Distributed Control for Nonlinear Time-Delayed Multi-Agent Systems with Connectivity Preservation Using Neural Networks

Hongwen Ma[1], Derong Liu[2](✉), and Ding Wang[1]

[1] The State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
{mahongwen2012,ding.wang}@ia.ac.cn

[2] School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China
derong@ustb.edu.cn

Abstract. Nonlinear time-delayed multi-agent systems with connectivity preservation are investigated in this paper. For each agent, the distributed controller is divided into five different parts which are designed to meet the requirements of the nonlinear time-delayed multi-agent systems, such as preserving connectivity, learning the unknown dynamics, eliminating time delays and reaching consensus. In addition, a σ-function technique is utilized to avoid the singularity in the developed distributed controller. Finally, simulation results demonstrate the effectiveness of the developed control protocol.

Keywords: Distributed control · Connectivity preservation · Nonlinear multi-agent systems · Neural networks · Time-delay

1 Introduction

With the growth of scale in large networked multi-agent systems, it is difficult to design an appropriate centralized controller to maintain the performance of the whole systems. Thus, distributed control is a good choice for solving this problem. Distributed control for multi-agent systems has been a hot topic in the last decade [1–3]. In [4], an output-based distributed control was proposed for nonlinear multi-agent systems with small-cyclic theorem. In [5], an online optimal learning approach was added to the decentralized control for a class of continuous-time nonlinear interconnected systems. To the best of authors' knowledge, it is the first time to investigate second-order nonlinear time-delayed multi-agent systems with connectivity preservation. In [6], in order to drive a group of ocean vessels to track a moving target and maintain the connectivity, adaptive neural network region tracking control was proposed. In [7], by virtue of neural networks, a decentralized robust adaptive control was designed to achieve consensus. In [8], a rendezvous protocol was proposed for the double-integrator multi-agent systems with preserved network connectivity. However, none of them

S. Arik et al. (Eds.): ICONIP 2015, Part III, LNCS 9491, pp. 34–42, 2015.
DOI: 10.1007/978-3-319-26555-1_5

takes time-delay into consideration. Thus, in this paper a Lyapunov-Krasovskii functional method is borrowed from [9,10] to eliminate the negative effect of time delays. Moreover, a σ-function is developed to circumvent the singularity in the distributed controller.

The remainder of this paper is given as follows. In Sect. 2, fundamental preliminaries and the problem statement are introduced. In Sect. 3, the distributed control protocol is developed which guarantees the achievement of consensus. Simulation example and conclusion are given in Sects. 4 and 5, respectively.

Notations: $(\cdot)^{\mathsf{T}}$ represents the transpose of a matrix. $\mathrm{tr}\,(\cdot)$ is the trace of a given matrix and $\|\cdot\|$ is the Frobenius norm or Euclidian norm.

2 Preliminaries

2.1 Graph Theory

A triplet $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{A}\}$ is called a weighted graph if $\mathcal{V} = \{1, 2, \ldots, N\}$ is the set of N nodes, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges, and $\mathcal{A} = (\mathcal{A}_{ij}) \in \mathbb{R}^{N\times N}$ is the $N\times N$ matrix of the weights of $\mathcal{G}$. Here we denote $\mathcal{A}_{ij}$ as the element of the ith row and jth column of the matrix $\mathcal{A}$. The ith node in graph $\mathcal{G}$ represents the ith agent, and a directed path from node i to node j is denoted as an ordered pair $(i, j) \in \mathcal{E}$, which means that agent i can directly transfer its information to agent j. $\mathcal{L} = \mathcal{D} - \mathcal{A}$ is the Laplacian matrix, where $\mathcal{D}$ is the $N\times N$ diagonal matrix whose diagonal elements are $d_i = \sum_{j\in\mathcal{N}_i} \mathcal{A}_{ij}$, $i = 1, 2, \ldots, N$ and $\mathcal{N}_i = \{j \in \mathcal{V} | (j, i) \in \mathcal{E}\}$ is the set of neighbour nodes of node i.

2.2 Radial Basis Function Neural Network

In this paper, radial basis function neural networks (RBFNNs) are used for approximating the unknown dynamics of the multi-agent systems. If $h(x)$ is a continuous unknown nonlinear function, then it can be approximated by RBFNNs as follows:

$$h(x) = W^{*\mathsf{T}}\Phi(x) + \theta, \tag{1}$$

where x is the input vector, W^* is the ideal weight matrix with suitable dimensions and θ is the approximating error with $\|\theta\| < \theta_N$. $\Phi(x) = [\gamma_1(x), \gamma_2(x), \ldots, \gamma_p(x)]^{\mathsf{T}}$ is the activation function vector and

$$\gamma_i(x) = \exp\left[\frac{-(x-\mu_i)^{\mathsf{T}}(x-\mu_i)}{\alpha_i^2}\right], \quad i = 1, 2, \ldots, p, \tag{2}$$

where α_i is the width of Gaussian function, p is the number of neurons and μ_i is the center of the receptive field. We denote $\hat{W}$ as the estimation of the ideal weight matrix W^*. Thus, the estimation of $h(x)$ can be written as $\hat{h}(x) = \hat{W}^{\mathsf{T}}\Phi(x)$, where $\hat{W}$ can be updated online. The online updating algorithm is provided in Sect. 3.

2.3 Problem Statement

In this paper, the second-order nonlinear time-delayed multi-agent system is modeled as follows:

$$\begin{aligned} &\dot{p}_i = v_i, \\ &\dot{v}_i = u_i + f_i(p_i(t), v_i(t)) + g_i(v_i(t-\tau_i)), \quad i = 1, 2, \ldots, N, \end{aligned} \tag{3}$$

where $p_i \in \mathbb{R}^2$ is the position of agent i, $v_i \in \mathbb{R}^2$ is the velocity of agent i, τ_i is the unknown time delay of agent i, $f_i(\cdot)\colon \mathbb{R}^2 \to \mathbb{R}^2$ and $g_i(\cdot)\colon \mathbb{R}^2 \to \mathbb{R}^2$ are continuous but unknown nonlinear vector functions, and $u_i(\cdot) \in \mathbb{R}^2$ is the control input. For simplicity, we will ignore time expression t in case there is no confusion.

Assume that all the agents have a common sensing radius R and we adopt the hysteresis function in [8] to avoid measurement noise. When the distance between two agents is greater than R, we say that the two agents lose connectivity. Our control objective is to make all the agents reach consensus with connectivity preservation. That is, $\forall i, j \in \mathcal{V}$,

$$\begin{cases} \lim\limits_{t\to\infty} \|p_i(t) - p_j(t)\| = 0, \\ \lim\limits_{t\to\infty} v_i(t) = v_j(t) = 0, \end{cases} \tag{4}$$

and no agent will lose connection with its neighbors. We adopt the definition of the potential function in [8] which is given as follows:

$$\varphi(\|p_{ij}\|) = \frac{\|p_{ij}\|^2}{R - \|p_{ij}\| + \dfrac{R^2}{\hat{P}}}, \quad \|p_{ij}\| \in [0, R], \tag{5}$$

where R is the radius of communication range, $\|p_{ij}\| = \|p_i(t) - p_j(t)\|$ and $\hat{P} > 0$ is a large constant. It should be noted that we utilize $\mathcal{A}(t)$, $\mathcal{N}(t)$ and $\mathcal{L}(t)$ to represent the switching topology.

3 Distributed Control for Nonlinear Time-Delayed Multi-Agent Systems

Before proceeding, we introduce two important assumptions for demonstrating our main theorem.

Assumption 1. *$g_i(v_i(t-\tau_i)), i = 1, 2, \ldots, N$, are unknown smooth nonlinear functions. The inequalities $\|g_i(v_i(t))\| \le \phi_i(v_i(t)), i = 1, 2, \ldots, N$, hold, where $\phi_i(\cdot), i = 1, 2, \ldots, N$, are known positive smooth scalar functions. Furthermore, $g_i(0) = 0$ and $\phi_i(0) = 0$, $i = 1, 2, \ldots, N$.*

Assumption 2. *The unknown time delays $\tau_i, i = 1, 2, \ldots, N$, are bounded by a known constant $\tau_{\max}$, i.e., $\tau_i \le \tau_{\max}, i = 1, 2, \ldots, N$.*

In order to avoid the singularity induced by the denominator of the developed distributed controller, we define $\sigma(\cdot)$ as follows:

$$\sigma(v_i) = \begin{cases} 1, \text{ if } v_i = 0, \\ 0, \text{ if } v_i \neq 0. \end{cases} \tag{6}$$

In order to eliminate the effect of time delays, we introduce a Lyapunov-Krasovskii functional as follows:

$$V_U(t) = \frac{1}{2}\sum_{i=1}^{N}\int_{t-\tau_i}^{t} U_i(v_i(\zeta))\mathrm{d}\zeta, \tag{7}$$

where $U_i(v_i(t)) = \phi_i^2(v_i(t))$. Then, the developed distributed controller is divided into five parts and they are given as follows:

$$\begin{aligned} u_i(t) &= u_{i1}(t) + u_{i2}(t) + u_{i3}(t) + u_{i4}(t) + u_{i5}(t), \qquad (8)\\ u_{i1}(t) &= -\sum_{j\in\mathcal{N}_i(t)} \nabla_{p_i}\varphi(\|p_{ij}\|),\\ u_{i2}(t) &= -\sum_{j\in\mathcal{N}_i(t)} \mathcal{A}_{ij}(t)(v_i - v_j),\\ u_{i3}(t) &= -\frac{1}{2}\frac{v_i}{\|v_i\|^2 + \sigma(v_i)}\phi_i^2(v_i(t)),\\ u_{i4}(t) &= -k_i(t)v_i,\\ u_{i5}(t) &= -\hat{W}_i^{\mathsf{T}}\varPhi_i(p_i, v_i), \end{aligned}$$

$$k_i(t) = k_{i0} + 1 + \frac{1}{\omega_i}\left(\frac{1}{2} + \frac{\int_{t-\tau_{\max}}^{t}\frac{1}{2}U_i(v_i(\zeta))\mathrm{d}\zeta}{\|v_i\|^2 + \sigma(v_i)} + \frac{\sum_{j\in\mathcal{N}_i(t)}\varphi(\|p_{ij}\|)}{\|v_i\|^2 + \sigma(v_i)}\right). \tag{9}$$

The online updating algorithm for the weight matrix of RBFNN is given as follows:

$$\dot{\hat{W}}_i = \begin{cases} \chi_i\varPhi_i(p_i, v_i)v_i^{\mathsf{T}}, \quad \text{if tr}\left(\hat{W}_i^{\mathsf{T}}\hat{W}_i\right) < W_i^{\max}, \text{ or} \\ \text{if tr}\left(\hat{W}_i^{\mathsf{T}}\hat{W}_i\right) = W_i^{\max} \text{ and } v_i^{\mathsf{T}}\hat{W}_i^{\mathsf{T}}\varPhi_i(p_i, v_i) < 0, \\ \chi_i\varPhi_i(p_i, v_i)v_i^{\mathsf{T}} - \chi_i\dfrac{v_i^{\mathsf{T}}\hat{W}_i^{\mathsf{T}}\varPhi_i(p_i, v_i)}{\text{tr}\left(\hat{W}_i^{\mathsf{T}}\hat{W}_i\right)}\hat{W}_i, \quad \text{otherwise}, \end{cases} \tag{10}$$

where $\tilde{W}_i = W_i^* - \hat{W}_i$ and χ_i is the updating rate. Moreover, the initial values of $\hat{W}_i$ should satisfy $\text{tr}\left(\hat{W}_i^{\mathsf{T}}(0)\hat{W}_i(0)\right) \leq W_i^{\max}$. Before proceeding, we define the potential energy function as follows:

$$P_i(t) = \sum_{j\in\mathcal{N}_i(t)}\varphi(\|p_{ij}\|) + \frac{1}{2}v_i^{\mathsf{T}}v_i + \frac{1}{2}\int_{t-\tau_i}^{t} U_i(v_i(\zeta))\mathrm{d}\zeta + \frac{1}{2}\text{tr}\left(\frac{1}{\chi_i}\tilde{W}_i^{\mathsf{T}}\tilde{W}_i\right). \tag{11}$$

Then, the total potential energy function is $P(t) = \sum_{i=1}^{N} P_i(t)$.

Theorem 1. *The multi-agent system (3) consists of N agents and all the agents are driven by the distributed controller (8). With Assumptions 1 and 2, if the initial network topology $\mathcal{G}(0)$ is connected and undirected, and the initial energy $P(0)$ is finite, then the consensus of the multi-agent system (3) can be achieved while preserving connectivity.*

Proof. The derivative of $P(t)$ is

$$\begin{aligned}
\dot{P}(t) =& \sum_{i=1}^{N} \sum_{j\in\mathcal{N}_i(t)} \dot{\varphi}(\|p_{ij}\|) + \sum_{i=1}^{N} v_i^\mathsf{T} \dot{v}_i + \dot{V}_U(t) - \sum_{i=1}^{N} \mathrm{tr}\left(\frac{1}{\chi_i}\tilde{W}_i^\mathsf{T}\dot{\hat{W}}_i\right) \\
=& \sum_{i=1}^{N} v_i^\mathsf{T} \sum_{j\in\mathcal{N}_i(t)} \nabla_{p_i}\varphi(\|p_{ij}\|) + \frac{1}{2}\sum_{i=1}^{N} \left(\phi_i^2(v_i(t)) - \phi_i^2(v_i(t-\tau_i))\right) \\
& + \sum_{i=1}^{N} v_i^\mathsf{T} \Bigg(- k_i(t)v_i - \sum_{j\in\mathcal{N}_i(t)} \nabla_{p_i}\varphi(\|p_{ij}\|) - \sum_{j\in\mathcal{N}_i(t)} \mathcal{A}_{ij}(t)(v_i - v_j) \\
& - \frac{1}{2}\frac{v_i}{\|v_i\|^2 + \sigma(v_i)}\phi_i^2(v_i(t)) - \hat{W}_i^\mathsf{T}\varPhi_i(\cdot) + g_i(v_i(t-\tau_i)) + W_i^{*\mathsf{T}}\varPhi_i(\cdot) + \theta_i \Bigg) \\
& - \sum_{i=1}^{N} \mathrm{tr}\left(\frac{1}{\chi_i}\tilde{W}_i^\mathsf{T}\dot{\hat{W}}_i\right),
\end{aligned} \tag{12}$$

where we denote $\varPhi_i(\cdot)$ as $\varPhi_i(p_i, v_i)$ for short. If $\mathrm{tr}\left(\hat{W}_i^\mathsf{T}(0)\hat{W}_i(0)\right) \le W_i^{\max}$, it is easy to demonstrate that $\mathrm{tr}\left(\hat{W}_i^\mathsf{T}(t)\hat{W}_i(t)\right) \le W_i^{\max}$. Thus, according to the updating algorithm (10), the inequality $\mathrm{tr}\left(\tilde{W}_i^\mathsf{T}\left(\frac{1}{\chi_i}\dot{\hat{W}}_i - \varPhi_i(\cdot)v_i^\mathsf{T}\right)\right) \ge 0$ holds. Merge the polynomial (12) we can obtain

$$\begin{aligned}
\dot{P}(t) =& \frac{1}{2}\sum_{i=1}^{N}(\phi_i^2(v_i(t)) - \phi_i^2(v_i(t-\tau_i))) + \sum_{i=1}^{N} \mathrm{tr}\left(\frac{1}{\chi_i}\tilde{W}_i^\mathsf{T}\dot{\hat{W}}_i\right) + \sum_{i=1}^{N} v_i^\mathsf{T}\Bigg(- k_i(t)v_i \\
& - \sum_{j\in\mathcal{N}_i(t)} \mathcal{A}_{ij}(t)(v_i - v_j) - \frac{1}{2}\frac{v_i}{\|v_i\|^2}\phi_i^2(v_i(t)) - \tilde{W}_i^\mathsf{T}\varPhi_i(\cdot) + g_i(v_i(t-\tau_i)) + \theta_i \Bigg) \\
=& - \sum_{i=1}^{N} k_i(t)\|v_i\|^2 - v^\mathsf{T}(\mathcal{L}(t)\otimes I_2)v - \sum_{i=1}^{N} \mathrm{tr}\left(\tilde{W}_i^\mathsf{T}\left(\frac{1}{\chi_i}\dot{\hat{W}}_i - \varPhi_i(\cdot)v_i^\mathsf{T}\right)\right) \\
& - \frac{1}{2}\sum_{i=1}^{N}\phi_i^2(v_i(t-\tau_i)) + \sum_{i=1}^{N} v_i^\mathsf{T}(g_i(v_i(t-\tau_i)) + \theta_i)
\end{aligned}$$

$$
\begin{aligned}
&\leq -\sum_{i=1}^{N} k_i(t)\|v_i\|^2 - \frac{1}{2}\sum_{i=1}^{N}\phi_i^2(v_i(t-\tau_i)) + \frac{1}{2}\sum_{i=1}^{N}\left(\|v_i\|^2 + \|g_i(v_i(t-\tau_i))\|^2\right) \\
&\quad + \frac{1}{2}\sum_{i=1}^{N}\left(\|v_i\|^2 + \|\theta_i\|^2\right) \quad \text{(with Assumption 1)} \\
&\leq -\sum_{i=1}^{N}(k_i(t)-1)\|v_i\|^2 + \epsilon,
\end{aligned}
$$

where $\epsilon = \frac{1}{2}\sum_{i=1}^{N}\theta_{N_i}^2$, $v = [v_1, v_2, \ldots, v_N]^{\mathsf{T}}$ and $\|\theta_i\| < \theta_{N_i}$. Thus, with Assumption 2, we have

$$
\begin{aligned}
\dot{P}(t) \leq &-\sum_{i=1}^{N} k_{i0}\|v_i\|^2 - \sum_{i=1}^{N}\frac{1}{\omega_i}\int_{t-\tau_{\max}}^{t}\frac{1}{2}U_i(v_i(\zeta))\mathrm{d}\zeta - \sum_{i=1}^{N}\frac{1}{2\omega_i}\|v_i\|^2 \\
&- \sum_{i=1}^{N}\frac{1}{\omega_i}\sum_{j\in\mathcal{N}_i(t)}\varphi(\|p_{ij}\|) - \sum_{i=1}^{N}\frac{2W_i^{\max}}{\omega\chi_i} + \sum_{i=1}^{N}\frac{2W_i^{\max}}{\omega\chi_i} + \theta \\
\leq &-\frac{1}{\omega}P(t) + \sum_{i=1}^{N}\frac{2W_i^{\max}}{\omega\chi_i} + \epsilon,
\end{aligned}
$$

where $\omega = \max_{i\in\mathcal{V}}\omega_i$ and $k_{i0} > 0$. Then, according to Lemma 1 in [7], we have

$$
P(t) \leq P(0)e^{-\frac{1}{\omega}t} + \nu\left(1 - e^{-\frac{1}{\omega}t}\right), \tag{13}
$$

where $\nu = \sum_{i=1}^{N}\frac{2W_i^{\max}}{\chi_i} + \omega\epsilon$.

The number of agents is finite, thus the switching times of the network topology are finite. We denote the switching times as $t_0, t_1, \ldots$, where t_0 is the initial time. By choosing appropriate parameters in (13) when $t \in [t_0, t_1)$, $P(t) \leq P(0) < P_{\max}$ holds. Therefore, network will not lose connectivity at t_1 and new edges will be added at t_1 because of the decrease of $P(t)$. Following the similar proof steps in the above analysis, when $t \in [t_{k-1}, t_k)$, $k = 2, 3, \ldots$, the connectivity can be guaranteed. In summary, if the initial undirected network topology $\mathcal{G}(0)$ is connected and the initial energy $P(0)$ is finite, the connectivity for $t > 0$ can be preserved. Then, we restrict the following discussion when the network has been fixed. Since every term in $P(t)$ is positive and bounded, all the terms in $P(t)$ will approach to zero, that is, $\lim_{t\to\infty} p_1 = p_2 = \cdots = p_N$ and $\lim_{t\to\infty} v_1 = v_2 = \cdots = v_N = 0$. Furthermore, $\lim_{t\to\infty}\|W_i^* - \hat{W}_i\| = 0$ shows that RBFNNs can learn the unknown dynamics of each agent. □

4 Simulation Example

We choose a multi-agent system with five agents which moves on a two-dimensional plane. We set the sensing radius $R = 2.5\,\mathrm{m}$ and choose the initial positions and velocities randomly from $[0, 4\,\mathrm{m}] \times [0, 4\,\mathrm{m}]$ and $[0, 2\,\mathrm{m/s}] \times [0, 2\,\mathrm{m/s}]$, respectively. Assume that all the existing communication weights are 1 and the time-delay vector is $\tau = [0.1, 0.05, 0.13, 0.08, 0.15]$. $\begin{bmatrix} l_1 \\ l_2 \end{bmatrix} = \begin{bmatrix} 0.4, & -0.65, & 0.5, & -0.75, & 0.1 \\ 0.5, & 0.45, & -0.6, & 0.4, & 1 \end{bmatrix}$ and $\begin{bmatrix} m_1 \\ m_2 \end{bmatrix} = \begin{bmatrix} 0.9, & 1.2, & -1.1, & 0.7, & 0.6 \\ 1.2, & -0.8, & 0.6, & 0.3, & 0.8 \end{bmatrix}$ are the coefficients of $f(\cdot)$ and $g(\cdot)$, respectively. The dynamics of time-delay term is given as follows:

$$g_i(v_i(t)) = \begin{bmatrix} m_{i1} v_{i1}(t) \cos(v_{i2}(t)) \\ m_{i2} v_{i2}(t) \sin(v_{i1}(t)) \end{bmatrix}. \tag{14}$$

Then, $\phi_i(v_i(t)) = \sqrt{(m_{i1} v_{i1}(t))^2 + (m_{i2} v_{i2}(t))^2}$. The unknown dynamics is chosen to be

$$f_i(p_i(t), v_i(t)) = \begin{bmatrix} l_{i1} p_{i1}(t) \sin(p_{i2}(t)) v_{i1} v_{i2} \\ l_{i2} p_{i2}(t) \cos(p_{i1}(t)) \sin(v_{i1} v_{i2}) \end{bmatrix}. \tag{15}$$

Suppose that all the five agents have the same parameters, $\hat{P} = 1000$, $\tau_{\max} = 0.15$, $k_{i0} = 10$, $\omega_i = 50$, $W_i^{\max} = 100$ and $\chi_i = 100$. The number of neurons for each RBFNN is 16 and $\alpha_i^2 = 2$. μ_i is distributed uniformly among the range $[0, 4] \times [0, 4]$.

In Fig. 1, the red asterisks are the initial positions and the blue arrows are the directions of initial velocities. In Fig. 2, the red pentagram is the final position showing that consensus can be achieved with the developed distributed

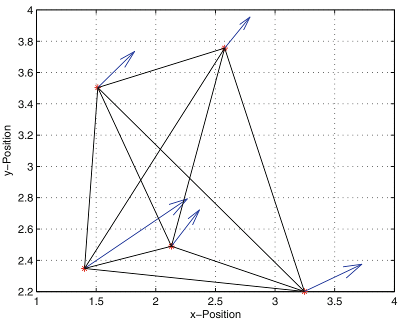

Fig. 1. Initial topology (Color figure online)

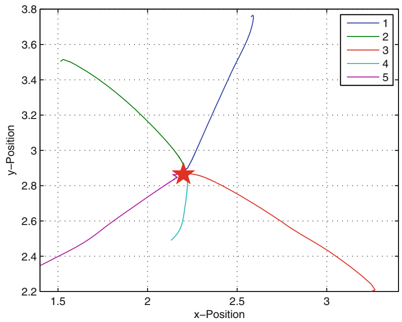

Fig. 2. Consensus of the five agents (Color figure online)

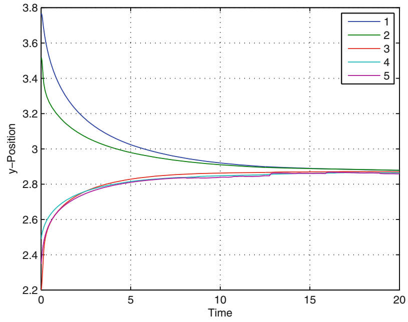

Fig. 3. Trajectories of position in y-axis

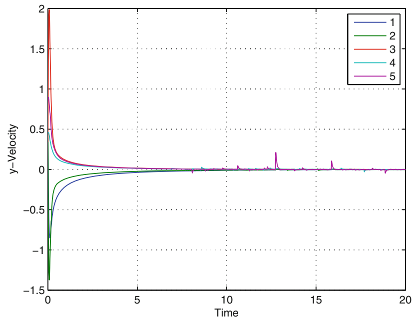

Fig. 4. Trajectories of velocity in y-axis

controller. We choose to show the trajectories of positions and velocities in y-axis in Figs. 3 and 4, respectively.

5 Conclusion

Nonlinear time-delayed multi-agent systems are investigated in this paper. The distributed controller is divided into five parts. By using RBFNNs, the distributed controller can learn the unknown nonlinear dynamics online. Furthermore, by introducing Lyapunov-Krasovskii functional, the effect of time delays can be eliminated. Finally, connectivity preservation can be guaranteed by designing a high-threshold potential function. Simulation results show the effectiveness of the developed distributed controller.

Acknowledgement. This work was supported in part by the National Natural Science Foundation of China under Grants 61233001, 61273140, 61304086, and 61374105, in part by Beijing Natural Science Foundation under Grant 4132078, and in part by the Early Career Development Award of SKLMCCS.

References

1. Ren, W., Beard, R.W., Atkins, E.M.: A survey of consensus problems in multi-agent coordination. In: Proceedings of the American Control Conference, pp. 1859–1864. Portland, OR, USA (2005)
2. Olfati-Saber, R., Fax, J.A., Murray, R.M.: Consensus and cooperation in networked multi-agent systems. Proc. IEEE **95**(1), 215–233 (2007)
3. Zhang, H., Zhang, J., Yang, G.H., Luo, Y.: Leader-based optimal coordination control for the consensus problem of multiagent differential games via fuzzy adaptive dynamic programming. IEEE Trans. Fuzzy Syst. **23**(1), 152–163 (2015)
4. Liu, T.F., Jiang, Z.P.: Distributed output-feedback control of nonlinear multi-agent systems. IEEE Trans. Autom. Control **58**(11), 2912–2917 (2013)

5. Liu, D., Wang, D., Li, H.: Decentralized stabilization for a class of continuous-time nonlinear interconnected systems using online learning optimal control approach. IEEE Trans. Neural Netw. Learn. Syst. **25**(2), 418–428 (2014)
6. Sun, X., Ge, S.S.: Adaptive neural region tracking control of multi-fully actuated ocean surface vessels. IEEE/CAA J. Automatica Sinica **1**(1), 77–83 (2014)
7. Hou, Z.G., Cheng, L., Tan, M.: Decentralized robust adaptive control for the multiagent system consensus problem using neural networks. IEEE Trans. Sys. Man Cybern. Part B: Cybern. **39**(3), 636–647 (2009)
8. Su, H., Wang, X., Chen, G.: Rendezvous of multiple mobile agents with preserved network connectivity. Syst. Control Lett. **59**(5), 313–322 (2010)
9. Chen, C., Wen, G.X., Liu, Y.J., Wang, F.Y.: Adaptive consensus control for a class of nonlinear multiagent time-delay systems using neural networks. IEEE Trans. Neural Netw. Learn. Syst. **25**(6), 1217–1226 (2014)
10. Ge, S., Hong, F., Lee, T.: Adaptive neural control of nonlinear time-delay systems with unknown virtual control coefficients. IEEE Trans. Syst. Man Cybern. Part B: Cybern. **34**(1), 499–516 (2004)