$ satisfies the following formula:

$$\alpha_1 I \geq \int_t^{t+T_0} Q(\tau) Q(\tau)^T d\tau \geq \alpha_2 I, \quad \forall t_0 \geq 0, \quad (5)$$

where I represents the identity matrix and α_1 and α_2 are constants. In function $Q(\tau)$, it is not necessary for each signal to satisfy the PE condition. By proving the cooperative PE condition, we can use the following lemma to analyse the stability.

Lemma 2 (See [25]). *Consider a linear time-varying system, which is represented in the following way:*

$$\begin{bmatrix} \dot{x}_0 \\ \dot{x}_1 \end{bmatrix} = \begin{bmatrix} \Lambda & Q(t)^T \\ -\Gamma Q(t) & 0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix}, \quad (6)$$

where $x_0 \in R^n$, $x_1 \in R^m$, $x = [x_0^T, x_1^T]^T$ represents the system state and the transpose of Λ is equal to itself. If the two conditions (a) the triple (Λ, E, Γ) is strictly positive real, that is, if there is a symmetric positive definite matrix S that satisfies $\Gamma\Lambda + \Lambda^T\Gamma = -S$ and (b) $Q(t)$ is continuous and bounded are met and it satisfies the PE condition, the derivative of $Q(t)$ is also bounded; then, $x = 0$ is uniformly globally exponentially stable in system (6)

At the same time, considering the influence of external disturbance on stability, we propose the following lemma based on [25].

Lemma 3 (See [25]). *The following system is considered:*

$$\dot{x} = f(x, t) + e(x, t), \quad (7)$$

where $f \in R^n$ is a continuous function on t , which satisfies the local Lipschitz condition in x , and e is locally bounded and meets $\|e(x, t)\| \leq \delta(t)$. We think e is the external noise; then, equation (7) is a system with disturbance added to equation (2). If $x = 0$ is the exponentially stable equilibrium point in system (7), then there is $\|e(x, t)\| \leq \psi$, and ψ and $\delta(t)$ are proportional. This shows that $x(t)$ is uniformly bounded.

Remark 1. In this article, we use Lemma 1 to prove that a conventional system is uniformly exponentially stable under the condition of PE. At the same time, using Lemma 2, we can prove that even after adding the disturbance term, the system can also converge to a smaller value when the disturbance term can converge to a smaller value.

2.4. RBF NNS. Since the RBF neural network can approximate an unknown continuous nonlinear function to any degree, it is often used in the control and recognition process of nonlinear systems. Therefore, for an unknown nonlinear function $f(x)$, in this article, a radial basis function neural network is used to estimate it. Suppose the unknown nonlinear equation is $f(Q)$; then, the RBF NN shown below is used to approximate its value over the compact set Ω_Q :

$$f(Q) = K(Q)^T W + \varepsilon(Q), \quad (8)$$

where $W = [w_1, w_2, \dots, w_n]^T$ represents the desired weight vector in the output layer of the neural network, $K(Q) = [k_1(Q), k_2(Q), \dots, k_n(Q)]^T$ is the activation function in the neural network, n shows the number of neurons in the hidden layer, and $\varepsilon(Q)$ denotes the estimated error of RBF NN. In this article, $k_i(Q)$ uses the activation function which is widely used in RBF NN: Gaussian function, which is in the following form:

$$k_i(Q) = \exp\left[-\frac{\|Q - c_i\|^2}{\eta^2}\right], \quad (9)$$

where $\eta > 0$, $c_i \in \Omega_Q$ denote the width and centre of the activation function, respectively. The desired weight W is regarded as follows:

$$W = \arg \min \left\{ \sup |f(Q) - S(Q)^T \widehat{W}| \right\}, \quad (10)$$

where \widehat{W} is an estimation of the desired weight W . Generally speaking, we think that W is bounded. If $f(Q)$ is continuous and the estimation domain of $f(Q)$ is a compact set, we can guarantee the existence of optimal weights. As for the localized RBF NN, we can use the continuous nonlinear equation shown below to estimate any trajectory:

$$f(Q(t)) = S_\tau(Q(t))^T W_\tau + \varepsilon_\tau(Q(t)), \quad (11)$$

where W_τ represents the subvector of the weight matrix in the neural network, S_τ denotes the activation function vector in RBF NNs, and ε_τ means the estimation error of the neural network. In general, the estimation error using this neural network is very small.

In order to analyse the stability of this neural network, we need to prove that S_τ meets the PE condition; then, the following lemma can be used to prove this requirement.

Lemma 4 (See [25]). *If a trajectory $Q(t)$ is periodic or periodic-like, $Q(t)$ is a continuous graph and its derivative is bounded in Ω . Then, for the localized RBF NN $S(Q)^T W$ whose centre is in a space that can contain Ω , the regression vectors $S_\tau(Q(t))$ defined in (9) of this neural network almost all meet the PE condition.*

3. Cooperative Adaptive Control of a Nonlinear System

3.1. Problem Formulation. Consider a nonlinear system as shown below:

$$\dot{x}_i = f(x_i) + u_i, \quad i = 1, 2, 3, \dots, N, \quad (12)$$

where $u_i \in R$ denotes the control input of system i , $x_i \in R$ is the state of system i , and N is the number of first-order systems. $f(x_i)$ is an unknown nonlinear function that needs to be fitted by RBF NN. The abovementioned expression shows the overall structure of the system that needs to be controlled. In this article, we consider that the state of x in the system is unknown, and the unknown nonlinear function of each node, that is, $f(x_i)$, is uniform. Using the radial basis function neural network mentioned above, we can fit this unknown nonlinear function in the following form:

$$f(x_i) = S(x_i)^T W_i + \varepsilon_i, \quad (13)$$

where ε_i represents the fitting error for this unknown nonlinear function.

For control problems, a reference signal is generally required for tracking. Assuming that the reference signal is uniformly bounded, the form of the reference signal defined in this article is

$$\dot{y}_i = F_d(y_i), \quad i = 1, 2, 3, \dots, N, \quad (14)$$

where $F_d(y_i)$ denotes the given known function, that is, the reference signal. N is the number of reference signals, which represents the same number as the system state. The purpose of this article is to be able to track the different states of the system to these reference trajectories after some reference signals are given. For this, we need to design a controller to make x_i track to y_i and achieve the goal of uniform convergence. The controller model proposed in this paper is

$$u_i = B_i z_i - S(x_i)^T \widehat{W}_i + \dot{\alpha}_i, \quad (15)$$

and among them $z_i = x_i - y_i$, $\dot{\alpha}_i = F_d(y_i)$. z_i represents the state tracking error, $B_i = \text{diag}\{-b_{i1}, -b_{i2}, -b_{i3}, \dots, -b_{in}\}$ is a designed diagonal matrix, and each element satisfies $b_{ij} > 0$, $i = 0, 1, \dots, N$, $j = 0, 1, \dots, n$.

Using the system state model, reference signal form, and controller model proposed above, this paper realizes the neural network-based cooperative adaptive control of the unknown nonlinear system. The control part will be described in detail below.

3.2. RBF NN-Based Control. As mentioned before, the RBF NN can be used to approximate the characteristics of the positional nonlinear function with arbitrary accuracy, so it is used to approximate $f(x_i)$. Inspired by the consensus theory and cooperative learning control, we propose the following cooperative control weight update rule:

$$\dot{\widehat{W}}_i = -\Psi_i [S(x_i, r_i)(x_i - y_i) + \sigma_i \widehat{W}_i] - \Theta_i \sum_{j=1}^N a_{i,j} (\widehat{W}_i - \widehat{W}_j), \quad (16)$$

where \widehat{W}_i represents the weight estimator of the system i . Ψ_i , $\Theta_i > 0$ are the design parameters. $a_{i,j}$ is an element of the adjacency matrix A in graph \mathcal{G} , and this item is used to represent the connection items of each control law. Besides, $\sigma_i > 0$ and $\sigma_i \widehat{W}_i$ participate in this rule as correction terms to ensure the convergence of weights. $-\Theta_i \sum_{j=1}^N a_{i,j} (\widehat{W}_i - \widehat{W}_j)$ shows the coupling relationship between the i -th weight update rule and its neighbours, which is a manifestation of the cooperative learning rule.

In addition, we also define the weight error in the neural network as $\widetilde{W}_i = \widehat{W}_i - W$, where W represents the optimal solution of weights in RBF NN. Then, the derivative of this closed-loop system is

$$\dot{\widetilde{W}}_i = -\Psi_i \left[S(x_i, r_i) z_i + \sigma_i \widehat{W}_i - \Theta_i \sum_{j=1}^N a_{i,j} (\widehat{W}_i - \widehat{W}_j) \right]. \quad (17)$$

We believe that when \widehat{W}_i converges to 0, the weights obtain the optimal solution.

3.3. Main Results. The performance of the distributed cooperative adaptive control is stated as follows. In order to be able to express more clearly in the following proofs, we first define the following notations:

- (1) $S_{\varphi_i}(x_i, r_i)$ and $S_{\overline{\varphi}_i}(x_i, r_i)$, respectively, represent the part of $S(x_i, r_i)$ near and away from the reference trajectory $F_d(y_i)$ when using RBF NN
- (2) $(\cdot)_{j,\varphi_i}$ and $(\cdot)_{j,\overline{\varphi}_i}$, respectively, represent the part of formula $(\cdot)_j$ that is near and away from the reference trajectory $F_d(y_i)$
- (3) $(\cdot)_{i,\varphi}$ and $(\cdot)_{i,\overline{\varphi}}$, respectively, represent the part of formula $(\cdot)_i$ that is near and away from the reference trajectory $F_d(y_i)$
- (4) $(\cdot)_{\varphi}$ and $(\cdot)_{\overline{\varphi}}$, respectively, represent the parts near and far away from all trajectories $F_d(y)$ in the form (\cdot)
- (5) \overline{W}_i represents the mean value of \widehat{W}_i over a period of time

Theorem 1. Consider a closed-loop system with (12) as the plant, which uses (14) as the reference model, (15) as the controller, and (16) as the weight update rule. For a given reference signal y_i , we have the following conclusions:

- (1) All signals in the closed-loop system remain bounded.
- (2) The state tracking error $z_i = x_i - y_i$ can exponentially converge to a small neighbourhood of zero when selecting appropriate design parameters. Also, the weight estimator \widehat{W}_i can also exponentially converge to the vicinity of the optimal weight W .
- (3) A locally accurate estimation $S(x_i)^T \overline{W}$ for the reference trajectory can be obtained, and the error ϵ can meet the desired level.

Proof. we use the following three parts to prove the results.

- (1) For $z_i = \widehat{x}_i - y_i$, the derivative is

$$\dot{z}_i = \dot{\widehat{x}}_i - \dot{y}_i = f(x_i) + u_i - \dot{\alpha}_i. \quad (18)$$

Substituting formula (15) into the abovementioned formula, we obtain

$$\dot{z}_i = B_i z_i + S(x_i)^T \widehat{W}_i - \varepsilon_i. \quad (19)$$

Using the estimated z_i of the system state and the weight update rule (17), we construct the following Lyapunov function:

$$V = \sum_{i=1}^N \frac{1}{2} z_i^T z_i + \sum_{i=1}^N \frac{1}{2 \Psi_i} \widetilde{W}_i^T \Psi_i \widetilde{W}_i. \quad (20)$$

defined that $\widetilde{W} = [\widetilde{W}_1^T, \widetilde{W}_2^T, \widetilde{W}_3^T, \dots, \widetilde{W}_N^T]$. Taking the derivative of the above Lyapunov function, we obtain

$$\dot{V} = \sum_{i=1}^N (z_i^T B_i z_i - z_i^T \varepsilon_i) - \sum_{i=1}^N \sigma_i \widetilde{W}_i^T \widehat{W}_i - \Theta_i \widetilde{W}^T (\mathcal{L} \otimes I_n) \widetilde{W}. \quad (21)$$

For the abovementioned formula, it is known from Lemma 1 that, for the Laplacian matrix \mathcal{L} in this case, there is one and only one zero eigenvalue whose eigenvector is $(1/\sqrt{N})1_n$, and the other eigenvalues are all positive numbers. For these $N-1$ eigenvalues, we express them as

$$0 < \kappa_2 \leq \kappa_3 \cdots \leq \kappa_N. \quad (22)$$

Then, for $\mathcal{L} \otimes I_{nl}$, it has nl zero eigenvalues, and the eigenvectors of these eigenvalues can be expressed as

$$t_1 = \frac{1}{\sqrt{N}}1_n \otimes e_1, t_2 = \frac{1}{\sqrt{N}}1_n \otimes e_2, \dots, t_v = \frac{1}{\sqrt{N}}1_n \otimes e_{nl}. \quad (23)$$

For the other $(N-1)nl$ nonzero eigenvalues in the formula, we express their eigenvectors as

$$t_{nl+1}, t_{nl+2}, \dots, t_{Nnl}. \quad (24)$$

Let $Q = (t_1, t_2, \dots, t_{Nnl})$, $\Lambda = \text{diag}\{\kappa_1 I_{nl}, \kappa_2 I_{nl}, \dots, \kappa_N I_{nl}\}$, and $D = Q^T \Lambda^{-1} Q$. Then, the following formula can be obtained:

$$\tilde{W}^T (\mathcal{L} \otimes I_{nl}) \tilde{W} = e^T D e, \quad (25)$$

where $e = (\mathcal{L} \otimes I_{nl}) \tilde{W}$.

In addition, assuming that $b = \min(b_{11}, b_{12}, \dots, b_{Nn})$, the following inequalities can be obtained:

$$\begin{aligned} z_i^T B_i z_i &\leq -b z_i^T z_i, \\ -z_i^T \varepsilon_i &\leq \frac{b}{2} z_i^T z_i + \frac{1}{2b} \varepsilon^2, \\ -\sum_{i=1}^N \sigma_i \tilde{W}_i^T \hat{W}_i &\leq -\frac{\bar{\sigma}}{2} \tilde{W}_i^T \hat{W}_i + \sum_{i=1}^N \frac{\sigma_i}{2} W^2, \\ -e^T D e &\leq -\frac{1}{\kappa_N} \|e\|^2 = -\frac{1}{\kappa_N} \tilde{W}^T (\mathcal{L} \otimes I_{nl})^2 \tilde{W}. \end{aligned} \quad (26)$$

Since Ψ_i is a designed matrix, we can make it a symmetric positive definite matrix. Then, there exists a positive definite matrix Φ that satisfies $\Phi \Phi^T = \Psi^{-1}$, and $\Psi^{-1} = \text{diag}\{\Psi_1^{-1}, \Psi_2^{-1}, \dots, \Psi_N^{-1}\}$. Using the abovementioned inequality, V can be reduced to

$$\dot{V} \leq -\frac{b}{2} \sum_{i=1}^N z_i^T z_i - \kappa_{\min}(\bar{\Phi}) \sum_{i=1}^N \tilde{W}_i^T \Psi_i^{-1} \hat{W}_i + \frac{1}{2b} N \varepsilon^2 + \frac{\sigma_i}{2} \sum_{i=1}^N \|W\|^2, \quad (27)$$

that is,

$$\dot{V} \leq -\rho V + \delta, \quad (28)$$

where

$$\bar{\Phi} = \Phi^{-1} \left(\frac{\bar{\sigma}}{2} \right) I_{Nnl} + \frac{1}{\kappa_N} (\mathcal{L} \otimes I_{nl})^2 \Phi^{-1}, \quad (29)$$

$$\rho = \min \left\{ b, \kappa_{\min} \frac{\bar{\Phi}}{2} \right\},$$

$$\delta = \frac{1}{2b} N \varepsilon^2 + \frac{\sigma_i}{2} \sum_{i=1}^N \|W\|^2. \quad (30)$$

So, there is

$$\frac{\delta}{\rho} + V(0) \exp(-\rho t) > \frac{\delta}{\rho} + \left(V(0) - \frac{\delta}{\rho} \right) \exp(-\rho t) \geq V(t) \geq 0. \quad (31)$$

Through the abovementioned formula, it can be reflected that the constructed Lyapunov function is bounded. Then, the elements z_i and \tilde{W}_i that make up V are bounded. Furthermore, x_i and \hat{W}_i are bounded. Therefore, we can draw a conclusion that all signals in this closed-loop system are bounded.

(2) In this part, we will construct a special form to use the lemma to prove. From the previous definition, we know $Q = (t_1, t_2, \dots, t_{Nnl})$, so for \tilde{W} , it can be expressed as follows:

$$\tilde{W} = a_1 t_1 + a_2 t_2 + a_3 t_3 + \dots + a_{Nnl} t_{Nnl}, \quad (32)$$

where $a = (a_1, a_2, a_3, \dots, a_{Nnl})$ denotes the corresponding coefficient. To analyse \tilde{W} relatively simply, a space Γ is introduced, where $\Gamma = (t_1, t_2, \dots, t_{nl})$. Then, if $\tilde{W} \in \Gamma$ exists, there is $\tilde{W}_1 = \tilde{W}_2 = \dots = \tilde{W}_N$.

If the condition of formula (30) is not satisfied, we can express the distance between \tilde{W} and the space Γ as D , which satisfies

$$D = \min \|\tilde{W} - l\|^2 = \sum_{i=nl+1}^{Nnl} a_i^2 \leq \frac{1}{k_2^2} \|e\|^2. \quad (33)$$

Reviewing that $e = (\mathcal{L} \otimes I_{nl}) \tilde{W}$, since e can reach a small value, it can be seen from equation (36) that $\min \|\tilde{W} - l\|^2$ can also reach a small value. This means that there exists a small positive Ω such that $\hat{W}_i - \hat{W}_j \leq \Omega$, which implies that

$$\hat{W}_1 = \hat{W}_2 = \dots = \hat{W}_N. \quad (34)$$

At the same time, due to the localization property of the radial basis function, according to formula (11), we can get

$$\dot{z}_i = B_i z_i + S_{\varphi_i}(x_i, r_i)^T \hat{W}_{i, \varphi_i} + \hat{W}_{i, \bar{\varphi}} S_{\bar{\varphi}_i}(x_i, r_i) - \varepsilon_{i, \varphi_i}. \quad (35)$$

We define the tracking error along the reference trajectory $F_d(y_i)$ as $E = \hat{W}_{i, \bar{\varphi}} S_{\bar{\varphi}_i}(x_i, r_i) + \varepsilon_{i, \varphi_i}$, and equation (35) can be rewritten as

$$\dot{z}_i = B_i z_i + S_{\varphi_i}(x_i, r_i)^T \widehat{W}_{i, \varphi_i} - E. \quad (36)$$

Using the form of a matrix, this closed-loop system can be expressed as

$$\begin{bmatrix} \dot{z}_i \\ \dot{\widehat{W}}_{i, \varphi_i} \end{bmatrix} = \begin{bmatrix} B_i & S_{\varphi_i}(x_i, r_i)^T \\ -\Psi_{i, \varphi_i} S_{\varphi_i}(x_i, r_i) & 0 \end{bmatrix} \begin{bmatrix} z_i \\ \widehat{W}_{i, \varphi_i} \end{bmatrix} - \begin{bmatrix} E \\ \sigma_i \Psi_{i, \varphi_i} \widehat{W}_{i, \varphi_i} + \Theta \Psi_{i, \varphi_i} \sum_{j=1}^N a_{i,j} (\widehat{W}_{i, \varphi_i} - \widehat{W}_{j, \varphi_i}) \end{bmatrix}. \quad (37)$$

Then, the weight error can be expressed as the following formula:

$$\dot{\widehat{W}}_{i, \varphi} = -\Psi_{i, \varphi} S_{\varphi_i}(x_i, r_i) z_i - \sigma_i \Psi_{i, \varphi} \widehat{W}_{i, \varphi} - \Theta \Psi_{i, \varphi} \sum_{j=1}^N a_{i,j} (\widehat{W}_{i, \varphi} - \widehat{W}_{j, \varphi}). \quad (38)$$

Since σ_i can be selected as a numerically small value, E and $\sigma_i \Psi_{i, \varphi_i} \widehat{W}_{i, \varphi_i}$ can also be a small value. In addition, since the final result of $\widehat{W}_i - \widehat{W}_j = \widehat{W}_i - \widehat{W}_j$ can also reach a small value, $\Theta \Psi_{i, \varphi_i} \sum_{j=1}^N a_{i,j} (\widehat{W}_{i, \varphi_i} - \widehat{W}_{j, \varphi_i})$ can reach a small value. At this time, from Lemma 4, we know that, for a given reference trajectory $F_d(y_i)$, $S_{\varphi_i}(x_i, r_i)$ almost always meets the PE condition. With Lemma 2, the origin $(z_i, \widehat{W}_{i, \varphi_i}) = 0$ in system (36) is exponentially stable. Then, according to Lemma 3, we can know that, for this system, the tracking error z_i and the neural network weight estimation error both converge to zero exponentially. The degree of convergence is related to the design parameters. Based on the abovementioned analysis, it is also shown that since the weight error value \widehat{W}_i converges to zero exponentially, the weight estimation value \widehat{W}_i all converges to the optimal weight W .

- (3) We can prove this part according to the RBF NN weight update rule proposed by DCL. Based on the abovementioned definition, the weights of neurons

far away from the target trajectory can be described as follows:

$$\dot{\widehat{W}}_{i, \varphi_i} = -\Psi_{i, \varphi_i} [S_{i, \varphi_i}(x_i, r_i) z_i + \sigma_i \widehat{W}_{i, \varphi_i}] - \Theta \sum_{j=1}^N a_{i,j} (\widehat{W}_{i, \varphi_i} - \widehat{W}_{j, \varphi_i}). \quad (39)$$

Since the radial basis function can finally approximate the unknown nonlinear function with arbitrary accuracy, $S_{\varphi_i}(x_i, r_i)$ could be a very small value. The σ_i in the σ correction term is usually also a small value. Since \widehat{W}_i and \widehat{W}_j are assumed to be zero at the initial condition in this paper, the update amplitude of $\widehat{W}_{i, \varphi_i}$ is small and it is always maintaining a small value. Then, this can prove that $S_{i, \varphi_i}(x_i, r_i) \widehat{W}_{i, \varphi_i}$, $S_{i, \varphi_i}(x_i, r_i) \overline{W}_{i, \varphi_i}$, $S_{i, \varphi_i}(F_d) \widehat{W}_{i, \varphi_i}$, and $S_{i, \varphi_i}(F_d) \overline{W}_{i, \varphi_i}$ are all small values.

Through RBF NN, the estimation of the nonlinear function $f(x)$ in the input state can be expressed in the following form:

$$\begin{aligned} f(F_d) &= S_{\varphi}(F_d)^T W_{\varphi} + \varepsilon_{\varphi} = S_{\varphi}(F_d)^T \widehat{W}_{i, \varphi_i} + \varepsilon_{i, \varphi_i} = S_{\varphi}(F_d)^T \widehat{W}_{i, \varphi_i} + \varepsilon_{i, \varphi_i}, \\ &= S_{\varphi}(F_d)^T \widehat{W}_{i, \varphi_i} + S_{\varphi}(F_d)^T \widehat{W}_{i, \varphi_i} + \varepsilon_{i, \varphi_i} - S_{i, \varphi_i}(F_d)^T \widehat{W}_{i, \varphi_i} = S_{\varphi}(F_d)^T \widehat{W}_i + \varepsilon_{i_1}, \\ &= S_{\varphi}(F_d)^T \overline{W}_{i, \varphi_i} + \varepsilon_{i, \varphi_2} = S_{\varphi}(F_d)^T \overline{W}_{i, \varphi_i} + S_{\varphi}(F_d)^T \overline{W}_{i, \varphi_i} + \varepsilon_{i, \varphi_i} - S_{\varphi}(F_d)^T \overline{W}_{i, \varphi_i} = S(F_d)^T \overline{W}_i + \varepsilon_{i_2}, \end{aligned} \quad (40)$$

where

$$\begin{aligned} \varepsilon_{i, \varphi_i} &= \varepsilon_{\varphi} - S_{\varphi}(F_d)^T \widehat{W}_{i, \varphi_i}, \\ \varepsilon_{i_1} &= \varepsilon_{i, \varphi_i} - S_{\varphi}(F_d)^T \widehat{W}_{i, \varphi_i}, \\ \varepsilon_{i_2} &= \varepsilon_{i, \varphi_i} - S_{\varphi}(F_d)^T \overline{W}_{i, \varphi_i}, \end{aligned} \quad (41)$$

and $\varepsilon_{i, \varphi_2}$ represents the inherent estimation error of the radial basis function neural network, which is considered as

a small value. To sum up, $S(x_i)^T \overline{W}$ can estimate the unknown nonlinear function $f(x)$ along the known trajectory F_d and ensure that the estimation error value is a very small number. \square

4. Simulation

In order to prove the effectiveness of the method proposed in this paper, the following simulation is implemented. In the

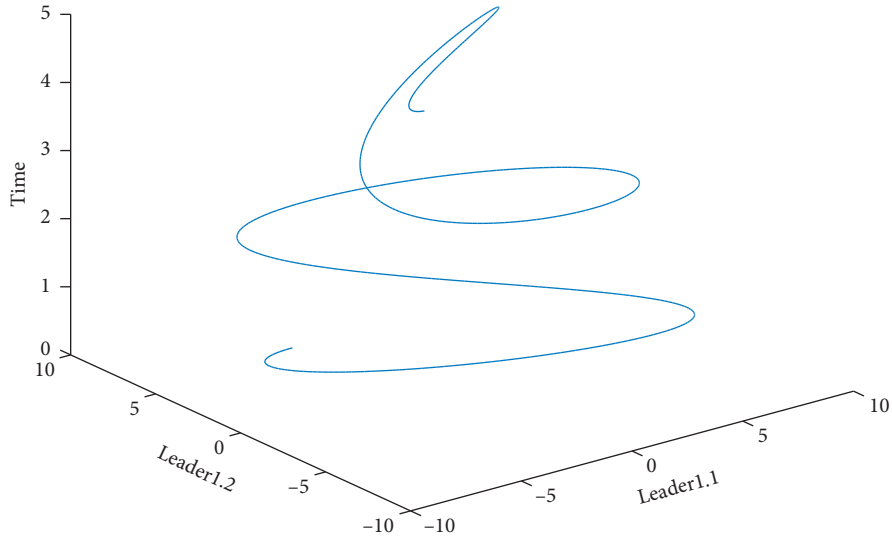


FIGURE 1: Reference trajectory of the leader.

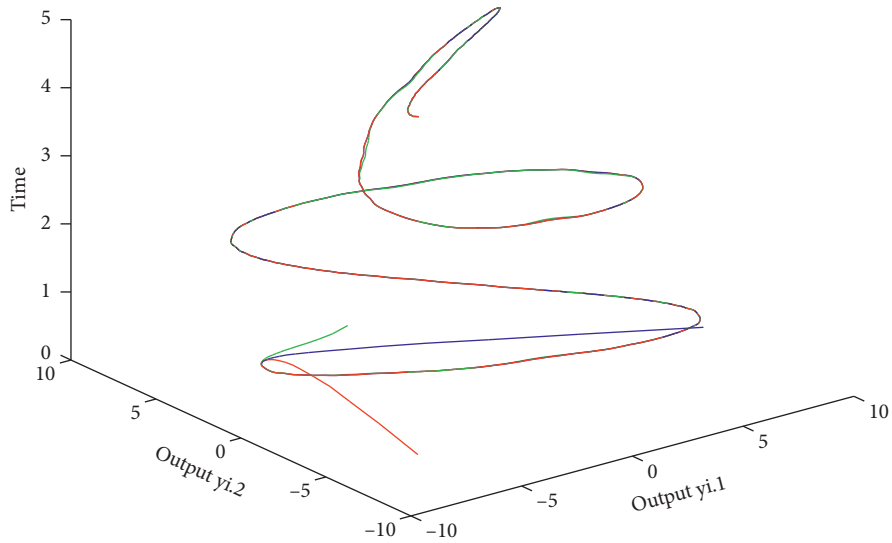


FIGURE 2: Consensus performance of three states following the leader.

simulation process, we consider a multiagent system constructed by three agents, and the dynamic equation of each multiagent is as follows:

$$\dot{x}_i = a_i \begin{pmatrix} x_{i,1} e^{-x_{i,2}^2} \\ x_{i,2} e^{-x_{i,1}^2} \end{pmatrix} + \int u_i + b_i \begin{pmatrix} \|x_i\|^2 \sin(\|x_i\|) \\ \|x_i\| \cos(\|x_i\|^3) \end{pmatrix}, \quad y_i = x_i, \quad i = 1, 2, 3. \quad (42)$$

The definitions of a_i, b_i are $a_i = (-1.5, -1.6, 2.3)$ and $b_i = (2, 0.9, 1.4)$, respectively, where $i = 1, 2, 3$. The dynamic equation of the leader is described as

$$F_d = (6 \sin(4t + 10), 6 \cos(3t + 0.3))^T. \quad (43)$$

In this example, the RBF NN model used is shown in equation (16). In this RBF NN, assume that there are 36 nodes in total, which are finally distributed in the space of $[-6, 6] \times [-6, 6]$. The basis function vector is represented as follows:

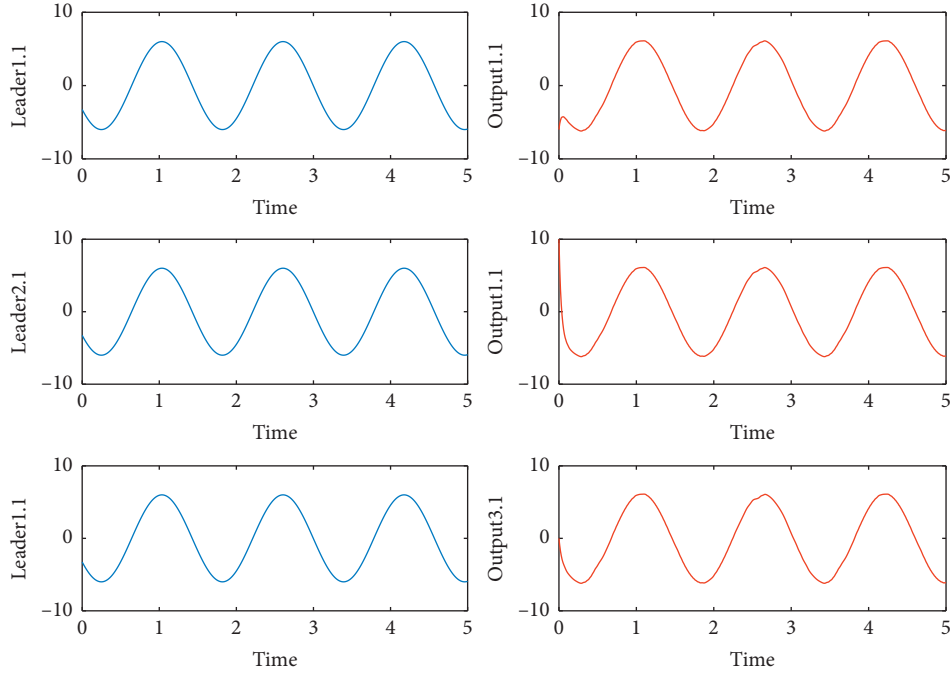


FIGURE 3: Performance of the first coordinate of the leader and three agents.

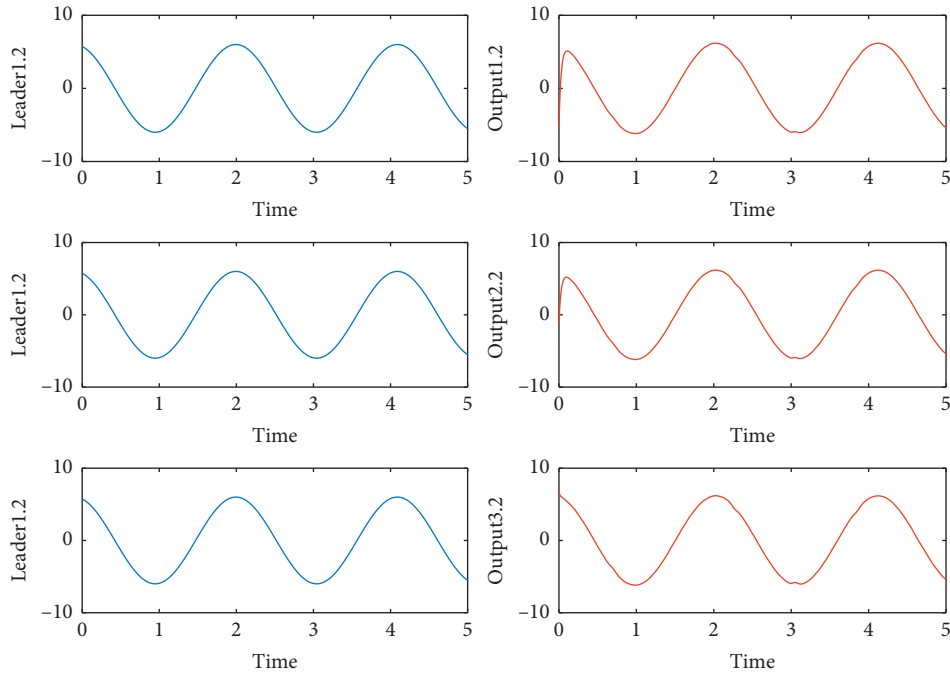


FIGURE 4: Performance of the second coordinate of the leader and three agents.

$$S_{ij}(z_i) = [s_1^{ij}(z_1), s_2^{ij}(z_2), \dots, s_{36}^{ij}(z_{36})]^T, \quad (44)$$

$$s_l^{ij} = \exp\left[-\frac{\|((z_l) - c_l)\|^2}{\eta^2}\right],$$

where width $\eta = 2, l = 1, 2, \dots, 36$. The design parameters in this simulation are $B_i = \text{diag}\{-30, -30\}, \Psi_i = \text{diag}\{0.1\}$,

$\sigma_i = 0.001, \Theta_i = 2$. We define the initial conditions as $\bar{W}_i(0) = 0, \hat{x}_i(0) = 0, \hat{x}_{11}(0) = (-6, 3^{-0.5})^T, \hat{x}_{21}(0) = (10, -4^{0.1})^T, \hat{x}_{31}(0) = (0, 5e^{0.3})^T$. By applying the controller u_i proposed in this paper to the abovementioned initial state, the system can be controlled.

Figures 1 and 2, respectively, represent the leader's reference signal and the state changes of the three agents after being controlled by the controller proposed in this

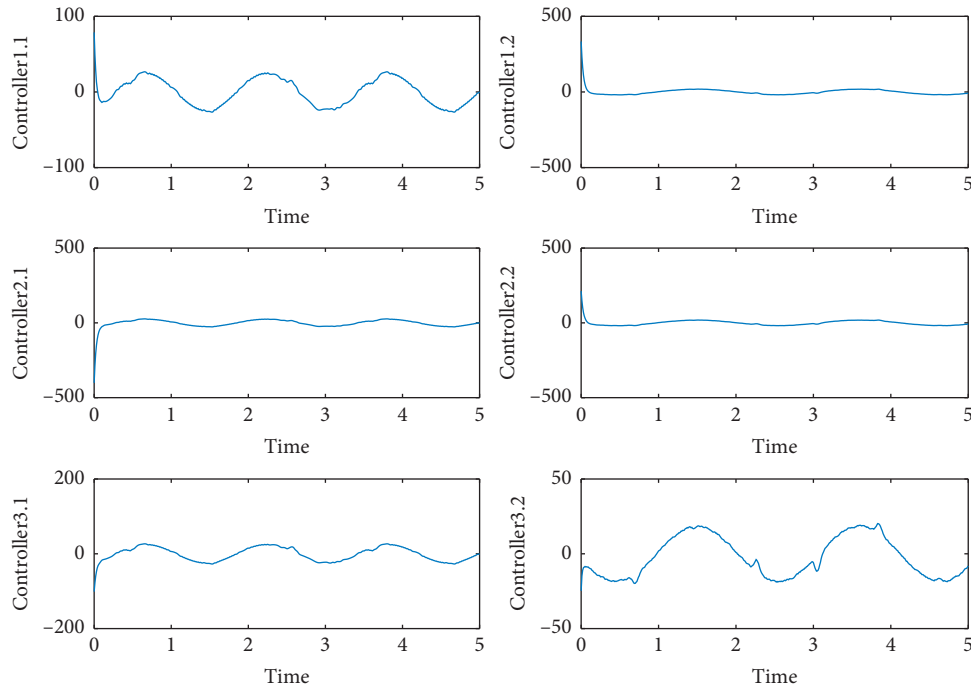


FIGURE 5: The first and second coordinates of the three controllers.

paper. From Figure 2, we can see that the consensus tracking problem has been well resolved. Figures 3 and 4 show that each state can well track a given reference trajectory at different times. Figure 5 shows that the controller converges to zero over time, reflecting the effectiveness of the control. Figures 1–5 show that all signals in this multiagent system are bounded.

5. Direction to Be Improved

The premise of this paper is that the movement trajectory of the leader is known, which indicates that we do not need to discuss its unknown state. However, in the practical application, the leader's state is often unmeasurable, which requires us to measure its unknown state. [31–36] discuss a variety of methods for estimating unknown states and eliminating estimation errors. Similar methods mentioned above are used to predict the movement state of the leader, so as to realize the control of the completely unknown system, which will be reflected in the later work.

6. Conclusions

Inspired by consensus theory and deterministic learning theory, this paper proposes a distributed adaptive learning control scheme for a kind of unknown nonlinear systems. In order to solve the unmeasured state that may exist in the system, this paper uses the approximation characteristics of the radial basis function for the unknown nonlinear system and proposes a novel weight update method. At the same time, by using Lyapunov's stability theory, it is proved that the control scheme proposed in this paper can ensure that all signals remain uniformly bounded and the tracking error

converges to zero. The simulation proves the effectiveness of the control strategy proposed in this paper.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

- [1] C. L. P. Chen, G.-X. Wen, Y.-J. Liu, and F.-Y. Wang, "Adaptive consensus control for a class of nonlinear multi-agent time-delay systems using neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 6, pp. 1217–1226, 2014.
- [2] M. Chen, S. S. Ge, and B. V. E. How, "Robust adaptive neural network control for a class of uncertain MIMO nonlinear systems with input nonlinearities," *IEEE Transactions on Neural Networks*, vol. 21, no. 5, pp. 796–812, 2010.
- [3] S. S. Ge, F. Hong, and T. H. Lee, "Adaptive neural control of nonlinear time-delay systems with unknown virtual control coefficients," *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 34, no. 1, pp. 499–516, 2004.
- [4] W. He, Y. Chen, and Z. Yin, "Adaptive neural network control of an uncertain robot with full-state constraints," *IEEE Transactions on Cybernetics*, vol. 46, no. 3, pp. 620–629, 2016.
- [5] J. Jang, "ANFIS: adaptive-network-based fuzzy inference system," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 23, no. 3, pp. 665–685, 1993.
- [6] Y. J. Liu, C. L. Chen, G. X. Wen, and S. Tong, "Adaptive neural output feedback tracking control for a class of uncertain

- discrete-time nonlinear systems,” *IEEE Transactions on Neural Networks*, vol. 22, no. 7, pp. 1162–1167, 2011.
- [7] D. Wang and J. Huang, “Neural network-based adaptive dynamic surface control for a class of uncertain nonlinear systems in strict-feedback form,” *IEEE Transactions on Neural Networks*, vol. 16, no. 1, pp. 195–202, 2005.
- [8] B. Widrow and M. A. Lehr, “30 years of adaptive neural networks: perceptron, madaline, and backpropagation,” *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1415–1442, 1990.
- [9] T. Zhang, S. S. Ge, and C. C. Hang, “Adaptive neural network control for strict-feedback nonlinear systems using backstepping design,” *Automatica*, vol. 36, no. 12, pp. 1835–1846, 2000.
- [10] C. Li, Z. Dong, G. Chen, B. Zhou, and X. Yu, “Data-driven planning of electric vehicle charging infrastructure,” *A Case Study of Sydney, Australia*, vol. 99, p. 1, 2021.
- [11] C. Li, X. Yu, T. Huang, G. Chen, and X. He, “A generalized hopfield network for nonsmooth constrained convex optimization: lie derivative approach,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 2, pp. 308–321, 2016.
- [12] A. Wu and Z. Zeng, “Output convergence of fuzzy neurodynamic system with piecewise constant argument of generalized type and time-varying input,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 12, pp. 1689–1702, 2016.
- [13] A. Wu and Z. Zeng, “Exponential stabilization of memristive neural networks with time delays,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 12, pp. 1919–1929, 2012.
- [14] A. Wu and Z. Zeng, “Lagrange stability of memristive neural networks with discrete and distributed delays,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 4, pp. 690–703, 2014.
- [15] A. Wu and Z. Zeng, “Global mittag-leffler stabilization of fractional-order memristive neural networks,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 1, pp. 206–217, 2017.
- [16] H. Li, Y. Wu, and M. Chen, “Adaptive fault-tolerant tracking control for discrete-time multiagent systems via reinforcement learning algorithm,” *IEEE Transactions on Cybernetics*, vol. 51, no. 3, pp. 1–12, 2020.
- [17] H. Ma, H. Y. Li, R. Q. Lu, and T. W. Huang, “Adaptive event-triggered control for a class of nonlinear systems with periodic disturbances,” *Science China Information Sciences*, vol. 63, no. 5, Article ID 150212, 2020.
- [18] B. Luo, Y. Yang, and D. Liu, “Adaptive Q-learning for data-based optimal output regulation with experience replay,” *IEEE Transactions on Cybernetics*, vol. 48, no. 12, pp. 3337–3348, 2018.
- [19] D. Shanling, C. GuanRong, L. Meiqin, and W. Zheng-Guang, “Cooperative Neural-Adaptive Fault-Tolerant Output Regulation for Heterogeneous Nonlinear Uncertain Multiagent Systems with Disturbance,” *Science China Information Sciences*, no. 1674-733X, 2017.
- [20] W. Chen, S. Hua, and S. Sam Ge, “Consensus-based distributed cooperative learning control for a group of discrete-time nonlinear multi-agent systems using neural networks,” *Automatica*, vol. 50, no. 9, pp. 2254–2268, 2014.
- [21] C. Wang and D. J. Hill, “Learning from neural control,” *IEEE Transactions on Neural Networks*, vol. 17, no. 1, pp. 130–146, 2006.
- [22] J. Xie, W. Chen, and H. Dai, “Distributed cooperative learning algorithms using wavelet neural network,” *Neural Computing and Applications*, vol. 31, no. 4, pp. 1007–1021, 2019.
- [23] Z. Zhou, D. Tong, Q. Chen, W. Zhou, and Y. Xu, “Adaptive NN control for nonlinear systems with uncertainty based on dynamic surface control q,” *Neurocomputing*, vol. 421, pp. 161–172, 2021.
- [24] J. Park and I. W. Sandberg, “Universal approximation using radial-basis-function networks,” *Neural Computation*, vol. 3, no. 2, pp. 246–257, 1991.
- [25] C. Wang, “Deterministic learning theory for identification, recognition, and control,” *Neurocomputing*, vol. 358, pp. 456–466, 2019.
- [26] R. E. Slavin, “Research on cooperative learning and achievement: what we know, what we need to know,” *Contemporary Educational Psychology*, vol. 21, no. 1, pp. 43–69, 1996.
- [27] W. Chen, S. Hua, W. Ren, and W. Hu, “Neural-network-based cooperative adaptive identification of nonlinear systems,” in *Proceedings of the International Conference on Control Automation Robotics & Vision*, Shenzhen, China, December 2013.
- [28] A. J. Laub, “Matrix analysis for scientists and engineers,” *IEEE Control Systems Magazine*, vol. 26, no. 2, pp. 94–95, 2005.
- [29] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*, New Jersey, NJ, USA, 2010, <https://www.jstor.org/publisher/princetonup>.
- [30] M. Bodson, *Adaptive Control: Stability, Convergence, and Robustness*, Prentice-Hall, New Jersey, NJ, USA, 2011.
- [31] C. Li, X. Yu, T. Huang, and X. He, “Distributed optimal consensus over resource allocation network and its application to dynamical economic dispatch,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 6, pp. 2407–2418, 2018.
- [32] A. Wu, H. Liu, and Z. Zeng, “Observer Design and H_{∞} performance for discrete-time uncertain fuzzy-logic systems,” *IEEE Transactions on Cybernetics*, 2019.
- [33] B. Zhang, C. Yang, H. Zhu, P. Shi, and W. Gui, “Controllable-domain-based fuzzy rule extraction for copper removal process control,” *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 3, pp. 1744–1756, 2018.
- [34] Y. Xu, M. Fang, Y. J. Pan, K. B. Shi, and Z. G. Wu, “Event-triggered output synchronization for nonhomogeneous agent systems with periodic denial-of-service attacks,” *International Journal of Robust and Nonlinear Control*, vol. 15, 2020.
- [35] Y. Xu and Z.-G. Wu, “Distributed adaptive event-triggered fault-tolerant synchronization for multiagent systems,” *IEEE Transactions on Industrial Electronics*, vol. 68, no. 2, pp. 1537–1547, 2021.
- [36] Y. Xu, J. Zhou, H. Rao, R. Lu, and L. Xie, “Reset moving horizon estimation for quantized discrete time systems,” *IEEE Transactions on Automatic Control*, vol. 99, p. 1, 2020.