

Article

Sliding Surface in Consensus Problem of Multi-Agent Rigid Manipulators with Neural Network Controller

Thang Nguyen Trong *  and Minh Nguyen Duc

Department of Electrical Engineering and Automation, Haiphong Private University,
Haiphong 181810, Vietnam; minhdc@hpu.edu.vn

* Correspondence: phdthangnguyen@yahoo.com; Tel.: +84-168-846-8555

Received: 12 November 2017; Accepted: 12 December 2017; Published: 14 December 2017

Abstract: Based on Lyapunov theory, this research demonstrates the stability of the sliding surface in the consensus problem of multi-agent systems. Each agent in this system is represented by the dynamically uncertain robot, unstructured disturbances, and nonlinear friction, especially when the dynamic function of agent is unknown. All system states use neural network online weight tuning algorithms to compensate for the disturbance and uncertainty. Each agent in the system has a different position, and their trajectory approach to the same target is from each distinct orientation. In this research, we analyze the design of the sliding surface for this model and demonstrate which type of sliding surface is the best for the consensus problem. Lastly, simulation results are presented to certify the correctness and the effectiveness of the proposed control method.

Keywords: Euler–Lagrange System; neural network; consensus; sliding mode control; multi-agent system

1. Introduction

In recent years, using Sliding Mode Control (SMC) for robotic trajectory control of nonlinear dynamic properties has received much attention from researchers. SMC is one of the influential nonlinear controllers for linear and nonlinear systems. In comparison with other nonlinear control methods. The SMC method is relatively easy to implement, but in nonlinear dynamic systems, there are many uncertainties, such as disturbance and friction, that are the main reasons for the reduction of control quality. Therefore, to deal with this problem, the uncertainty of the system is compensated for by using of sliding mode control combined with a Neural Network (NN), which has been introduced in [1–7], in which the control algorithm drawn on Lyapunov theorem, the sliding mode control structure, and the neural network learning algorithm are developed. Therefore, the system stability is ensured, and the system dynamic performance is enhanced. In addition, other researchers used the optimal method to extract a rotation rule for a sliding surface against disturbance and variations in [8].

In order to further improve the aforementioned results, in this research, we introduce the control method for the nonlinear multi-agent system. The most important issue in this research is the consensus problem in robots, which is the precise control of multiple robots operating at the same time at the same goal. When working collectively, one robot will be installed as the leader. We only control the leader robot, and all other robots will receive the control signal from this leader to operate. In [9,10], the controller has been designed with a sliding mode control for linear multi-agent systems. A second-order sliding mode protocol has been designed for diminishing the chattering phenomenon. However, the disadvantage of this method is that it cannot radically reduce the chattering phenomenon around the sliding surface. In [11–14], a new consensus algorithm combining the concepts of graph theory and using fuzzy logic for the compensation of uncertainties is introduced to deal with consensus problems of the nonlinear multi-agent system.

In this research, in order to model the communication topology between agents, the graph theory in [6,15] is utilized. For each agent, the nonlinear dynamic model with uncertainty is considered. In addition, an Artificial Neural Network (ANN) with online learning weights has been built in [16,17] to solve the disadvantages of the traditional sliding mode control and the weights are adapted to such a point that the overall system performance is ameliorated. Not with standing that the research [17] has proposed some significant results for nonlinear multi-agent systems when using sliding mode control type, the problems of disturbance rejection with uncertainties for the entire system has not been considered completely in literature.

Motivated by the discussion above, different from the previous works, in this research, a synergistic combination of Proportional–Integral–Derivative (PID) sliding mode control with neural networks is proposed to reject disturbances and uncertainties of a nonlinear dynamic system in the consensus problem of multi-agent systems. Relying on Lyapunov stability theory in [18–28], it is demonstrated that the proposed control method ensures the outputs of all agents can trace the reference trajectory under the condition that the communication topology between the agents is described by a directed graph (Graph theory). In this research, for the first time, the NN in collaboration with PID sliding mode control [29,30] is implemented to the uncertain nonlinear multi-agent systems for solving the robust adaptive consensus problem. Simulation studies are carried out to verify the effectiveness of the proposed control method. The research' main contributions are threefold:

- (i) An adaptive neural network controller is designed to neutralize the disturbance and uncertain nonlinear dynamics in the multi-agent system. The fact that the proposed approach based on Lyapunov theory guarantees the system's stability has been demonstrated.
- (ii) The parametric uncertainties are estimated.
- (iii) The consensus algorithms are modified to solve formation control problems of multi-agent systems.

The remnant of this research is structured as follows. Some backgrounds of sliding mode control theory, graph theory and tracking control of the Euler–Lagrange system with the artificial neural network are presented in Section 2. Stability analysis and design neural network controller are introduced in Section 3. Numerical simulation of the multi-agent system is introduced in Section 4, and lastly in Section 5, some conclusions are presented.

Notation: Throughout this research, $\|\cdot\|$ stands for Euclidean norm of vectors and induced norm of matrices. $\text{Sign}(\cdot)$ denotes the sign function. R^n and $R^{n \times m}$ denote the n -dimensional Euclidean space and the set of $n \times m$ real matrices, respectively.

2. Problem Formulation

2.1. Sliding Mode Control

This subsection will present some preliminary knowledge for sliding mode control theory. For more detail, the interested readers may refer to [31–33].

As is well-known, sliding mode control combined with an online learning neural network is a very robust method for controlling of uncertain nonlinear systems. This research focuses on choosing the sliding surface in the consensus problem when we have some agents with different positions but they have the same mission. Because each agent has a different position, consequently to approach the object they must have distinct desired trajectories.

Define a general sliding surface as follows:

$$s(t) = e(t) + \Lambda_1 \frac{de(t)}{dt} + \dots + \Lambda_{n-1} \frac{d^{n-1}e(t)}{dt^{n-1}} \quad (1)$$

where $s(t)$ denotes a sliding surface, $e = q - q_d$ denotes error of the real trajectory and desired trajectory, $\Lambda_i = (\Lambda_1, \Lambda_2, \dots, \Lambda_{n-1})$ with $i = 1, 2, \dots, n - 1$ denote coefficients. When $s_{(t=0)} = 0$, in order to ensure that $\lim_{t \rightarrow 0} e(t) = 0$, we must choose the coefficients Λ_i of characteristic polynomial as follows:

$$\Lambda(z) = 1 + \Lambda_1 z + \dots + \Lambda_{n-1} z^{n-1} \quad (2)$$

where $\Lambda(z)$ is a Hurwitz polynomial with real coefficients. All the coefficients of $\Lambda(z)$ should be the same sign. In this case, we assume that $\Lambda_k > 0$, for $k = 1, 2, \dots, n - 1$.

Usually, with agents have nonlinear dynamic properties [34], a sliding surface usually is chosen in Proportional–Derivative (PD) or Proportional–Integral–Derivative (PID) forms. The (Proportional–Integral) PI sliding surface isn't chosen because it is very difficult to control the stability of system, and the transition time is long.

According to the design principles in [31,32], using the sign function, the sliding surface respectively will be the following:

Sliding surface with PD form:

$$s = \dot{e} + \Lambda e \quad (3)$$

Sliding surface with PID form:

$$s = \dot{e} + \Lambda_1 e + \Lambda_2 \int_0^t e dt \quad (4)$$

where $s = [s_1, s_2, \dots, s_n]^T$ and Λ_1, Λ_2 are diagonal positive definite matrices. e denotes error between the reference signal and the output signal.

Remark 1. The selection of the sliding surface type will be determined by the feature of the control objects. As pointed in [33], for the control of uncertain systems with multi-input-multi-output system, a sliding mode control method with high-order is developed. The PD sliding mode control method (3) is suitable for the control of a robot manipulator which a feature is the uncertain nonlinear system [1,31]. In order to reduce the chattering efficiently, enhance the control performance, disturbance rejection and system stability in [1,31], the PID sliding mode control method (4) is proposed in [32]. In the other research [12], a general sliding surface (1) is presented for the consensus problem of high-order nonlinear systems.

2.2. Tracking control of Euler–Lagrange System with an Artificial Neural Network

Consider a general n degree of freedom rigid manipulator, which considers the disturbances and friction, the Euler–Lagrange equation [16,17,25] is represented in the form

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + F_r\dot{q} + \tau_d = \tau \quad (5)$$

where $D(q) \in R^{n \times n}$ represents a symmetric and positive definite matrix. $C(q, \dot{q}) \in R^n$ denotes the centrifugal and Coriolis forces. $F_r(\dot{q}) \in R^n$ denotes friction. $G(q) \in R^n$ represents the gravity forces. $\tau_d \in R^n$ denotes a general nonlinear disturbance. $\tau \in R^n$ denotes the torque input controls vector. $q, \dot{q}, \ddot{q} \in R^n$ represent the angle, velocity and acceleration vector of link, respectively.

Property 1. The inertia matrix $D(q)$ is symmetric and uniformly positive definite, and satisfying

$$\forall s \in R^n, \quad g s^T s \leq s^T D s \leq \bar{g} s^T s$$

where g and \bar{g} are known positive constants.

Property 2. The Coriolis and centrifugal matrix $C(q, \dot{q})$ can be appropriately determined such that $(\dot{D} - 2C)$ is skew-symmetric, it is easy to verify that $s^T (\dot{D} - 2C)s = 0, \forall s \neq 0$.

In Equation (5), we let

$$F(q, \dot{q}, \ddot{q}, t) = F_r\dot{q} + \tau_d \quad (6)$$

and

$$\tau_0 = D(q)\dot{q} + C(q, \dot{q})\dot{q} + G(q) \quad (7)$$

Now, Equation (6) is rewritten as

$$\tau = \tau_0 + F(q, \dot{q}, \ddot{q}, t) \quad (8)$$

In the nonlinear dynamic systems, disturbance and friction are the main reasons to reduce the control quality. Consequently, to improve the control quality, in this research, we define an artificial neural network for the purposes of compensation in (6).

In studies on robust adaptive control, NN in [5] are mostly used for the unknown nonlinearities as approximation models because of their inherent capabilities of approximation. A simple artificial neural network structure for approximating function may be rewritten as

$$F(q, \dot{q}, \ddot{q}, t) = F(s) = W\sigma + \varepsilon \quad (9)$$

where ε denotes the error of the approximation, and $\bar{\varepsilon}$ is the limit of ε , ($|\varepsilon| \leq \bar{\varepsilon}$). σ denotes the acting Gaussian function.

We denote the weights w_{ji} to construct an approximated neural network, consequently the Equation (9) could be rewritten as

$$F(s) = \sum_{j=1}^n w_{ji}\sigma_i + \varepsilon \quad i = 1, 2, \dots, n \quad (10)$$

The acting Gaussian function is described as

$$\sigma_i = \exp\left(-\frac{s_i - c_i}{\lambda_i^2}\right) \quad (11)$$

where $S_i = [S_{i1}, S_{i2}, \dots, S_{in}]^T$ denotes the input to the Radial Basis Function (RBF) network, $c_i = [c_1, c_2, \dots, c_n]^T$ is the centers of the basic function and λ_i is the widths of the basic function, freely chosen.

Remark 2. It is noted that similar to the ANN presented in [25–27,32], the ANN structure above is developed (refer to Equations (6) and (9)–(11)). However, it applies to the consensus problem of uncertain nonlinear multi-agent systems is more complex and is not easy to verify. Additionally, current disturbances are also created from the communication topology among each agent during the movement. This problem can be solved by appropriately selecting an ANN with online learning weights where the number of the neuron on each layer of the ANN, c_i and λ_i could be free to find a suitable value.

2.3. Graph Theory and the Laplacian Matrix

A graph theory is applied to represent the transmission relationship among agents, which is called Communication Graph. If the number of agents is m , the graph G includes of a node set $\gamma = \{v_1, v_2, \dots, v_m\}$, an edge set $\zeta \subseteq \gamma \times \gamma$ and a weighted adjacent matrix $A = [\delta_{ij}] \in R^{m \times m}$, where $\delta_{ij} > 0$ means that agent i can acquire the information from agent j , otherwise $\delta_{ij} = 0$. The adjacent matrix A is a symmetric matrix and defined as $\delta_{ij} = \delta_{ji}$. Associated with A , the Laplacian matrix is introduced by $L = [l_{ij}] \in R^{m \times m}$, where $[l_{jj}] = \sum_{i=1, i \neq j}^m \delta_{ij}$ and $l_{ij} = \delta_{ij}$, $i \neq j$.

We determine a limited set including hCommunication Graphs $G^* = \{G_1, G_2, \dots, G_h\}$ such that all hCommunication Graphshave the same node set ($\gamma_1 = \gamma_2 = \dots = \gamma_h = \gamma$). But the edge set for each hCommunication Graphs is different ($\zeta_1 \neq \zeta_2 \neq \dots \neq \zeta_h$), which results in a different weighted adjacent matrix for each Communication Graphs, specifically, $A_1 \neq A_2 \neq \dots \neq A_h$. Accordingly, the Laplacian matrix associated with each $i \in \{1, 2, \dots, h\}$ Communication Graphs, named L_i , will also

be dissimilar. We must note that all hCommunication Graphs are connected. Thus, L_i is a positive semi-definite matrix $\forall i \in \{1, 2, \dots, h\}$.

Lemma 1. Zero is a simple eigenvalue of L if and only if graph G has a directed spanning tree [35].

Lemma 2. In [15], the graph G is strongly connected if and only if $\text{rank}(L) = n - 1$.

Lemma 2. On the same row of the Laplacian matrix L , the algebraic cofactors of the elements are equal.

The following are the objectives of this research:

- (1) Presenting a sliding mode control suited to uncertain nonlinear multi-agent system (5) such that it obtains fine transient performance without accurate parameters.
- (2) Designing an adaptive neural network controller such that the tracking error $e = q - q_d$ congregates to a predefined little neighborhood of zero in a limited time.
- (3) Presenting a consensus or synchronization methods for multi-agent rigid manipulator with uncertain nonlinear dynamics.

3. Problem Formulation

Typically, the sliding mode control method includes determining the sliding surface as the system state functions and using finite-time stability theory in [19] to prove that the trajectories of closed-loop system arrive this surface infinite time. In order to solve the disadvantages in the typical sliding mode control, a neural network is added. The learning algorithm of neural network and sliding mode controller are constructed relying on Lyapunov theorem to ensure the stability of the system.

In this section, two kinds of sliding mode control in the forms (3) and (4) are used to demonstrate the system stability when using the neural network for compensating for the disturbance and uncertain dynamic system. The analysis result shown in the end of this section will prove that the PD or PID sliding mode controller is suitable for the consensus problem of multi-agent nonlinear dynamic system.

According to Equations (5) and (6),

$$\begin{aligned}\ddot{q} &= D^{-1}(q)[\tau - C(q, \dot{q})\dot{q} - G(q) - F_r\dot{q} - \tau_d] \\ &= D^{-1}(q)\tau - D^{-1}(q)(C(q, \dot{q})\dot{q} - G(q) - F(q, \dot{q}, \ddot{q}, t))\end{aligned}\quad (12)$$

Denoted in [3], we propose a control structure for the agent with nonlinear dynamic properties. The control system structure is shown as Figure 1. The NN with online learning is used for compensating the disturbances and friction. The control objective is to make q follow a certain desired trajectory q_d in the presence of system uncertainty and disturbance.

To deal with this problem, we determine a Lyapunov function $V(t)$, and relying on Lyapunov stability theory, if we make a control law so that the time derivative of $V(t)$ is negative, then the trajectory of the agent will converge to the sliding surface $s = 0$ in a finite time and keep them on the sliding surface.

Choosing the Lyapunov function as below,

$$V(t) = \frac{1}{2}(s^T D s + \sum_{j=1}^n w_j^T w_j)\quad (13)$$

It can be noted that this function is positive definite $V(t) > 0, \forall s \neq 0$.

Differentiating Equation (13) with respect to time, we have

$$\begin{aligned} \dot{V}(t) &= \frac{1}{2}[\dot{s}^T Ds + s^T \dot{D}s + s^T D\dot{s} + \sum_{j=1}^n (\dot{w}_j^T w_j + w_j^T \dot{w}_j)] \\ &= \frac{1}{2}s^T \dot{D}s + s^T D\dot{s} + \sum_{j=1}^n w_j^T \dot{w}_j \end{aligned} \tag{14}$$

According to dynamic Equation (5), both $C(q, \dot{q})$ and $D(q)$ satisfy Property 2,

$$\begin{aligned} s^T (\dot{D} - 2C)s &= 0, \quad \forall s \in R^n \\ \leftrightarrow s^T \dot{D}s &= 2s^T Cs \end{aligned} \tag{15}$$

The matrix $(\dot{D} - 2C)$ is a symmetric matrix. This characteristic ensures the system unaffected by the force is defined by $C(q, \dot{q})\dot{q}$.

Consequently, from (14) and (15) we have

$$\dot{V}(t) = s^T Cs + s^T D\dot{s} + \sum_{j=1}^n w_j^T \dot{w}_j \tag{16}$$

In the sequel, we consider the following two types of sliding surfaces.

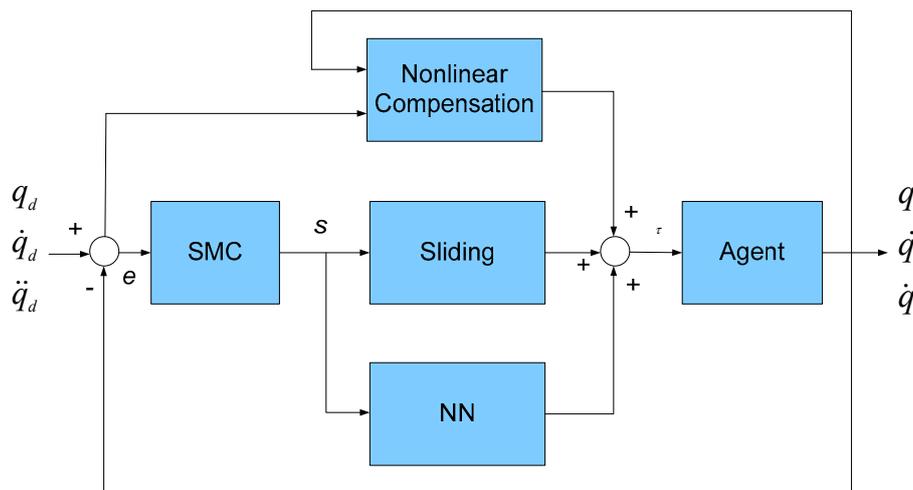


Figure 1. The controller structure with nonlinear compensation Neural Network (NN).

3.1. PD Type Sliding Mode Control

With this sliding surface type and neural network (10), to track desired trajectory q_d of Euler–Lagrange system (5) with error $e = (q - q_d) \rightarrow 0$, we propose the following control law τ :

$$\begin{aligned} \tau &= D\ddot{q}_d + C\dot{q}_d + G - D\Lambda\dot{e} - C\Lambda e \\ &\quad - Ks - \gamma \frac{s}{\|s\|} + (1 + \eta)W\sigma \end{aligned} \tag{17}$$

and the learning algorithm \dot{w}_i

$$\dot{w}_i = -\eta s \sigma_i \tag{18}$$

where K denotes a symmetric positive matrix, $\gamma, \eta > 0$.

Theorem 1. Consider the Euler–Lagrange system (5), sliding mode control (3), approximating neural network defined in (10) with Gaussian function is described in (11), the proposed state feedback control law (17) and

learning algorithm (18). All the system signals are limited and the tracking error congregates to sliding surface s in the limit time.

Proof. Substituting the PD sliding surface described in the form (3) to the Equation (16), it can be obtained that:

$$\begin{aligned}\dot{V}(t) &= s^T [C(\dot{e} + \Lambda e + D(\ddot{e} + \Lambda \dot{e})) + \sum_{j=1}^n w_j^T \dot{w}_j] \\ &= s^T [C(\dot{q} - \dot{q}_d + \Lambda e) + D(\ddot{q} - \ddot{q}_d + \Lambda \dot{e})] + \sum_{j=1}^n w_j^T \dot{w}_j \\ &= s^T [C(-\dot{q}_d + \Lambda e) + D(-\ddot{q}_d + \Lambda \dot{e}) + C\dot{q} + D\ddot{q}] + \sum_{j=1}^n w_j^T \dot{w}_j\end{aligned}\quad (19)$$

□

According to Equations (5) and (6), we have

$$\begin{aligned}D\ddot{q} + C\dot{q} &= \tau - G - F(q, \dot{q}, \ddot{q}, t) \\ &= \tau - G - W\sigma - \varepsilon\end{aligned}\quad (20)$$

By substituting (20) to (19), it is obtained that

$$\begin{aligned}\dot{V}(t) &= s^T [C(-\dot{q}_d + \Lambda e) + D(-\ddot{q}_d + \Lambda \dot{e}) + \\ &\quad + \tau - G - W\sigma - \varepsilon] + \sum_{j=1}^n w_j^T \dot{w}_j\end{aligned}\quad (21)$$

Replacing τ in (17) and (18) into (21) leads to

$$\dot{V}(t) = s^T [-Ks - \gamma \frac{s}{\|s\|} + \eta W\sigma - \varepsilon] - \eta \sum_{j=1}^n w_j^T s \sigma_j\quad (22)$$

We have $w_j^T s = s^T w_j$ and from (10) and (11), so the last part in Equation (22) can be rewritten to the following:

$$\eta \sum_{j=1}^n w_j^T s \sigma_j = \eta s^T \sum_{j=1}^n w_j \sigma_j = \eta s^T W\sigma\quad (23)$$

Therefore Equation (22) becomes

$$\begin{aligned}\dot{V}(t) &= s^T [-Ks - \gamma \frac{s}{\|s\|} + \eta W\sigma - \varepsilon] - \eta s^T W\sigma \\ &= s^T [-Ks - \gamma \frac{s}{\|s\|} - \varepsilon]\end{aligned}\quad (24)$$

From the Equation (24) we can observe that $\dot{V}(t) < 0$ for all $s \neq 0$ and $\dot{V}(t) = 0$ if and only if $s = 0$. So, with sliding surface (3), control law (17), learning algorithm (18) and neural network (9), the agent will be tracking the desired trajectory q_d with error $e \rightarrow 0$.

3.2. PID Type Sliding Mode Control

In this case, to stabilize the Euler–Lagrange system (5), we propose the following control law τ

$$\begin{aligned}\tau &= D\ddot{q}_d + C\dot{q}_d + G - D(\Lambda_1 \dot{e} + \Lambda_2 e) - \\ &\quad - C(\Lambda_1 e + \Lambda_2 \int_0^t e dt) - Ks - \gamma \frac{s}{\|s\|} + (1 + \eta)W\sigma\end{aligned}\quad (25)$$

Theorem 2. Consider the Euler–Lagrange system (5), sliding mode control (4), approximating neural network defined in (10) with Gaussian function is described in (11), control law (25) and learning algorithm (18). All the system signals are limited and the tracking error congregates to sliding surface s in the limit time.

Proof. From (16) and (4), we have

$$\begin{aligned} Cs + D\dot{s} &= C(\dot{e} + \Lambda_1 e + \Lambda_2 \int_0^t e dt) + D(\ddot{e} + \Lambda_1 \dot{e} + \Lambda_2 e) \\ &= C(-\dot{q}_d + \Lambda_1 e + \Lambda_2 \int_0^t e dt) \\ &\quad + D(-\ddot{q}_d + \Lambda_1 \dot{e} + \Lambda_2 e) + C\dot{q} + D\ddot{q} \end{aligned} \quad (26)$$

□

Substituting (20) into (26), we have

$$\begin{aligned} Cs + D\dot{s} &= C(-\dot{q}_d + \Lambda_1 e + \Lambda_2 \int_0^t e dt) + \\ &\quad + D(-\ddot{q}_d + \Lambda_1 \dot{e} + \Lambda_2 e) + \tau - G - W\sigma - \varepsilon \end{aligned} \quad (27)$$

Replacing (25) into (27), we have

$$Cs + D\dot{s} = -Ks - \gamma \frac{s}{\|S\|} + \eta W\sigma - \varepsilon \quad (28)$$

Substituting (28) into (16), yields

$$\dot{V}(t) = s^T (-Ks - \gamma \frac{s}{\|S\|} + \eta W\sigma - \varepsilon) + \sum_{j=1}^n w_j^T \dot{w}_j \quad (29)$$

From the Equations (10), (11), (18), and (23), Equation (29) is rewritten as

$$\begin{aligned} \dot{V}(t) &= s^T (-Ks - \gamma \frac{s}{\|S\|} + \eta W\sigma - \varepsilon) - \eta W\sigma \\ &= s^T (-Ks - \gamma \frac{s}{\|S\|} - \varepsilon) \end{aligned} \quad (30)$$

From the Equation (30) we can observe that $\dot{V}(t) < 0$ for all $s \neq 0$ and $\dot{V}(t) = 0$ if and only if $s = 0$. So, with sliding surface (4), control law (25), learning algorithm (18) and neural network (9), the agent will be tracking the desired trajectory q_d with error $e \rightarrow 0$. Relying on the Lyapunov stability theorem, it can be acknowledged that the state trajectories are able to reach the sliding surface $s = 0$ in a finite time. We can conclude that the nonlinear multi-agent system is asymptotically stable.

Observed the above stability analysis of Euler–Lagrange system, with PD and PID sliding mode control we choose the control law as (17) and (25), respectively. In traditional problems of robot control, we often use PD sliding surface combined with a neural network to track the desired trajectory [17]. Nevertheless, the applications of the traditional sliding mode control suffers in practical motion has disadvantages. The first one is the difficulty in obtaining parameters of the system. The second one is the consistent existence of high-frequency oscillation in the control input, it is chattering phenomenon. For diminishing this chattering phenomenon and improve the accuracy of the multi-agent system, a PID sliding surface with the online learning neural network are presented. With integral part participated in the sliding surface (4), when $e \rightarrow 0$, the system brings $\int e dt \rightarrow 0$ and eliminates the effects of disturbances and integrated errors, leading to higher accuracy.

Remark 3. In the two cases above analysis, it can be seen from Equations (20)–(22) and (27)–(29) that the sign of the $\dot{V}(t)$ in (24) and (30) depend on control law is proposed in (17) and (25). It is also noted in [24], handling the existence of the residue of the NN approximation is the pupose of constructing a robust controller.

The proposed controller can manage the unknown upper bounds of the NN approximation error Based on the robust technique.

Remark 4. A PID sliding mode control using saturated function has been suggested by some researchers in literature [36–38]. The advantage of this method can be rapidly convergence, but the system stability has not been proved and chattering still exists. So, in this research, the PID-SMC is not the main controller, it is only a type of the sliding surface (4) used in the control process.

Remark 5. The consensus problem of multi-agent systems is valuable and very interesting. Many authors have studied some new models e.g., [20–23,27–29,39,40] and many control approaches are proposed. Their models can be linear systems or nonlinear systems but almost all of them omit friction and disturbance or if it exists, then removing it is not focused. To solve these drawbacks, the control method in this paper is presented by combining neural network with PID-SMC, such that the uncertainty and disturbances of nonlinear multi-agent dynamic system are thoroughly suppressed as well as the system control performance is improved significantly. The numerical simulation in the next section will be proved for this conclusion.

4. Numerical Simulations

In this part, based on the findings acquired in the previous sections, we present several simulation results for illustrating the feasibility and effectiveness of our theoretical outcomes. The PID sliding mode control problem formulation for multi-agent systems combined with a neural network has been implemented in Matlab simulation.

The neural network [41] used for identification is as (10), the acting Gaussian function [42] is described in form (11), where the Gaussian function parameters are as follows:

$$\lambda_1 = 1, \lambda_2 = 1, c_1 = 0.01, c_2 = 0.01$$

The first status for the neural network is

$$w_{0,0} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

Learning rate $\eta = 0.1$ and $\gamma = 2$. Then the input/output responses of the plant of the leader robot are shown in Figure 2, the responses of the reference model of the neural network controller for training the leader robot are shown in Figure 3.

In Figure 2, the input is the leader robot input, the output is the leader robot output after being trained and identified by the neural network.

The reference model is a neural network with the numbers of layers and neurons in each class are proposed and tested by the authors in Matlab software.

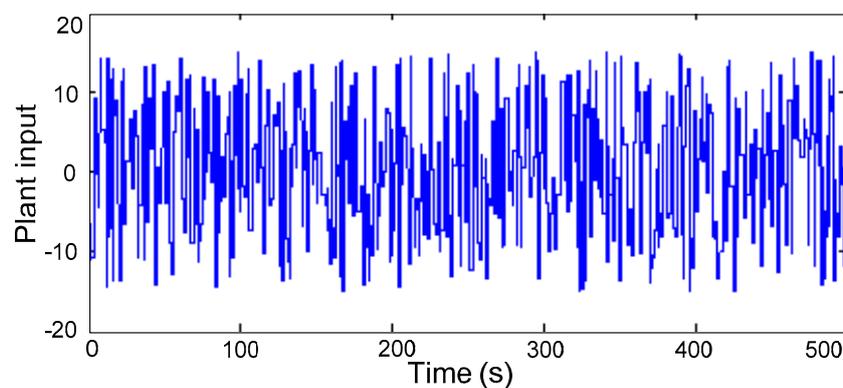


Figure 2. Cont.

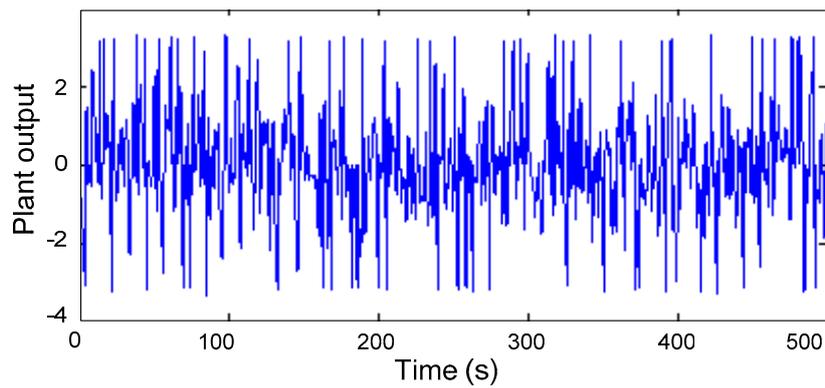


Figure 2. Plant input and output.

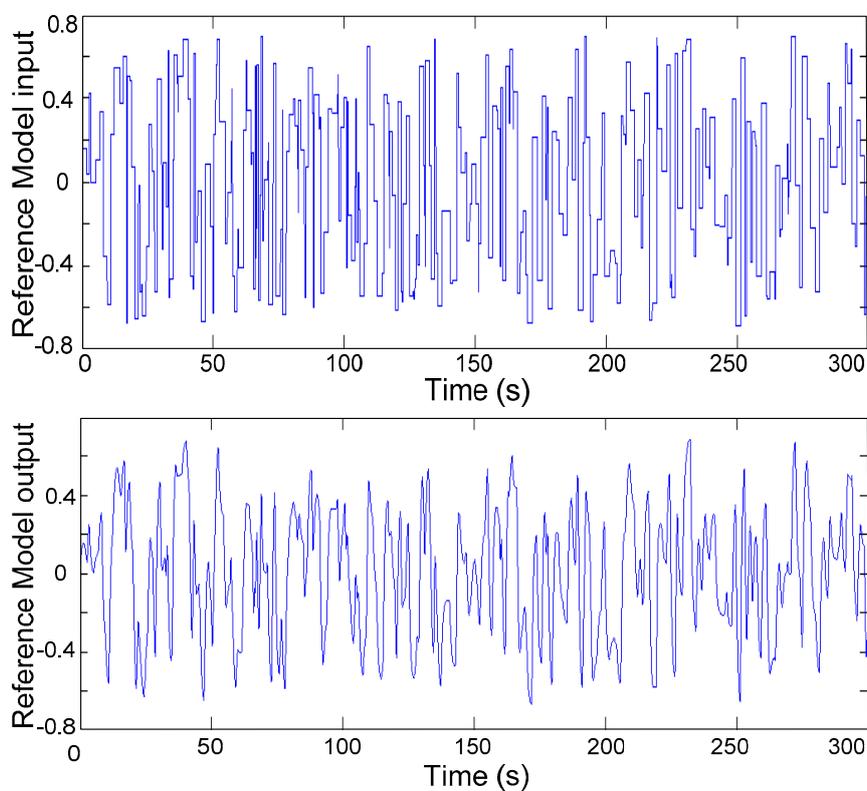


Figure 3. Reference Model input and output.

To prove the effectiveness of the proposed approach, a nonlinear multi-agent consensus example is given. In the example, the multi-agent system includes five agents, each agent in the system is a 2 Degrees of Freedom (DOFs) rigid manipulator and the dynamic function of agent is unidentified.

The dynamic equation of 2 DOFs rigid manipulator is represented by Euler–Lagrange system [13,17] which is described as follows:

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} g_1(q_1, q_2) \\ g_2(q_1, q_2) \end{bmatrix} \quad (31)$$

where

$$\begin{aligned}
 D_{11}(q_2) &= (m_1 + m_2)l_1^2 + m_2l_2^2 + 2m_2l_1l_2c_2 \\
 D_{12}(q_2) &= D_{21}(q_2) = m_2l_2^2 + m_2l_1l_2c_2 \\
 D_{22}(q_2) &= m_2l_2^2 \\
 C_1 &= -m_2l_1l_2\dot{q}_2s_2 - 2m_2l_1l_2\dot{q}_1s_2 \\
 C_2 &= m_2l_1l_2\dot{q}_1s_2 \\
 g_1(q_1, q_2) &= (m_1 + m_2)gl_1c_1 + m_2gl_2c_{12} \\
 g_2(q_1, q_2) &= m_2gl_2c_{12}
 \end{aligned}$$

And $c_1 = \cos(\theta_1)$, and $c_{12} = \cos(\theta_1 + \theta_2)$. m_1, l_1, m_2 , and l_2 denotes the mass and lengths of two links, respectively. θ_1 and θ_2 denotes the link variables. D_{ij}, C_i, g_i , and τ_i are the matrix and dynamic torque defined in (5).

Setting the parameters of each manipulator: $g = 10 \text{ m/s}^2, m_1 = 1 \text{ kg}, l_1 = 1 \text{ m}, m_2 = 1 \text{ kg}, l_2 = 1 \text{ m}$ and substitute them to $D_{11}, D_{12}, D_{22}, C_1, C_2$, and g , then substitute to (31), which yields:

$$\begin{aligned}
 \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} &= \begin{bmatrix} 3 + 2c_2 & 1 + c_2 \\ 1 + c_2 & 1 \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} \\
 + \begin{bmatrix} -\dot{q}_2s_2 - 2\dot{q}_1s_2 \\ \dot{q}_1s_2 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} &+ \begin{bmatrix} 20c_1 + 10c_{12} \\ 10c_{12} \end{bmatrix}
 \end{aligned}$$

Suppose that the information exchanges topology among the agents is given by Figure 4. From Lemmas 2 and 3, the Graph theory in [17] is applied to obtain the following Laplacian matrix.

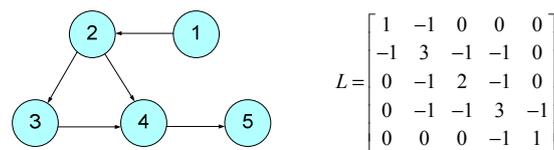


Figure 4. Communication topology with multi-agent system and its Laplacian.

The multi-agent system consists of five agents is illustrated in Figure 5, they have the same mission to move an object from the initial position to the required position, where $q_{da1}, q_{da2}, \dots, q_{da5}$ the desired trajectory of agent 1, agent 2, . . . , agent 5, respectively.

We define $q_{1i}, q_{2i}, (i = 1, 2, \dots, 5)$ as the first and second link trajectories of the five agents, respectively. The initial conditions of the five agents indicated by 1, 2, 3, 4, 5 were set as $q_{11}(0) = -4(\text{rad}), q_{21}(0) = \pi/2(\text{rad}); q_{12}(0) = 1.5(\text{rad}), q_{22}(0) = 1(\text{rad}); q_{13}(0) = 4.5(\text{rad}), q_{23}(0) = 4(\text{rad}); q_{14}(0) = -2(\text{rad}), q_{24}(0) = -1.5(\text{rad}); q_{15}(0) = -5(\text{rad}), q_{25}(0) = -5(\text{rad})$, respectively.

Choose reference motions trajectory as

$$q_{1id} = q_{2id} = -0.3 \cos(\pi t)$$

The sliding surface of the five agents are given as follows

$$\begin{aligned}
 s_{1i} &= \dot{e}_{1i} + \Lambda_{11}e_{1i} + \Lambda_{12}\int_0^t e_{1i}dt \quad (e_{1i} = q_{1i} - q_{1id}; i = 1, 2, \dots, 5) \\
 s_{2i} &= \dot{e}_{2i} + \Lambda_{21}e_{2i} + \Lambda_{22}\int_0^t e_{2i}dt \quad (e_{2i} = q_{2i} - q_{2id}; i = 1, 2, \dots, 5)
 \end{aligned}$$

where

$$\Lambda = \begin{bmatrix} 2 & 3 \\ 3 & 4 \end{bmatrix}$$

Set a friction on each link of the five agents as

$$F_r(\dot{q}) = \begin{bmatrix} 3\dot{q}_{1i} + 0.02\text{sgn}(\dot{q}_{1i}) \\ 2\dot{q}_{2i} + 0.02\text{sgn}(\dot{q}_{2i}) \end{bmatrix}$$

where $\text{sgn}(\cdot)$ denotes the sign function and disturbance as

$$\tau_d = \begin{bmatrix} 0.05 \sin(20t) \\ 0.1 \sin(20t) \end{bmatrix}$$

Model of the 5 agent's rigid manipulators attached move frames is shown in Figure 5.

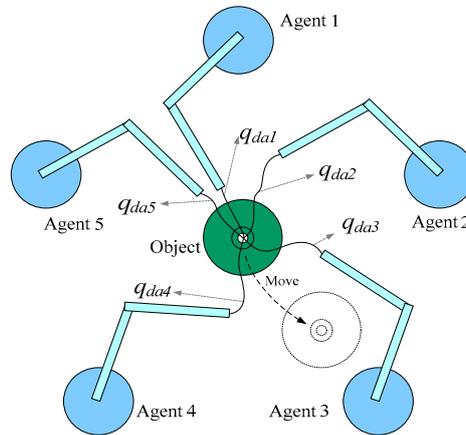


Figure 5. Multi-agent system model.

By employing the proposed combining neural network and PID sliding mode control approach for multi-agent system model described in Figure 5, simulation results were obtained, some of which are given in Figure 6.

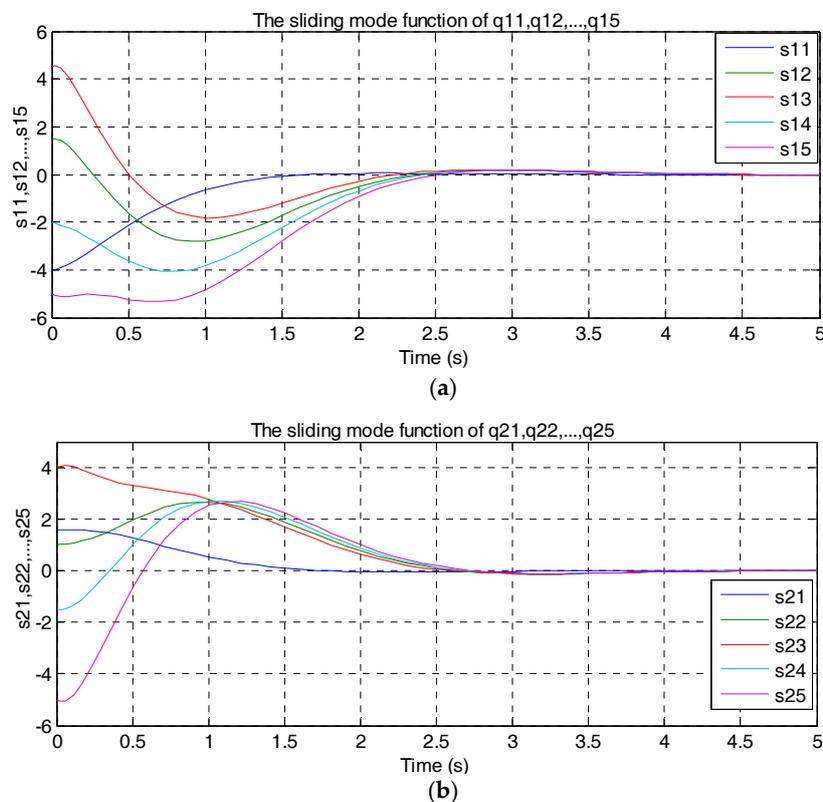


Figure 6. Response of sliding mode function of multi-agent system with q_{1i} , q_{2i} , respectively. (a) Response of s_{1i} ; (b).Response of s_{2i} .

5. Conclusions

This research presents a synergistic combination of PID type sliding mode control with neural networks. The neural network parameters and sliding mode term are calculated online with an intention to guarantee the stability of the system in a closed loop. This control scheme is constructed for tracking the desired motion trajectory of a multi-agent system, and the disturbance and uncertainty are rejected. Compare the result when using the control method of PD sliding mode control [17], we observed the significance of this method is that the stability and robustness against parameter uncertainties are ensured. In conclusion, some simulation findings are established to demonstrate the effectiveness of the proposed theoretical methods.

Author Contributions: Minh Nguyen Duc proposed the initial idea. Thang Nguyen Trong and Minh Nguyen Duc developed the research, analyzed the results, and wrote the paper together. Thang Nguyen Trong edited and finalized the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Peng, J.Z.; Wang, Y.N.; Sun, W.; Liu, Y. A Neural Network Sliding Mode Controller with Application to Robotic Manipulator. In Proceedings of the 6th World Congress on Intelligent Control and Automation, Dalian, China, 21–23 June 2006; pp. 2101–2105.
2. Cheng, L.; Zeng, G.H.; Tan, M. Adaptive Neural Network Tracking Control of Manipulators Using Quaternion Feedback. In Proceedings of the IEEE International Conference on Robotics and Automation, Pasadena, CA, USA, 19–23 May 2008; pp. 3371–3376.
3. Xie, X.L.; Cheng, L.; Zeng, G.H.; Cheng, J. Adaptive Neural Network Control of a 5 DOF Robot Manipulator. In Proceedings of the International Conference on Intelligent Control and Information Processing, Dalian, China, 13–15 August 2010; pp. 376–381.
4. Yan, Z.; Wang, J. Robust Model Predictive Control of Nonlinear Systems with Unmodeled Dynamics and Bounded Uncertainties Based on Neural Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2014**, *25*, 457–469. [[CrossRef](#)] [[PubMed](#)]
5. Chen, M.; Ge, S.S.; Bernard, V.E.H. Robust Adaptive Neural Network Control for a Class of Uncertain MIMO Nonlinear Systems with Input Nonlinearities. *IEEE Trans. Neural Netw.* **2010**, *21*, 796–812. [[CrossRef](#)] [[PubMed](#)]
6. Philip Chen, C.L.; 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.* **2014**, *25*, 1217–1226. [[CrossRef](#)]
7. Tee, K.P.; Ge, S.S.; Francis, E.H.T. Adaptive Neural Network Control for Helicopters in Vertical Flight. *IEEE Trans. Control Syst. Technol.* **2008**, *16*, 753–762.
8. Sajad, S.G.; Mohammad, B.H. Optimal Design of Rotating Sliding Surface for Sliding Mode Control. In Proceedings of the American Control Conference, St. Louis, MO, USA, 10–12 June 2009; pp. 774–777.
9. Zhao, N.; Zhu, J.D. Sliding Mode Control for Robust Consensus of Linear Multi-agent Systems. In Proceedings of the IEEE 10th World Congress on Intelligent Control and Automation, Beijing, China, 6–8 July 2012; pp. 1378–1382.
10. Haghshenas, H.; Badamchizadeh, M.A.; Baradarannia, M. Containment control of heterogeneous linear multi-agent systems. *Automatica* **2016**, *54*, 210–216. [[CrossRef](#)]
11. Chang, Y.H.; Chang, C.W.; Chan, W.C. Fuzzy Sliding Mode Consensus Control for Multi-Agent Systems. In Proceedings of the American Control Conference, San Francisco, CA, USA, 29 June–1 July 2011; pp. 1636–1641.
12. Wang, W.; Wang, D.; Peng, Z.H. Robust Adaptive Consensus of High-order Nonlinear Systems Using Fuzzy Logical Systems and Continuous Second-order Sliding Mode. In Proceedings of the 33rd Chinese Control Conference, Nanjing, China, 28–30 July 2014; pp. 1197–1201.
13. Xu, Y.Y.; Tong, S.C.; Li, Y.M. Adaptive fuzzy fault-tolerant dynamic surface control of nonlinear MIMO systems with actuator failures. In Proceedings of the 32th Chinese Control Conference, Xi'an, China, 26–28 July 2013; pp. 3423–3428.

14. Min, H.; Sun, F.; Wang, S.; Li, H. Distributed adaptive consensus algorithm for networked Euler–Lagrange systems. *IET Control Theory Appl.* **2011**, *5*, 145–154. [[CrossRef](#)]
15. Daniel, A.; Spielman. Algorithms, Graph Theory, and Linear Equations in Laplacian Matrices. In Proceedings of the International Congress of Mathematicians, Hyderabad, India, 19–27 August 2010.
16. Minh, N.D.; Thang, N.T. Neural Network Structures for Identification of Nonlinear Dynamic Robotic Manipulator. In Proceedings of the IEEE International Conference on Mechatronics and Automation, Tianjin, China, 3–6 August 2014; pp. 1575–1580.
17. Minh, N.D.; Wang, W.; Zhuang, Y. Consensus of Multi-Agent Systems with Euler–Lagrange System Using Neural Networks Controller. *ICIC Express Lett.* **2016**, *10*, 1697–1704.
18. Li, H.F.; Jiang, H.J.; Hu, C. Existence and global exponential stability of periodic solution of memristor-based BAM neural networks with time-varying delays. *Neural Netw.* **2016**, *75*, 97–109. [[CrossRef](#)] [[PubMed](#)]
19. Cai, Z.W.; Huang, L.H.; Zhua, M.X.; Wang, D.S. Finite-time stabilization control of memristor-based neural networks. *Nonlinear Anal. Hybrid Syst.* **2016**, *20*, 37–54. [[CrossRef](#)]
20. Ge, S.S.; Yang, C.G.; Dai, S.L.; Jiao, Z.X.; Lee, T.H. Robust adaptive control of a class of nonlinear strict-feedback discrete-time systems with exact output tracking. *Automatica* **2009**, *45*, 2537–2545. [[CrossRef](#)]
21. Dai, S.L.; Lin, H.; Ge, S.S. Scheduling-and-Control Codesign for a Collection of Networked Control Systems with Uncertain Delays. *IEEE Trans. Control Syst. Technol.* **2010**, *18*, 66–78. [[CrossRef](#)]
22. Chen, M.; Ge, S.S. Direct Adaptive Neural Control for a Class of Uncertain Nonaffine Nonlinear Systems Based on Disturbance Observer. *IEEE Trans. Cybern.* **2013**, *43*, 1213–1225. [[CrossRef](#)] [[PubMed](#)]
23. Bin, X.; Liang Wang, J. A recurrent neural network for nonlinear optimization with a continuously differentiable objective function and bound constraints. *IEEE Trans. Neural Netw.* **2000**, *11*, 1251–1262.
24. Wang, Z.P.; Ge, S.S.; Lee, T.H. Robust Adaptive Neural Network Control of Uncertain Nonholonomic Systems with Strong Nonlinear Drifts. *IEEE Trans. Syst. Man Cybern. B Cybern.* **2004**, *34*, 2048–2059. [[CrossRef](#)] [[PubMed](#)]
25. Ge, S.S.; Hang, C.C. Structural Network Modeling and Control of Rigid Body Robots. *IEEE Trans. Robot. Autom.* **1998**, *14*, 823–827. [[CrossRef](#)]
26. Ge, S.S.; Hang, C.C.; Zhang, T. Stable Adaptive Control for Nonlinear Multivariable Systems with a Triangular Control Structure. *IEEE Trans. Autom. Control* **2000**, *45*, 1221–1225. [[CrossRef](#)]
27. Ge, S.S.; Wang, C. Adaptive Neural Control of Uncertain MIMO Nonlinear Systems. *IEEE Trans. Neural Netw.* **2004**, *15*, 647–692. [[CrossRef](#)] [[PubMed](#)]
28. Zeng, G.H.; Madan, M.G.; Peter, N.N.; Tan, M.; Cheng, L. A Recurrent Neural Network for Hierarchical Control of Interconnected Dynamic Systems. *IEEE Trans. Neural Netw.* **2007**, *18*, 466–481.
29. Schimmack, M.; Feistauer, E.E.; Amancio-Filho, S.T.; Mercorelli, P. Hysteresis Analysis and Control of a Metal-Polymer Hybrid Soft Actuator. *Energies* **2017**, *10*, 508. [[CrossRef](#)]
30. Xu, Q. Digital sliding-mode control of piezoelectric micropositioning system based on input–output model. *IEEE Trans. Ind. Electron.* **2014**, *61*, 5517–5526.
31. Thai, N.H.; Minh, P.X.; Son, H.M.; Dan, N.C.; Quyet, H.G. Robust Adaptive Control of Robots Using Neural Network and Sliding Mode Control. In Proceedings of the IEEE International Conference on Control, Automation and Information Sciences, NhaTrang, Vietnam, 25–28 November 2013; pp. 322–327.
32. Cat, P.T.; Hiep, N.T. Robust PID sliding mode control of robot manipulators with online learning neural network. In Proceedings of the European Control Conference, Budapest, Hungary, 23–26 August 2009; pp. 2187–2192.
33. Tang, W.Q.; Cai, Y.L. A High Order Sliding Mode Control Scheme Based on Adaptive Radial Basis Function Neural Network. In Proceedings of the IEEE Conference on Decision and Control and European Control Conference, Orlando, FL, USA, 12–15 December 2011; pp. 6343–6348.
34. Liu, Y.; Jia, Y.M. Adaptive consensus protocol for networks of multiple agents with nonlinear dynamics using neural networks. *Asian J. Control* **2012**, *14*, 1328–1339. [[CrossRef](#)]
35. Li, Z.K.; Duan, Z.S.; Chen, G.R.; Huang, L. Consensus of Multiagent Systems and Synchronization of Complex Networks: A Unified Viewpoint. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2010**, *57*, 213–224.
36. Zhao, L.; Huang, J.; Liu, H.; Li, B.; Kong, W. Second-order sliding-mode observer with online parameter identification for sensorless induction motor drives. *IEEE Trans. Ind. Electron.* **2014**, *61*, 5280–5289. [[CrossRef](#)]

37. Vieira, R.P.; Gastaldini, C.C.; Azzolin, R.Z.; Grundling, H.A. Sensorless sliding-mode rotor speed observer of induction machines based on magnetizing current estimation. *IEEE Trans. Ind. Electron.* **2014**, *61*, 4573–4582. [[CrossRef](#)]
38. Bernardes, T.; Montagner, V.F.; Gründling, H.A.; Pinheiro, H. Discrete-time sliding mode observer for sensorless vector control of permanent magnet synchronous machine. *IEEE Trans. Ind. Electron.* **2014**, *61*, 1679–1691. [[CrossRef](#)]
39. Wang, Y.P.; Cheng, L.; Zeng, G.H.; Yu, Z.J.; Tan, M. Optimal Formation of Multirobot Systems Based on a Recurrent Neural Network. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *27*, 322–333. [[CrossRef](#)] [[PubMed](#)]
40. Dai, S.L.; Wang, C.; Wang, M. Dynamic Learning From Adaptive Neural Network Control of a Class of Nonaffine Nonlinear Systems. *IEEE Trans. Neural Netw. Learn. Syst.* **2014**, *25*, 111–123. [[PubMed](#)]
41. Zhao, L.; Jia, Y.M. Neural network-based adaptive consensus tracking control for multi-agent systems under actuator faults. *Int. J. Syst. Sci.* **2016**, *47*, 1931–1942. [[CrossRef](#)]
42. Liu, P.; Zeng, Z.G.; Wang, J. Complete stability of delayed recurrent neural networks with Gaussian activation functions. *Neural Netw.* **2016**, *85*, 21–32. [[CrossRef](#)] [[PubMed](#)]



© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).