Received September 14, 2017, accepted October 30, 2017, date of publication November 15, 2017,
date of current version December 22, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2773544

Neural Networks for the Output Tracking-Control Problem of Nonlinear Strict-Feedback System

YUHUAN CHEN[1,2], JIHUA REN[3], AND CHENGFU YI[4,5]
[1]Gannan Normal University, Ganzhou 341000, China
[2]Shenzhen University, Shenzhen 518060, China
[3]Beijing University of Technology, Beijing 100124, China
[4]Nanchang Institute of Technology, Nanchang 330044, China
[5]Jiangxi University of Science and Technology, Ganzhou 341000, China

Corresponding author: Chengfu Yi (itchenve@gmail.com)

This work was supported in part by the National Science Foundation of China under Grant 11561029 and in part by programs under Grant GJJ14428 and Grant jxxj11048.

ABSTRACT This paper focuses on the tracking-control problem of nonlinear strict-feedback system by utilizing neural networks. Combining a novel recurrent neural network and gradient-based neural network, we investigate, develop and design a new controller based on the synthesized neural network model (N–G model) to track the output trajectory performance of the nonlinear strict-feedback system. This presented control scheme could have a good output tracking performance for the nonlinear strict-feedback system. For comparing with the presented N-G model, the classic backstepping design method is also employed to design the control input for the nonlinear strict-feedback control system in this paper. The computer simulation results demonstrate that the controller based on the N–G model could be used to tackle the tracking-control problem with accuracy and effectiveness, together with the faster convergent speed than that based on the backstepping algorithm. Generally speaking, with the appropriate increase of design parameters, the controller based on the N–G model could improve convergence performance for nonlinear strict-feedback system.

INDEX TERMS Output tracking-control, nonlinear strict-feedback system, recurrent neural network, backstepping algorithm.

I. INTRODUCTION

It is well known that nonlinear system has a wide range of applications in the fields of engineering and science, especially in control systems, such as digital control [1], adaptive control [2], navigation system [3], and the remote control system [4]. As a significant subject of nonlinear system in control fields, the output tracking-control problem emerges frequently in practical engineering, such as mobile robot control [5], [6], flight control [7] and motor control [8]. The main objective of tracking-control is for the design of controller to be able to tracking the desired output trajectory. Generally speaking, there exist many design methods for the controller of output tracking-control in nonlinear control areas [9]–[19], where many design methods are developed based on the classic backstepping algorithm [12]–[19].

In the 1990s, Kanellakopoulos *et al.*'s proposed a recursive design procedure, named as adaptive backstepping (i.e., backstepping algorithm), to track the control problem of strict-feedback systems with parameters [12]. This approach could avoid some restrictions such as matching condition, extended matching condition or growth conditions on system nonlinearities being made to guarantee the global stability [13], [14]. In an attempt to extend the research of backstepping idea, Krstic *et al.* [15] further investigated the backstepping idea for parametric strict-feedback systems with unknown virtual control coefficients. It has been proven that the design methods based on the backstepping algorithm can guarantee the global stabilities, tracking and transient performance for a class of strict-feedback systems [15]. In [16] and [17], the authors have illustrated that this design method is applicable to non-linear systems as well, particularly is effective for the strict-feedback nonlinear systems with parameters. Recently, the adaptive backstepping has been studied to design a speed controller for a novel hybrid excitation synchronous machine with nonlinear coupling and parametric uncertainty [18]. In [19], an intelligent backstepping tracking control system is developed for wheel inverted pendulum with unknown system dynamic and external disturbance.

As an alternative method for the controller design of output tracking-control, the in-depth research has been carried out for the neural network (NN) control methods [20], [21]. For example, an adaptive multiple neural-network control with a supervisory controller is developed for a class of uncertain nonlinear systems in [20]. With the help of a supervisory controller, the resulting closed-loop system is globally stable in the sense that all signals involved are uniformly bounded and the Lyapunov function-based design of adaptation laws guarantees the global stability of the closed-loop system. Recently, Liu *et al*. have proposed an efficient neural network approach to tackle the tracking-control problem of autonomous surface vehicle (ASV) [21]. The NN approach forces the ASV to track the desired trajectory with good control performance through the on-line learning of the NN without any off-line learning procedure. Especially, many scholars have already exhibited their interests in the combination of backstepping design and neural network methods to obtain the nonlinear system controller [17], [22], [23]. For example, by combining adaptive neural network design with backstepping methodology, an integral-type Lyapunov function is investigated and plays an important role in conquering the singularity problem [17].

In this paper, we investigate and develop a new neural network approach to deal with the controller design problem for the output tracking-control of nonlinear strict-feedback system. This method is a combination of a novel recurrent neural network (NRNN, also named as ZNN) proposed by Guo and Zhang [24] and a traditional based-gradient neural network (GNN) [25]. These two methods are often used for the online solution of linear/nonlinear matrix equation problem [24]–[26]. Through synthesizing NRNN and GNN models, a new neural network, i.e., N-G model, is developed and utilized to design the controller in the form of time-derivative for the nonlinear strict-feedback system. In order to facilitate the comparison and analysis with the design process of the presented N-G model, the classic backstepping design approach is also employed to design the controller for the same nonlinear strict-feedback system. Simulation studies illustrate that the problem of such a controller design could be solved efficiently by the presented N-G model. Through the comparative analysis of output trajectory performance and tracking error, we can have the conclusion that our presented N-G model could be used to design a controller to tackle the tracking-control problem effectively and exactly, and moreover, it can obtain a superior convergence performance than that based on the backstepping algorithm.

The rest of this paper is organized into four sections. Section 2 firstly presents the problem description of nonlinear control system briefly, and then introduce the simple development of the N-G model to obtain the controller for solving the output tracking-control problem of the nonlinear strict-feedback system. In Section 3, we provide the specific procedure of the controller based on the classic backstepping design for the same nonlinear system. For illustrative and comparative purpose, the two controllers are employed to simulate and analyze for the tracking performance and convergence performance of nonlinear strict-feedback in Section 4. The simulation results show that the controller based on our presented N-G model could track the desired output trajectory effectively and exactly. Section 5 concludes this paper with final remarks.

II. CONTROLLER BASED ON N-G MODEL

In this section, we firstly introduce the problem description of output tracking problem in nonlinear strict-feedback system. In order to get the controller based on the novel recurrent neural networks (NRNN, i.e., ZNN) for the nonlinear strict-feedback system, inspired by Guo and Zhang [24], Yi *et al*. [25] [26], Zhang and Ge [27], and Yi and Liu [28] research work, a novel recurrent neural network (NRNN) is presented for the controller design in the form of $\dot{u}(t)$ utilized to solve the output tracking-control problem of 2nd-order nonlinear strict-feedback system, together with the gradient-based neural network (GNN).

A. PROBLEM DESCRIPTION

In general, the key purpose for the tracking-control problem is to design a input controller, i.e., $u(t)$, to make the actual output $y(t)$ of a nonlinear system able to track the desired trajectory $y_d(t)$ effectively, or to make the tracking error able to keep within a permissible error range as soon as possible for the system. The control principle can be shown in Fig. 1. As a classic solution scheme for the controller of the nonlinear strick-feedback systems, the computational method based on Backstepping algorithm could avoid the matching conditions of satisfying the nonlinear function constraints [13], and mainly be employed to solve a class of so-called lower-triangular structure system control problems, which no longer meet with the certain restrictions usually used to step up the relations between the states in the nonlinear differential equations for nonlinear systems [22], [29]. The typical representatives of such control systems are strict-feedback system and output-feedback system. These two kinds of systems have been attracted much attention because the general nonlinear system satisfying some geometric conditions could be transformed into them by using the diffeomorphism method [30]. In this paper, combining NRNN and GNN models, we investigate and develop a new design method based on neural networks to achieve the controller for the output tracking-control problem of nonlinear strict-feedback system written as follows. Note that, For the convenience of description, the argument t will be omitted in some places.

$$\begin{cases} \dot{x}_1 = x_2 + f_1(x_1) \\ \dot{x}_2 = x_3 + f_2(x_1, x_2) \\ \quad \vdots \\ \dot{x}_n = u + f_n(x_1, x_2, \cdots, x_n) \\ y = x_1 \end{cases} \tag{1}$$

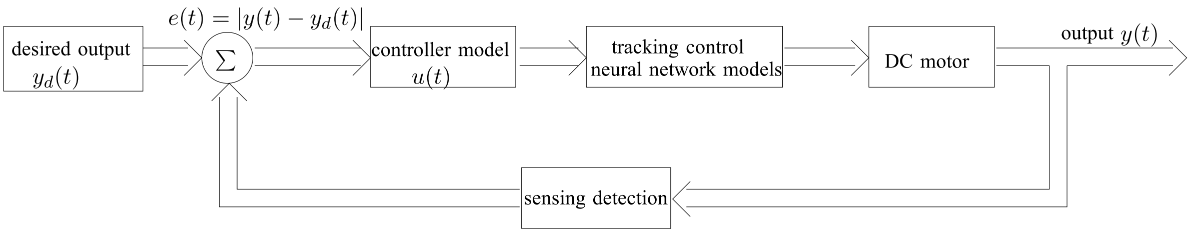


FIGURE 1. The control principle of nonlinear system.

where $x = [x_1, x_2, \cdots, x_n]^T \in R^n$, $u \in R$, and $y \in R$ are the state variables, system control law (or system controller) and output, respectively; $f_i(\cdot)$ $(i = 1, 2, \cdots, n)$ is the unknown smooth function and may not be linearly parameterized. The reason for referring to the system (1) as "strict-feedback" is that the nonlinearities $f(\cdot)$ in the x_i-equation $(i = 1, 2, \cdots, n)$ depend only on $x_1, x_2, \cdots, x_n$, that is, on state variables that are "fed back" [15]. The control objective is to design a system controller $u(t)$ for system (1) such that the system actual output $y(t)$ can track the given desired trajectory $y_d(t)$.

B. DEVELOPMENT OF NRNN AND GNN MODELS

According to Guo and Zhang [24], Yi *et al.* [25] [26], and Zhang and Ge's [27] design idea, a vector-valued indefinite error function $Z \in R^n$ is firstly constructed for NRNN model. In order to make each-element $z_i \in R(i = 1, 2, 3, \cdots, n)$ in the vector-valued error function $Z \in R^n$ be able to converge to zero (in mathematics), we have the following design formula of NRNN.

$$\dot{Z}(t) = \frac{dZ(t)}{dt} := -\gamma \mathcal{F}(Z(t)), \tag{2}$$

where the design parameter (or learning rate) $\gamma > 0 \in R$ is used to scale the convergence rate for the NRNN solution [27], $\mathcal{F}(\cdot)$: $R^n \to R^n$ denotes an activation-function array of neural model. Generally speaking, there are four commonly used activation-functions, such as linear, power, sigmoid and power-sigmoid functions [25]. Studies have shown that arbitrary monotonically increasing excitation function can be used to construct the network model to improve the convergence performance [26], [27]. In the past related research work, NRNN is usually applied to solve the time-varying matrix equation problems, such as matrix inversion [27] and linear/nonlinear equation solving [25], [26].

As for the traditional classical gradient-based neural network (GNN), it is usually designed based on a scalar-valued non-negative energy function $\varepsilon(u(t))$ [26]. Evolving along a decent direction resulting from the negative gradient of such energy function, we could obtain the following GNN mathematical expression:

$$\dot{u}(t) := -\mu \frac{\partial \varepsilon(u(t))}{\partial u(t)}, \tag{3}$$

where the design parameter (or learning rate) $\mu > 0 \in R$ is used to scale the convergence rate of the gradient dynamic solution [26]. Similar to the NRNN design formula, we could also add an activation function $\mathcal{F}(\cdot)$ to improve the convergence performance. The GNN model (3) was originally designed for constant (i.e., time-invariant) problems solving [25]. Recent researches have shown that the neural models based on such a traditional gradient search algorithm could not solve the time-varying problems exactly and effectively. It only approaches approximately to the time-varying theoretical solution [25], [26].

C. CONTROLLER DESIGN BY N-G MODEL

In this subsection, by synthesizing the above-presented NRNN and GNN models, we can obtain a N-G model, which can be extended and applied to design a new nonlinear controller $u(t)$ for the strict-feedback system (1). Differing from the traditional controller in the form of $u(t)$, the N-G model is exploited in this paper to solve the tracking-control problem of the nonlinear system in the form of $\dot{u}(t)$ (i.e., the time-derivative of $u(t)$), which can be referred as in formula (6). To present conveniently the design process, the linear activation function is utilized in NRNN model (2), and GNN model (3) is activated by the power-sigmoid activation function. The design steps for controller are listed as follows.

Step 1: From system (1), the 1st NRNN error function z_1 is constructed as follows:

$$z_1 = y - y_d = x_1 - y_d.$$

Combining the NRNN design formula (2), i.e., $\dot{z}_1 = -\gamma z_1$, we have

$$\dot{x_1} - \dot{y_d} = -\gamma(x_1 - y_d).$$

Step 2: we also construct the following 2nd NRNN error function z_2:

$$z_2 = \dot{z}_1 + \gamma z_1.$$

By applying the formula (2) again, we have the time-derivative of z_2:

$$\dot{z}_2 := -\gamma z_2 = \ddot{z}_1 + \gamma \dot{z}_1.$$

Step 3: Similarly, we can construct the 3rd NRNN error function z_3:

$$z_3 = \dot{z}_2 + \gamma z_2$$

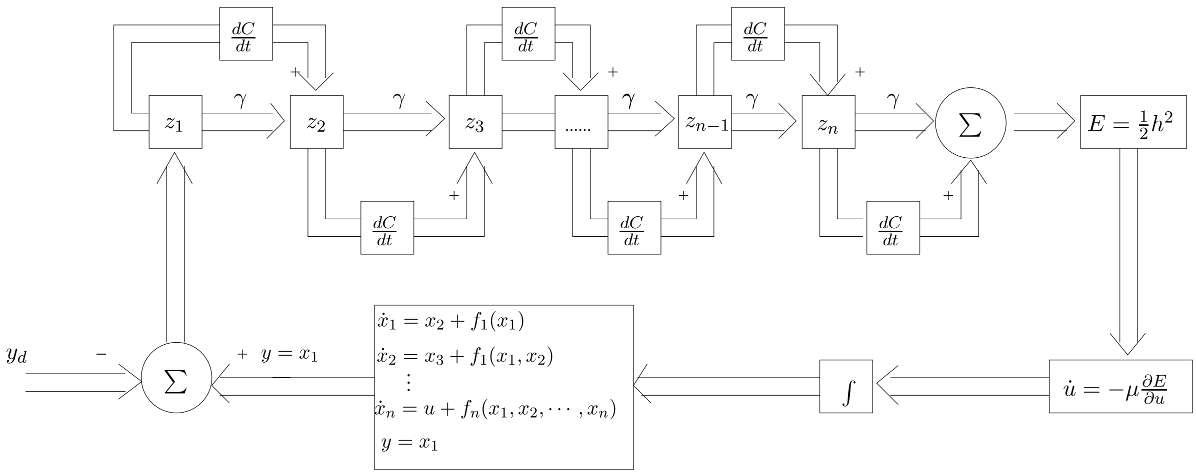


FIGURE 2. The controller for the nonlinear strict-feedbcak system (1) based on N-G model.

By substituting $\dot z_2$ into the time-derivative of z_3, we have

$$\dot z_3 := -\gamma z_3 = \dddot{z}_1 + \gamma \ddot z_1 + \gamma \dot z_2.$$

$$\vdots$$

Step n: Therefore, based on the NRNN design formula (2), we can construct the nth-order NRNN error function z_n and the time derivative of z_n (i.e., $\dot z_n$), which are written as the following expressions:

$$\begin{cases} z_n = \dot z_{n-1} + \gamma z_{n-1} \\ = z_1^{(n-1)} + \gamma z_1^{(n-2)} + \gamma z_2^{(n-3)} \\ + \cdots + \gamma \dot z_{(n-2)} + \gamma z_{(n-1)}, \\ \dot z_n := -\gamma z_n. \end{cases} \tag{4}$$

Then, differentiating the first sub-equation in (4), we can obtain the following general formula of $\dot z_n$.

$$\begin{aligned} \dot z_n &= z_1^{(n)} + \gamma z_1^{(n-1)} + \gamma z_2^{(n-2)} \\ &\quad + \gamma z_3^{(n-3)} + \cdots + \gamma z_{n-2}^{(2)} + \gamma \dot z_{n-1}. \end{aligned}$$

By the second sub-equation in (4), let $h := \dot z_n + \gamma z_n$, which should be zero theoretically, then we could obtain the expression:

$$\begin{aligned} h &= z_1^{(n)} + \gamma z_1^{(n-1)} + \gamma z_2^{(n-2)} + \gamma z_3^{(n-3)} \\ &\quad + \cdots + \gamma z_{n-2}^{(2)} + \gamma \dot z_{n-1} + \gamma\big(z_1^{(n-1)} + \gamma z_1^{(n-2)} \\ &\quad + \gamma z_2^{(n-3)} + \cdots + \gamma \dot z_{n-2} + \gamma z_{n-1}\big). \end{aligned} \tag{5}$$

In the last step, by following GNN (3), we firstly define the energy function $\varepsilon(h(t)) := h^2(t)$ and add the power-sigmoid activation function $\mathcal{F}(\cdot)$. Then, the new controller model in the form of $\dot u(t)$ for nonlinear system (1) can be designed as below.

$$\dot u = -\mu \frac{\partial \varepsilon}{\partial u} = -2\mu \mathcal{F}(h). \tag{6}$$

Note that, if we can encounter u before the nth step, we don't have to design n steps. That is, once we get the equation containing u during the design process, we could stop using the aforementioned NRNN design formula, and then apply GNN (3) directly to achieve the controller $u(t)$. The block diagram can be seen in Fig. 2.

D. ANALYTICAL EXAMPLE

In order to analyze the design steps of N-G method for the controller $u(t)$ exhaustively, in this subsection, we would present a specific analytical example of the 2nd-order nonlinear strict-feedback system as depicted below.

$$\begin{cases} \dot x_1 = x_2 + f_1(x_1) \\ \dot x_2 = u + f_2(x_1, x_2) \\ y = x_1. \end{cases} \tag{7}$$

According to the design procedure in Section (II-C), we could get the following design steps for the 2nd-order nonlinear strict-feedback system (7).

At first, the 1st NRNN error function z_1 is defined as:

$$z_1 = y - y_d = x_1 - y_d \in R.$$

By following the NRNN model (2), substituting (7) into the above equation, we could get

$$\dot x_1 - \dot y_d = x_2 + f_1(x_1) - \dot y_d = -\gamma(x_1 - y_d).$$

Then, we could construct the 2nd NRNN error function as follows:

$$z_2 = \dot z_1 + \gamma z_1 = x_2 + f_1(x_1) - \dot y_d + \gamma(x_1 - y_d).$$

Thus, we have the time-derivative of z_2 (i.e., $\dot z_2$) as follows.

$$\begin{aligned} \dot z_2 &= \dot x_2 + \dot f_1(x_1) - \ddot y_d + \gamma(\dot x_1 - \dot y_d) \\ &= \dot x_2 + \dot x_1 \frac{\partial f_1(x_1)}{\partial x_1} - \ddot y_d + \gamma(\dot x_1 - \dot y_d) \\ &= u + f_2(x_1, x_2) + \dot x_1 \frac{\partial f_1(x_1)}{\partial x_1} - \ddot y_d + \gamma(\dot x_1 - \dot y_d). \end{aligned} \tag{8}$$

Evidently, Eq. (8) contains the control input u. Therefore, we can employ GNN model (3) directly to obtain the controller $u(t)$. As the above mentioned in Section II-C, let h define as:

$$\begin{aligned} h &:= \dot{z}_2 + \gamma z_2 \\ &= u + f_2(x_1, x_2) + \dot{x}_1 \frac{\partial f_1(x_1)}{\partial x_1} - \ddot{y}_d + \gamma(\dot{x}_1 - \dot{y}_d) \\ &\quad + \gamma\big(x_2 + f_1(x_1) - \dot{y}_d + \gamma(x_1 - y_d)\big), \end{aligned} \tag{9}$$

which should be zero theoretically.

Therefore, by applying GNN (3), the novel controller in the form of $\dot{u}$ for output tracking-control of the 2nd-order nonlinear strict-feedback system (7) is ultimately designed as below:

$$\dot{u} = -2\mu\mathcal{F}(h), \tag{10}$$

where the energy function ε in (3) is defined as $\varepsilon := h^2$, and the power-sigmoid activation function $\mathcal{F}(\cdot)$ is adopted for the better convergence performance.

III. CONTROLLER BASED ON BACKSTEPPING ALGORITHM

The Backstepping design idea was firstly presented in [15]. As a traditional classic recursive algorithm, the Backstepping method has been widely applied to design the controller for the nonlinear strict-feedback system in much literature [17]–[19], [23]. In this section, in comparison with our presented N-G model, the Backstepping algorithm is also used for the controller design of nonlinear system (7). In fact, the design essence of Backstepping algorithm is to select some appropriate functions of state variables recursively as pseudo/virtual control inputs for lower dimension subsystems of the overall system. During the design process, an intermediate virtual function α_i shall be developed by using an appropriate Lyapunov function V_i. Next, we would like to present a summary of nth-order Backstepping design procedure.

Firstly, we define error variables $z_1 = y - y_d$ and $z_2 = x_2 - \alpha_1$, where α_1 is a virtual control law to be derived as follows. Hence, the dynamic equation of $\dot{z}_1$ is given by

$$\dot{z}_1 = \dot{y} - \dot{y}_d = x_2 + f_1(x_1) - \dot{y}_d,$$

Define the following Lyapunov function candidate

$$V_1 = \frac{1}{2} z_1^2.$$

By combining $x_2 = z_2 + \alpha_1$, then the time-derivative of V_1 is written as:

$$\begin{aligned} \dot{V}_1 = z_1\dot{z}_1 &= z_1(x_2 + f_1(x_1) - \dot{y}_d) \\ &= z_1(z_2 + \alpha_1 + f_1(x_1) - \dot{y}_d) \\ &= z_1 z_2 + z_1(\alpha_1 + f_1(x_1) - \dot{y}_d). \end{aligned}$$

In order to make the Lyapunov function V_1 stable, let $-k_1 z_1 = \alpha_1 + f(x_1) - \dot{y}_d$ with the design parameter $k_1 > 0$. Thus, we have

$$\dot{V}_1 = -k_1 z_1^2 + z_1 z_2,$$

where $z_1 z_2$ would be processed in the next step.

Therefore, we have

$$\alpha_1 = -k_1 z_1 - f_1(x_1) + \dot{y}_d.$$

Secondly, since $z_2 = x_2 - \alpha_1$, we could get its time-derivative given by

$$\begin{aligned} \dot{z}_2 = \dot{x}_2 - \dot{\alpha}_1 &= u + f_2(x_1, x_2) - \frac{\partial \alpha_1}{\partial x_1}\big(x_2 + f_1(x_1)\big) \\ &\quad - \frac{\partial \alpha_1}{\partial y_d}\dot{y}_d - \frac{\partial \alpha_1}{\partial \dot{y}_d}\ddot{y}_d. \end{aligned}$$

Consider the following Lyapunov candidate function:

$$V_2 = V_1 + \frac{1}{2} z_2^2.$$

Then, we have

$$\begin{aligned} \dot{V}_2 &= \dot{V}_1 + z_2\dot{z}_2 = -k_1 z_1^2 + z_2(z_1 + \dot{z}_2) \\ &= -k_1 z_1^2 + z_2\Big(z_1 + u + f_2(x_1, x_2) \\ &\quad - \frac{\partial \alpha_1}{\partial x_1}\big(x_2 + f_1(x_1)\big) - \frac{\partial \alpha_1}{\partial y_d}\dot{y}_d - \frac{\partial \alpha_1}{\partial \dot{y}_d}\ddot{y}_d\Big). \end{aligned}$$

In order to make $\dot{V}_2 < 0$ for the stability of $\dot{z}_2$, we define $-k_2 z_2 = z_1 + u + f_2(x_1, x_2) - \frac{\partial \alpha_1}{\partial x_1}\big(x_2 + f_1(x_1)\big) - \frac{\partial \alpha_1}{\partial y_d}\dot{y}_d - \frac{\partial \alpha_1}{\partial \dot{y}_d}\ddot{y}_d$. Then, the control input $u(t)$ (i.e., the controller model of the 2nd-order nonlinear strict-feedback system (7)) is given by

$$\begin{aligned} u(t) &= -z_1 - k_2 z_2 - f_2(x_1, x_2) \\ &\quad + \frac{\partial \alpha_1}{\partial x_1}\big(x_2 + f_1(x_1)\big) + \frac{\partial \alpha_1}{\partial y_d}\dot{y}_d + \frac{\partial \alpha_1}{\partial \dot{y}_d}\ddot{y}_d. \end{aligned} \tag{11}$$

Similarly, for the general nth-order nonlinear strict-feedback system, we could obtain the generalized design formula for the controller $u(t)$ as follows.

Firstly, the error variant(s) should be defined based on the coordinate transformations

$$z_1 = y - y_d,$$

and

$$z_i = x_i - \alpha_{i-1}, i = 2, \cdots, n-1.$$

Besides, choose the suitable virtual control functions

$$\alpha_1 = -k_1 z_1 - f_1(x_1) + \dot{y}_d,$$

and

$$\begin{aligned} \alpha_{i-1} &= -z_{i-1} - k_i z_i - f_i(x_1, x_2, \cdots, x_i) \\ &\quad + \sum_{k=1}^{i-1} \frac{\partial \alpha_{i-1}}{\partial x_k}\big(x_{k+1} + f_k(x_1, x_2, \cdots, x_k)\big) \\ &\quad + \sum_{k=0}^{i-1} \frac{\partial \alpha_{i-1}}{\partial y_d^{(k)}} y_d^{(k+1)}, (i = 2, \cdots, n-1). \end{aligned}$$

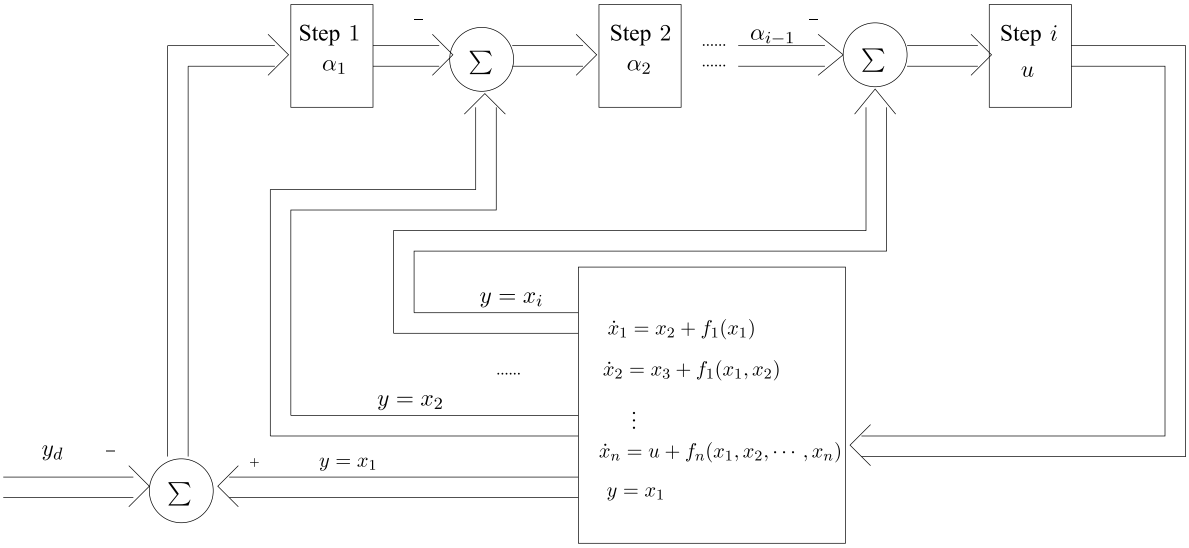


FIGURE 3. The controller for the nonlinear strict-feedback system (1) based on Backstepping algorithm.

Constructing the Lyapunov function $V_n = \sum_{j=1}^{n} \frac{1}{2}(z_j)^2$, we could obtain the following expression:

$$
\begin{aligned}
\dot{V}_n &= -\sum_{j=1}^{n-1} k_j(z_j)^2 + z_{n-1}z_n + z_n\dot{z}_n \\
&= \sum_{j=1}^{n-1} k_j(z_j)^2 + z_n\Big(z_{n-1} + f_n + u \\
&\quad -\sum_{k=1}^{i-1} \frac{\partial \alpha_{i-1}}{\partial x_k}\big(x_{k+1} + f_k(x_1, x_2, \cdots, x_k)\big) \\
&\quad -\sum_{k=0}^{i-1} \frac{\partial \alpha_{i-1}}{\partial y_d^{(k)}} y_d^{(k+1)}\Big).
\end{aligned}
$$

In order to make $\dot{V}_n < 0$, we let

$$
\begin{aligned}
-k_n z_n = z_{n-1} + f_n + u - \sum_{k=0}^{i-1} \frac{\partial \alpha_{i-1}}{\partial y_d^{(k)}} y_d^{(k+1)} \\
- \sum_{k=1}^{i-1} \frac{\partial \alpha_{i-1}}{\partial x_k}\big(x_{k+1} + f_k(x_1, x_2, \cdots, x_k)\big).
\end{aligned}
$$

Hence, we could achieve the controller for the nth-order nonlinear strict-feedback system as follows:

$$
\begin{aligned}
u = -z_{n-1} - k_n z_n - f_n + \sum_{k=0}^{i-1} \frac{\partial \alpha_{i-1}}{\partial y_d^{(k)}} y_d^{(k+1)} \\
+ \sum_{k=1}^{i-1} \frac{\partial \alpha_{i-1}}{\partial x_k}\big(x_{k+1} + f_k(x_1, x_2, \cdots, x_k)\big), \quad (12)
\end{aligned}
$$

and the block diagram can be shown in Fig. 3.

IV. ILLUSTRATIVE SIMULATION RESULTS

As for the output tracking-control problem fo nonlinear strict-feedback styem, we have presented the design procedures of controller by using two types of methods (i.e., N-G model and Backstepping algorithm) in Sections II and III, respectively. In this section, we will make use of MATLAB simulation techniques to verify the correctness of the presented N-G model for achieving the controller of the following nonlinear strict-feedback system presented in [23].

$$
\begin{cases}
\dot{x}_1 = x_2 + x_1 \sin x_1 \\
\dot{x}_2 = u + 0.2x_1x_2^2 \\
y = x_1.
\end{cases}
\quad (13)
$$

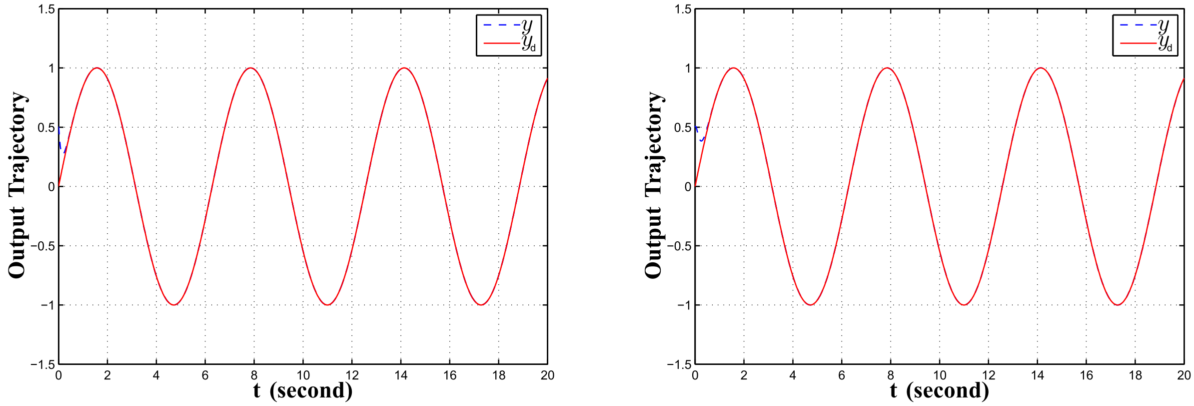
Through the N-G model (5) and (6), we can get the controller in the form of the time-derivative of $u(t)$ as below:

$$
\begin{aligned}
\dot{u}(t) = -2\mu\mathcal{F}\Big(&0.2x_1x_2^2 + u + (x_2 + x_1 \sin x_1) \\
&(\sin x_1 + x_1 \cos x_1) - \ddot{y}_d \\
&+ \gamma(x_2 + x_1 \sin x_1 - \dot{y}_d) \\
&+ \gamma\big(x_2 + x_1 \sin x_1 - \dot{y}_d + \gamma(x_1 - y_d)\big)\Big). \quad (14)
\end{aligned}
$$

In addition, by using Backstepping method (12), we get the controller as follows:

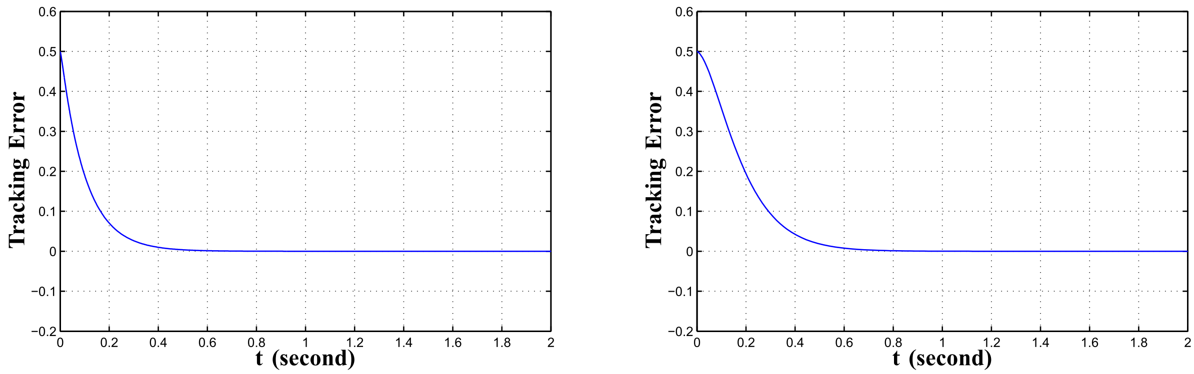
$$
\begin{aligned}
u = -(&k_1 + k_2 + \sin x_1 + x_1 \cos x_1) \\
&(x_2 + x_1 \sin x_1) - 0.2x_1x_2^2 - k_1k_2x_1 \\
&+ \ddot{y}_d + (k_1 + k_2)\dot{y}_d - (1 + k_1k_2)(x_1 - y_d). \quad (15)
\end{aligned}
$$

To illustrate the aforementioned analytical results that the N-G model could be used to solve the tracking-control problem in the form of $\dot{u}(t)$, simulations of tracking the desired output $y_d = \sin(t)$ [23] are performed for the 2nd-order nonlinear system (13) equipped with N-G model controller (14). During the simulation process, the design



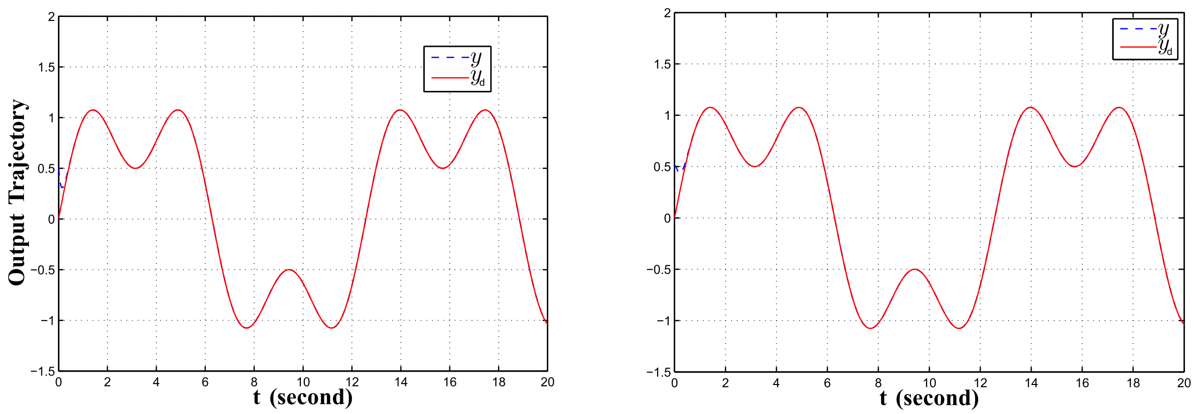
(a) (b)

FIGURE 4. The output performance of nonlinear strict-feedback system (13) tracking the desired trajectory $y_d = \sin(t)$. (a) Output trajectory by N-G model (6) with $\gamma = \mu = 10$. (b) Output trajectory by Backstepping formula (12) with $k_1 = k_2 = 10$.



(a) (b)

FIGURE 5. Output tracking error performance of nonlinear system (13) for the desired trajectory $y_d = \sin(t)$. (a) Tracking error $|y - y_d|$ by the controller (6) with $\gamma = \mu = 10$. (b) Tracking error $|y - y_d|$ by the controller (12) with $k_1 = k_2 = 10$.



(a) (b)

FIGURE 6. The output performance of nonlinear strict-feedback system for the desired trajectory $y_d = \sin(0.5t) + 0.5\sin(1.5t)$. (a) Output trajectory by the controller (6) with $\gamma = \mu = 10$. (b) Output trajectory by the controller (12) with $k_1 = k_2 = 10$.

parameters $\gamma = \mu = 10$. In order to comparative purpose, the controller by Backstepping (15) is also used to solve the output tracking-control problem of the same nonlinear system (13) with design parameters $k_1 = k_2 = 10$. Besides, the initial conditions are selected as $x_1(0) = x_2(0) = 0.5$ and $u(0) = 0$.

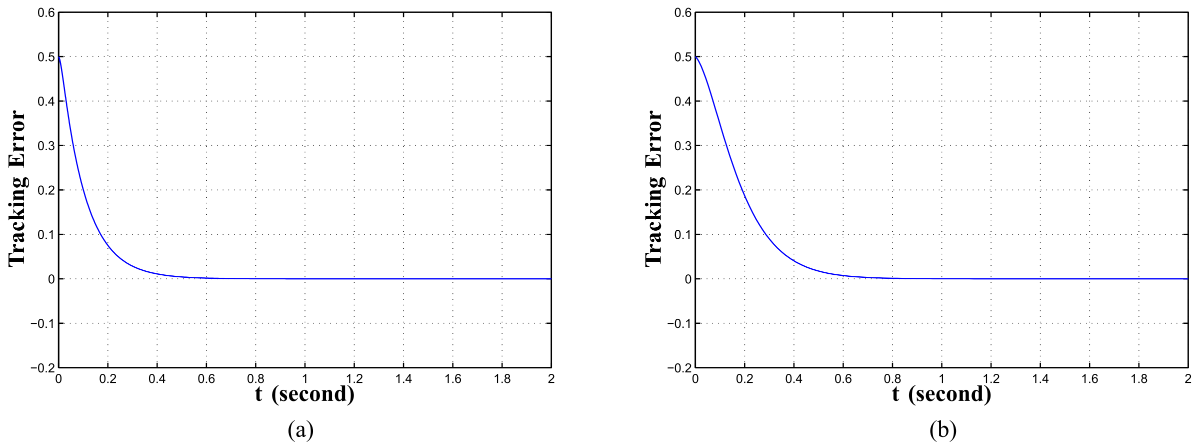


FIGURE 7. Output tracking error performance of nonlinear system for the desired trajectory $y_d = \sin(0.5t) + 0.5\sin(1.5t)$ with $\gamma = \mu = 10$ and $k_1 = k_2 = 10$. (a) Tracking error $|y - y_d|$ by using (6). (b) Tracking error $|y - y_d|$ by (12).

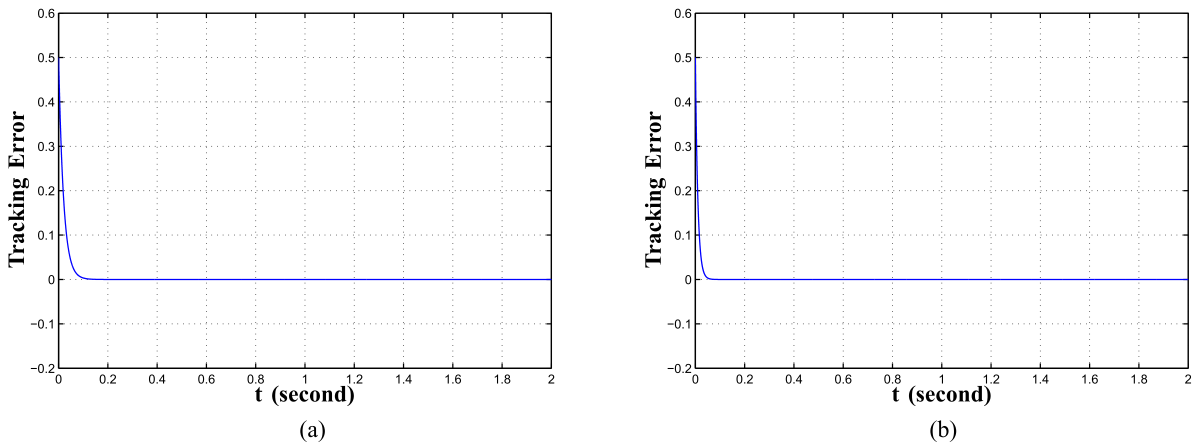


FIGURE 8. Tracking error $|y - y_d|$ of nonlinear strict-feedback system by the controller (6) based on N-G model for the desired trajectory $y_d = \sin(t)$. (a) Tracking error with $\gamma = \mu = 50$. (b) Tracking error with $\gamma = \mu = 100$.

When the controllers (14) and (15) are used to track $y_d(t) = \sin(t)$, we can get the output trajectory of the nonlinear strict-feedback system (13) as shown in Fig. 4, where the solid red curves correspond to the desired trajectory $y_d(t)$ and the dotted blue curves correspond to the actual output trajectory $y(t)$. Note that, the controller (14) is activated by the power-sigmoid activation function. As shown in Fig. 4, starting from the given initial values, the output trajectory generated by the above two controllers can track the desired objective $y_d(t)$ with accuracy and effectiveness. This demonstrates that the N-G model could be used to design the controller for tracking the tracking-control problem of nonlinear strict-feedback system exactly and efficaciously. However, the derivations of the controller based on N-G model is relatively simpler than those based on the Backstepping algorithm.

In addition, we can use the tracking error $|y-y_d|$ to monitor the convergence performance. Figure 5 shows the tracking error performances of the controllers based on N-G model and Backstepping, respectively. After 0.6s or so, the tracking error $|y - y_d|$ by (14) is convergent to zero, which is shown in Fig. 5(a), while it needs 0.8s to converge to zero for the controller (15) from Fig. 5(b). That is to say, the presented controller based on the N-G method has superior convergence performance than that based on the Backstepping algorithm.

To further substantiate the accuracy and validity of the presented controller based on the new N-G model, another desired trajectory, i.e., $y_d = \sin(0.5t) + 0.5\sin(1.5t)$ presented in [17], is to be tracked by using the above-mentioned two controllers. Figs. 6 and 7 show the simulation results of tracking the desired trajectory of the nonlinear strict-feedback system via the controller u obtained by our presented N-G model and Backstepping algorithm, respectively. In this simulation, we apply the power-sigmoid activation and set the design parameter $\gamma = \mu = 10$, $k_1 = k_2 = 10$. In addition, the running time is 20s and the initial value is $x_1(0) = x_2(0) = 0.5$ and $u(0) = 0$. It demonstrates that both controllers (i.e., (6) and (12)) could be used to track the desired trajectory precisely. Therefore, as seen from Figs. 4(a) and 6(a), the presented controller based on the N-G model can track the different desired trajectories for the nonlinear strict-feedback system. Specially, the convergence of (6) is superior than that of (12), as shown in Fig. 7.

Note that, if the design parameters are set as different values, we can achieve different convergence performance.

Figs. 8(a) and (b) exhibit the tracking performance and convergence characteristics of the controller (6) with the design parameters $\gamma = \mu = 50$ and $\gamma = \mu = 100$. Comparing and analyzing Figs. 5(a), 8(a) and 8(b), we can conclude that, through increasing the values of design parameters, the convergence performance could be improved by using the controller (6).

In summary, the above simulation results have further illustrated that the reliability and the effectiveness of our presented controller based on the N-G model for solving tracking-control problem of nonlinear strict-feedback system and the faster convergence rate than that based on the Backstepping algorithm.

V. CONCLUSIONS

In this paper, combining the novel neural networks by Zhang and the traditional classic gradient-based algorithm, we design and investigate a control input $u(t)$ for the nonlinear strict-feedback system. For comparison purposes, the classic Backstepping algorithm is also used to exploit the controller for the same nonlinear system. There exist some different points for the two models, such as the derivation complexity and exhibited formula form. The computer simulation results further shows that our presented controller can track the desired trajectories effectively and exactly. Furthermore, in comparison with the controller based on the Backstepping algorithm, the tracking error by the our presented controller is convergent to zero with faster speed than that based on the Backstepping algorithm.

VI. ACKNOWLEDGEMENTS

The authors would like to thank the editors and the anonymous reviewers for their valuable suggestions and constructive comments to help to improve the quality and the presentation of this paper. *(Yuhuan Chen and Chengfu Yi contributed equally to this work).*

REFERENCES

[1] I. Bessa, H. Ismail, R. Palhares, L. Cordeiro, and J. E. C. Filho, "Formal non-fragile stability verification of digital control systems with uncertainty," *IEEE Trans. Comput.*, vol. 66, no. 3, pp. 545–552, Mar. 2017.

[2] R. Sharma, F. Zare, D. Nešic, and A. Ghosh, "A hidden block in a grid connected active front end system: Modelling, control and stability analysis," *IEEE Access*, to be published.

[3] Y. Shi, H. Li, H. Fang, and N. Kato, "Guest editorial focused section on advanced control and navigation for marine mechatronic systems," *IEEE/ASME Trans. Mechatronics*, vol. 22, no. 3, pp. 1117–1120, Jun. 2017.

[4] J.-I. Furukawa, T. Noda, T. Teramae, and J. Morimoto, "Human movement modeling to detect biosignal sensor failures for myoelectric assistive robot control," *IEEE Trans. Robot.*, vol. 33, no. 4, pp. 846–857, Aug. 2017.

[5] C.-F. Juang, T.-L. Jeng, and Y.-C. Chang, "An interpretable fuzzy system learned through online rule generation and multiobjective ACO with a mobile robot control application," *IEEE Trans. Cybern.*, vol. 46, no. 12, pp. 2706–2718, Dec. 2016.

[6] S. Pathak, L. Pulina, and A. Tacchella, "Evaluating probabilistic model checking tools for verification of robot control policies," *AI Commun.*, vol. 29, no. 2, pp. 287–299, 2016.

[7] K. Alexis, C. Papachristos, R. Siegwart, and A. Tzes, "Robust model predictive flight control of unmanned rotorcrafts," *J. Intell. Robot. Syst.*, vol. 81, nos. 3–4, pp. 443–469, 2016.

[8] A. Gentsch, A. Weber, M. Synofzik, G. Vosgerau, and S. Schütz-Bosbach, "Towards a common framework of grounded action cognition: Relating motor control, perception and cognition," *Cognition*, vol. 146, pp. 81–89, Jan. 2016.

[9] S. Song, X. Qiu, J. Wang, and M. Q.-H. Meng, "Real-time tracking and navigation for magnetically manipulated untethered robot," *IEEE Access*, vol. 4, pp. 7104–7110, 2016.

[10] C.-C. Tsai, H.-C. Huang, and S.-C. Lin, "Adaptive neural network control of a self-balancing two-wheeled scooter," *IEEE Trans. Ind. Electron.*, vol. 57, no. 4, pp. 1420–1428, Apr. 2010.

[11] N. Wang, J.-C. Sun, and M.-J. Er, "Global adaptive practical output tracking control for a class of genuinely nonlinear uncertain systems: Adding an universal power integrator approach," *IEEE Access*, vol. 4, pp. 10136–10146, 2016.

[12] I. Kanellakopoulos, P. V. Kokotovic, and A. S. Morse, "Systematic design of adaptive controllers for feedback linearizable systems," *IEEE Trans. Autom. Control*, vol. AC-36, no. 11, pp. 1241–1253, Nov. 1991.

[13] I. Kanellakopoulos, P. V. Kokotovic, and R. Marino, "Robustness of adaptive nonlinear control under an extended matching condition," in *Proc. IFAC Symp. Nonlinear Control Syst. Design*, 1989, pp. 192–197.

[14] D. G. Taylor, P. V. Kokotovic, R. Marino, and I. Kannellakopoulos, "Adaptive regulation of nonlinear systems with unmodeled dynamics," *IEEE Trans. Autom. Control*, vol. 34, no. 4, pp. 405–412, Apr. 1989.

[15] M. Krstic, I. Kanellakopoulos, and P. V. Kokotovic, *Nonlinear and Adaptive Control Design*. New York, NY, USA: Wiley, 1995.

[16] H. F. Ho, Y. K. Wong, and A. B. Rad, "Adaptive fuzzy approach for a class of uncertain nonlinear systems in strict-feedback form," *ISA Trans.*, vol. 47, no. 3, pp. 286–299, 2008.

[17] 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.

[18] L. Sheng, G. Xiaojie, and Z. Lanyong, "Robust adaptive backstepping sliding mode control for six-phase permanent magnet synchronous motor using recurrent wavelet fuzzy neural network," *IEEE Access*, vol. 5, pp. 14502–14515, 2017.

[19] C. L. P. Chen, G.-X. Wen, Y.-J. Liu, and Z. Liu, "Observer-based adaptive backstepping consensus tracking control for high-order nonlinear semi-strict-feedback multiagent systems," *IEEE Trans. Cybern.*, vol. 46, no. 7, pp. 1591–1601, Jul. 2016.

[20] C.-Y. Lee and J.-J. Lee, "Adaptive control for uncertain nonlinear systems based on multiple neural networks," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 1, pp. 325–333, Feb. 2004.

[21] Y. Liu, J. Yu, H. Yu, C. Lin, and L. Zhao, "Barrier Lyapunov functions-based adaptive neural control for permanent magnet synchronous motors with full-state constraints," *IEEE Access*, vol. 5, pp. 10382–10389, 2017.

[22] B. Chen, H. Zhang, and C. Lin, "Observer-based adaptive neural network control for nonlinear systems in nonstrict-feedback form," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 1, pp. 89–98, Jan. 2016.

[23] Y. Pan, T. Sun, Y. Liu, and H. Yu, "Command-filtered backstepping neural network control with composite learning," *Neural Netw.*, vol. 95, pp. 134–142, Aug. 2017.

[24] D. Guo and Y. Zhang, "Novel recurrent neural network for time-varying problems solving [research frontier]," *IEEE Comput. Intell. Mag.*, vol. 7, no. 4, pp. 61–65, Nov. 2012.

[25] C. Yi, Y. Chen, and X. Lan, "Comparison on neural solvers for the Lyapunov matrix equation with stationary & nonstationary coefficients," *Appl. Math. Model.*, vol. 37, no. 4, pp. 2495–2502, 2013.

[26] C. Yi, D. Guo, and Y. Zhang, "A new type of recurrent neural networks for real-time solution of Lyapunov equation with time-varying coefficient matrices," *Math. Comput. Simul.*, vol. 92, pp. 40–52, Jun. 2013.

[27] Y. Zhang and S. S. Ge, "Design and analysis of a general recurrent neural network model for time-varying matrix inversion," *IEEE Trans. Neural Netw.*, vol. 16, no. 6, pp. 1477–1490, Nov. 2005.

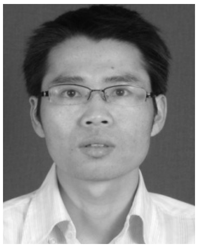
[28] C. F. Yi and Y. H. Liu, "Online solution of time-varying Lyapunov matrix equation by Zhang neural networks," *Recent Patents Comput. Sci.*, vol. 6, no. 1, pp. 25–32, 2013.

[29] R. Ma and J. Zhao, "Backstepping design for global stabilization of switched nonlinear systems in lower triangular form under arbitrary switchings," *Automatica*, vol. 46, no. 11, pp. 1819–1823, 2010.

[30] M. Zamani and P. Tabuada, "Backstepping design for incremental stability," *IEEE Trans. Autom. Control*, vol. 56, no. 9, pp. 2184–2189, Sep. 2011.

YUHUAN CHEN received the B.S. degree from Gannan Normal University, Ganzhou, China, in 2003, and the M.S. degree from the Jiangxi University of Science and Technology, Ganzhou, China, in 2007. She is currently pursuing the Ph.D. degree with the College of Information Engineering, Shenzhen University, Shenzhen, China. Her current research interests include intelligence information processing, neural network, saliency detection, object segmentation, and pattern recognition.

JIHUA REN was born in Jiangxi, China, in 1975. He received the B.S. and M.S. degrees from the Jiangxi University of Science and Technology in 2000 and 2012, respectively, where he is currently pursuing the Ph.D. degree. His current research interests include mathematical modeling and rotation of gear and intelligence information processing.



CHENGFU YI was born in Jiangxi, China, in 1978. He received the B.S. degree in communication engineering from Northeast Dianli University, Jilin, China, in 2001, the M.S. degree in computer applied technology from the Jiangxi University of Science and Technology, Ganzhou, China, in 2007, and the Ph.D. degree in communication and information systems from Sun Yat-Sen University, Guangzhou, China. He is currently a Teacher with the Nanchang Institute of Science and Technology and the Jiangxi University of Science and Technology. His current research interests include intelligent information processing and neural networks.