International Journal of Bifurcation and Chaos, Vol. 12, No. 5 (2002) 1097–1109

UNCERTAIN CHAOTIC SYSTEM CONTROL VIA ADAPTIVE NEURAL DESIGN

S. S. GE* and C. WANG†
Department of Electrical & Computer Engineering, National University of Singapore, Singapore 117576
†*Department of Electronic Engineering, City University of Hong Kong, Hong Kong SAR, P.R. China*
**elegesz@nus.edu.sg*

Received December 30, 2000; Revised October 1, 2001

Though chaotic behaviors are exhibited in many simple nonlinear models, physical chaotic systems are much more complex and contain many types of uncertainties. This paper presents a robust adaptive neural control scheme for a class of uncertain chaotic systems in the disturbed strict-feedback form, with both unknown nonlinearities and uncertain disturbances. To cope with the two types of uncertainties, we combine backstepping methodology with adaptive neural design and nonlinear damping techniques. A smooth singularity-free adaptive neural controller is presented, where nonlinear damping terms are used to counteract the disturbances. The differentiability problem in controlling the disturbed strict-feedback system is solved without employing norm operation, which is usually used in robust control design. The proposed controllers can be applied to a large class of uncertain chaotic systems in practical situations. Simulation studies are conducted to verify the effectiveness of the scheme.

Keywords: Robust adaptive NN design; uncertain chaotic systems; chaos control.

1. Introduction

Over the past decade, there have been tremendous interests in the study of controlling chaotic systems in various fields of science. One of the reasons for the interests is that chaotic behaviors have been discovered in many nonlinear dynamical systems in physics, chemistry, biology, engineering, and many other disciplines. It is believed that the study of controlling chaos will lead to a wide range of potential applications (see [Chen, 1999] and the references therein).

It is noticed that though chaotic behaviors can arise from many simple nonlinear models, e.g. Brusselator [Nicolis & Prigogine, 1977], Lorenz system [Lorenz, 1963], Chua's circuits [Chua *et al.*, 1986], etc., the actual systems described by these models may be much more complex in practice. For example, the Brusselator is used to model a certain set of chemical reactions [Nicolis & Prigogine, 1977], while the Lorenz system is a highly simplified model of a convecting fluid [Lorenz, 1963]. Both Brusselator and Lorenz system are derived from partial differential equations (PDE) after a series of approximations. Therefore, there must exist modeling errors in the two models. The physical systems of the chemical reactions or convecting fluid should be much more complicated than these models. In other words, the practical chaotic systems may contain many types of uncertainties, such as unknown parameters, unknown nonlinearities, exogenous

*Author for correspondence.

disturbances, unmodeled dynamics, etc. Thus, it is important and necessary to design robust controllers for uncertain chaotic systems with various types of uncertainties.

As an alternative, in recent years, adaptive neural control (ANC) has received much attention and become an active research area. ANC is a nonlinear control methodology which is particularly useful for the control of highly uncertain, nonlinear and complex systems (see [Lewis *et al.,* 1999; Ge *et al.,* 2001] and the references therein). In adaptive neural control design, neural networks are mostly used as approximators for unknown nonlinear functions in system models. With the help of NN approximation, it is not necessary to spend much effort on system modeling in case such a modeling is difficult. In the earlier neural control schemes, optimization techniques were mainly used to derive parameter adaptation laws. One of the main disadvantages of these schemes lies in the lack of analytical results for stability and performance of the control systems. To overcome these problems, some adaptive neural control approaches based on Lyapunov's stability theory have been proposed for nonlinear systems with certain types of matching conditions[1] [Polycarpou & Ioannou, 1992; Sanner & Slotine, 1992; Chen & Liu, 1994; Rovithakis & Christodoulou, 1994; Ge *et al.*, 1998; Yesidirek & Lewis, 1995; Spooner & Passino, 1996]. By using the idea of adaptive backstepping design [Krstic *et al.*, 1995], several adaptive neural controllers [Polycarpou & Mears, 1998; Lewis *et al.*, 2000; Ge *et al.*, 2001; Ge & Wang, 2001] have been proposed for uncertain nonlinear systems in strict-feedback form without the requirement of matching conditions. In these adaptive neural control approaches, the update laws for the weights of the neural networks are derived based on Lyapunov synthesis method, rather than the optimization techniques used in the earlier neural controllers. The main advantages of adaptive neural design include that the neural weights are tuned online without the training phase, and the stability and performance of the closed-loop systems can be readily guaranteed.

Since adaptive neural control is suitable for the control of uncertain nonlinear systems, it is naturally a good candidate for controlling uncertain chaotic systems. In the literature, Qin *et al.* [1999] proposed a Gaussian RBF NN based adaptive controller for a class of uncertain chaotic systems with rigorous mathematical analysis by the Lyapunov stability theory. This result is applicable only to the uncertain chaotic systems satisfying the matching conditions, e.g. the Duffing oscillator, and cannot be applied to many other chaotic systems, such as Brusselator [Nicolis & Prigogine, 1977]. In the research of chaos, some chaotic systems have been extensively studied and have been taken as benchmark examples due to their rich dynamical behaviors. These examples include: Duffing oscillator [Duffing, 1918], van der Pol oscillator [van der Pol, 1927], Brusselator [Nicolis & Prigogine, 1977], Rössler system [Rössler, 1976], and Chua's circuit [Chua *et al.*, 1986]. It has been shown [Ge *et al.*, 2000] that all of these chaotic systems can be rewritten into the strict-feedback form. Thus, the adaptive neural control approaches [Polycarpou & Mears, 1998; Ge *et al.,* 2001; Ge & Wang, 2001] provide the theoretical possibilities for controlling these chaotic systems subjected to uncertain nonlinear functions.

In this paper, we consider adaptive neural control of uncertain chaotic systems in a more general class of strict-feedback form

$$
\begin{aligned}
\dot{x}_i &= f_i(\overline{x}_i) + g_i(\overline{x}_i)x_{i+1} \\
&\qquad + \phi_i(\overline{x}_i)d_i\,, \quad 1 \le i \le n-1 \\
\dot{x}_n &= f_n(\overline{x}_n) + g_n(\overline{x}_{n-1})u \\
&\qquad + \phi_n(\overline{x}_n)d_n\,, \quad n \ge 2 \\
y &= x_1
\end{aligned} \tag{1}
$$

where $\overline{x}_i = [x_1, \ldots, x_i]^T \in R^i$, $i = 1, \ldots, n$, $u \in R$, $y \in R$ are state variables, system input and output, respectively, f_i and $g_i \neq 0$ are unknown nonlinear smooth functions (note that $g_n(\overline{x}_{n-1})$ is assumed to be independent of state x_n), $\phi_i(\cdot)$ is an unknown smooth function of $(x_1, \ldots, x_i)$, and d_i is the uniformly bounded disturbance input with its bound not necessarily known. Here, the disturbance input d_i is allowed to take any form of uniformly bounded terms, including unmodeled states and external disturbances. The model (1) can be used to depict the physical chaotic systems subjected to various types of uncertainties.

It is noticed that when $d_i = 0$, the result in [Ge & Wang, 2001] can be applied directly to the control

[1]Matching condition means that the uncertainties enter through the same channels as the control inputs in a state space representation.

of uncertain chaotic systems in the strict-feedback form (1). When $d_i \neq 0$, however, it is shown [Krstic *et al.*, 1995] that the existence of disturbance terms $\phi_i(\overline{x}_i)d_i$ might drive the system states to escape to infinity in a finite time, even if d_i is an exponentially decaying disturbance. Thus, these disturbance terms have to be taken into account in the robust control design. For the control of the disturbed strict-feedback system (1), while a great deal of progress has been achieved when the affine terms are known, see, e.g. [Freeman & Kokotovic, 1996; Pan & Basar, 1998; Jiang & Praly, 1998; Freeman *et al.*, 1998], only a few results are available in the literature [Gong & Yao, 2001; Arslan & Basar, 2001] when the affine terms are unknown nonlinearities. The difficulties in controlling the disturbed strict-feedback system (1) include:

(i) when the affine terms g_i $(i = 1, \ldots, n)$ are unknown, if feedback linearization type controllers $\alpha_i = (1/\hat{g}_i(\cdot))(-\hat{f}_i(\cdot) + \upsilon_i)$ are considered in backstepping design procedures, where $\hat{f}_i(\cdot)$ and $\hat{g}_i(\cdot)$ are the estimates of $f_i(\cdot)$ and $g_i(\cdot)$, respectively, and υ_i is a new control to be defined, controller singularity problem arises when $\hat{g}_i(\cdot) \to 0$;

(ii) Due to the differentiation of the virtual controls in backstepping design, the derivatives of virtual controls will involve the uncertainties $f_i(\cdot)$, $g_i(\cdot)$ and $\phi_i(\cdot)d_i$, and thus are unknown and unavailable for implementation;

(iii) To cope with the disturbance term $\phi_i(\cdot)d_i$, which is usually assumed to be bounded in Euclidean norm, robust control often contains norm operation. Since the Euclidean norm is not differentiable with respects to its arguments, differentiability becomes a technical problem in the robust adaptive neural design.

To overcome the controller singularity problem, projection algorithm is often employed to keep the approximations of the affine terms bounded away from zero. The disadvantage of using projection is that, it usually requires *a priori* knowledge for the feasible parameter set and no systematic procedure is available for constructing such a set for a general plant [Ioannou & Sun, 1995]. In this paper, by using NNs to approximate the unknown nonlinear functions in the controller, rather than the unknown nonlinearities $f_i(\cdot)$ and $g_i(\cdot)$, the controller singularity problem is avoided since there is no need to explicitly estimate the affine terms g_i in the controller. On the other hand, to satisfy the differentiation requirement in backstepping design, we need to combine backstepping methodology with adaptive neural design and nonlinear damping techniques. A smooth adaptive neural controller is proposed, where nonlinear damping terms are used in the controller design to counteract the disturbances. The adaptive neural control scheme achieves semi-global uniform ultimate boundedness of all the signals in the closed-loop. The output of the system is proven to converge to a neighborhood of the desired trajectory. The control performance of the closed-loop system is guaranteed by suitably choosing the design parameters. The proposed controller can be applied to a large class of uncertain chaotic systems in practical situations. Simulation studies are conducted to verify the effectiveness of the approach.

The rest of the paper is organized as follows: The problem formulation is presented in Sec. 2. In Sec. 3, a robust adaptive neural controller is presented for controlling uncertain nonlinear system (1). Simulation results performed on uncertain Brusselator model are included to show the effectiveness of the approach in Sec. 4. Section 5 contains the conclusions.

2. Problem Formulation

The control objective is to design a direct adaptive NN controller for system (1) such that (i) all the signals in the closed-loop remain uniformly semi-globally ultimately bounded, and (ii) the output y follows a desired trajectory y_d generated from the following smooth, bounded reference model

$$\begin{aligned} \dot{x}_{di} &= f_{di}(x_d)\,, \quad 1 \le i \le m \\ y_d &= x_{d1}\,, \quad m \ge n \end{aligned} \tag{2}$$

where $x_d = [x_{d1}, x_{d2}, \ldots, x_{dm}]^T \in R^m$ are the states, $y_d \in R$ is the system output, $f_{di}(\cdot)$, $i = 1, 2, \ldots, m$ are known smooth nonlinear functions. Assume that the states of the reference model remain bounded, i.e. $x_d \in \Omega_d$, $\forall t \ge 0$.

In the derivation of the adaptive neural controller, NN approximation is only guaranteed within some compact sets. Accordingly, the stability results obtained in this work are semi-global in the sense that, as long as the input variables of the NNs remain within some compact sets, where the compact sets can be made as large as desired, there exists controller(s) with sufficiently large number of NN nodes such that all the signals in the closed-loop remain bounded.

We make the following assumptions for the uncertainties in system (1), which will be used throughout the paper.

Assumption 1. The signs of $g_i(\cdot)$ are known, and there exist constants $g_{i1} \geq g_{i0} > 0$ such that $g_{i1} \geq |g_i(\cdot)| \geq g_{i0}$, $\forall \overline{x}_n \in \Omega \subset R^n$.

The above assumption implies that smooth functions $g_i(\cdot)$ are strictly either positive or negative. Without losing generality, we shall assume $g_{i1} \geq g_i(\overline{x}_i) \geq g_{i0} > 0$, $\forall \overline{x}_n \in \Omega \subset R^n$.

Assumption 2. There exist constants $g_{id} > 0$ such that $|\dot{g}_i(\cdot)| \leq g_{id}$, $\forall \overline{x}_n \in \Omega \subset R^n$.

Assumption 3. $|\phi_i(\overline{x}_i)| \leq \varphi_i(\overline{x}_i)$, where $\varphi_i(\overline{x}_i)$ are known smooth nonlinear functions.

Assumption 4. $|d_i| \leq d_i^*$, with d_i^* not necessarily known.

In control engineering, Radial Basis Function (RBF) neural network (NN) is usually used as a tool for modeling nonlinear functions because of their good capabilities in function approximation. The RBF NN can be considered as a two-layer network in which the hidden layer performs a fixed nonlinear transformation with no adjustable parameters, i.e. the input space is mapped into a new space. The output layer then combines the outputs in the latter space linearly. Therefore, they belong to a class of linearly parameterized networks. In this paper, the following RBF NN [Haykin, 1999] is used to approximate the continuous function $h(Z) : R^q \to R$,

$$h_{nn}(Z) = W^T S(Z) \tag{3}$$

where the input vector $Z \in \Omega \subset R^q$, weight vector $W = [w_1, w_2, \ldots, w_l]^T \in R^l$, the NN node number $l > 1$; and $S(Z) = [s_1(Z), \ldots, s_l(Z)]^T$, with $s_i(Z)$ being chosen as the commonly used Gaussian functions, which have the form

$$s_i(Z) = \exp\left[\frac{-(Z-\mu_i)^T(Z-\mu_i)}{\eta_i^2}\right], \quad i = 1, 2, \ldots, l \tag{4}$$

where $\mu_i = [\mu_{i1}, \mu_{i2}, \ldots, \mu_{iq}]^T$ is the center of the receptive field and η_i is the width of the Gaussian function.

It has been proven that network (3) can approximate any continuous function over a compact set $\Omega_Z \subset R^q$ to arbitrary any accuracy as

$$h(Z) = W^{*T} S(Z) + \varepsilon\,, \quad \forall Z \in \Omega_Z \tag{5}$$

where W^* is ideal constant weights, and ε is the approximation error.

Assumption 5. There exist ideal constant weights W^* such that $|\varepsilon| \leq \varepsilon^*$ with constant $\varepsilon^* > 0$ for all $Z \in \Omega_Z$.

The ideal weight vector W^* is an "artificial" quantity required for analytical purposes. W^* is defined as the value of W that minimizes $|\varepsilon|$ for all $Z \in \Omega_Z \subset R^q$, i.e.

$$W^* \triangleq \arg \min_{W \in R^l} \left\{ \sup_{Z \in \Omega_Z} |h(Z) - W^T S(Z)| \right\} \tag{6}$$

3. Robust Adaptive Neural Control Design

In this section, we present the robust adaptive neural control design for uncertain chaotic systems in strict-feedback form (1) with disturbance terms $\phi_i(\cdot)d_i$ $(i = 1, \ldots, n)$. By combining backstepping methodology with adaptive neural design, as well as nonlinear damping technique, we extend our previous result in [Ge & Wang, 2001], and propose a smooth adaptive neural controller for system (1). The nonlinear damping terms are used in the controller design to counteract the disturbances $\phi_i(\cdot)d_i$.

The detailed design procedure is described in the following steps. For clarity and conciseness of presentation, Steps 1 and 2 are described with detailed explanations, while Step i $(i = 3, \ldots, n)$ is simplified, with redundant equations and explanations being omitted.

Step 1. Define $z_1 = x_1 - x_{d1}$. Its derivative is

$$\dot{z}_1 = f_1(x_1) + g_1(x_1)x_2 + \phi_1(x_1)d_1 - \dot{x}_{d1}$$

By viewing x_2 as a virtual control input, there exists a desired feedback control

$$\begin{aligned}\alpha_1^* &= -c_1 z_1 - \frac{1}{g_1(x_1)}[f_1(x_1) + \phi_1(x_1)d_1 - \dot{x}_d] \\ &= -c_1 z_1 - \frac{1}{g_1(x_1)}[f_1(x_1) - \dot{x}_d] - \frac{\phi_1(x_1)d_1}{g_1(x_1)}\end{aligned} \tag{7}$$

where $c_1 > 0$ is a design constant.

Since $f_1(x_1)$, $g_1(x_1)$ and $\phi_1(x_1)d_1$ are unknown, the desired feedback control α_1^* cannot be implemented in practice. In Eq. (7), two type of uncertainties exist, one is the unknown nonlinearity $h_1(Z_1) \triangleq (1/g_1(x_1))[f_1(x_1) - \dot{x}_d]$, which is a continuous function of x_1 and $\dot{x}_d$. Thus, $h_1(Z_1)$ can be approximated by an RBF neural network $W_1^T S_1(Z_1)$, i.e.

$$h_1(Z_1) = W_1^{*T} S_1(Z_1) + \varepsilon_1 \tag{8}$$

where $Z_1 \triangleq [x_1, \dot{x}_d]^T \subset R^2$, W_1^* denotes the ideal constant weights, and $|\varepsilon_1| \le \varepsilon_1^*$ is the approximation error with constant $\varepsilon_1^* > 0$. The other uncertainty is the disturbance term $(\phi_1(x_1)d_1)/(g_1(x_1))$, where the disturbance input d_1 is bounded but unavailable through measurement. This term cannot be approximated by neural networks. To cope with this uncertainty, a nonlinear damping term $-p_1\varphi_1^2(x_1)z_1$ ($p_1 > 0$) is introduced to counteract the disturbance term, as will be explained in the following procedure.

Since x_2 is only taken as a virtual control, not as the real control input for the z_1-subsystem, by introducing the error variable $z_2 = x_2 - \alpha_1$, the practical virtual control α_1 is chosen as

$$\alpha_1 = -c_1 z_1 - \hat{W}_1^T S_1(Z_1) - p_1\varphi_1^2(x_1)z_1 \tag{9}$$

where $\hat{W}_1$ is the estimate of the neural network to be tuned online. The $\dot{z}_1$ equation becomes

$$\begin{aligned}\dot{z}_1 &= g_1(x_1)\left[z_2 + \alpha_1 + \frac{1}{g_1(x_1)}(f_1(x_1) - \dot{x}_d)\right] + \phi_1(x_1)d_1 \\ &= g_1(x_1)[z_2 - c_1 z_1 - \tilde{W}_1^T S_1(Z_1) + \varepsilon_1 - p_1\varphi_1^2(x_1)z_1] + \phi_1(x_1)d_1 \end{aligned}\tag{10}$$

Consider the following Lyapunov function candidate

$$V_1 = \frac{1}{2g_1(x_1)} z_1^2 + \frac{1}{2}\tilde{W}_1^T \Gamma_1^{-1}\tilde{W}_1 \tag{11}$$

where $\Gamma_1 = \Gamma_1^T > 0$ is an adaptation gain matrix.

The derivative of V_1 is

$$\begin{aligned}\dot{V}_1 &= \frac{z_1\dot{z}_1}{g_1} - \frac{\dot{g}_1 z_1^2}{2g_1^2} + \tilde{W}_1^T\Gamma_1^{-1}\dot{\tilde{W}}_1 \\ &= \frac{z_1}{g_1}[g_1(z_2 - c_1 z_1 - \tilde{W}_1^T S_1(Z_1) + \varepsilon_1 - p_1\varphi_1^2 z_1) + \phi_1 d_1] - \frac{\dot{g}_1}{2g_1^2} z_1^2 + \tilde{W}_1^T\Gamma_1^{-1}\dot{\tilde{W}}_1 \\ &= z_1 z_2 - c_1 z_1^2 - \frac{\dot{g}_1}{2g_1^2} z_1^2 + z_1\varepsilon_1 + \tilde{W}_1^T\Gamma_1^{-1}[\dot{\hat{W}}_1 - \Gamma_1 S_1(Z_1)z_1] - p_1\varphi_1^2 z_1^2 + \frac{\phi_1 d_1}{g_1} z_1 \end{aligned}\tag{12}$$

Consider the following adaptation law

$$\dot{\hat{W}}_1 = \dot{\tilde{W}}_1 = \Gamma_1[S_1(Z_1)z_1 - \sigma_1\hat{W}_1] \tag{13}$$

where $\sigma_1 > 0$ is a small constant.

Let $c_1 = c_{10} + c_{11}$, with c_{10} and $c_{11} > 0$. Because of the following inequalities

$$\begin{aligned} -\sigma_1\tilde{W}_1^T\hat{W}_1 &= -\sigma_1\tilde{W}_1^T(\tilde{W}_1 + W_1^*) \le -\sigma_1\|\tilde{W}_1\|^2 + \sigma_1\|\tilde{W}_1\|\,\|W_1^*\| \\ &\le -\frac{\sigma_1\|\tilde{W}_1\|^2}{2} + \frac{\sigma_1\|W_1^*\|^2}{2} \\ -c_{11}z_1^2 + z_1\varepsilon_1 &\le -c_{11}z_1^2 + z_1|\varepsilon_1| \le \frac{\varepsilon_1^2}{4c_{11}} \le \frac{\varepsilon_1^{*2}}{4c_{11}} \\ -\left(c_{10} + \frac{\dot{g}_1}{2g_1^2}\right) z_1^2 &\le -\left(c_{10} - \frac{g_{1d}}{2g_{10}^2}\right) z_1^2 \\ -p_1\varphi_1^2 z_1^2 + \frac{\phi_1 d_1}{g_1} z_1 &\le -p_1\phi_1^2 z_1^2 + \frac{\phi_1 z_1}{g_{10}} d_1 \le \frac{d_1^2}{4p_1 g_{10}^2} \le \frac{d_1^{*2}}{4p_1 g_{10}^2} \end{aligned}\tag{14}$$

we have

$$\dot{V}_1 = z_1 z_2 - \left(c_{10} + \frac{\dot{g}_1}{2g_1^2}\right) z_1^2 - c_{11} z_1^2 + z_1 \varepsilon_1 - \sigma_1 \tilde{W}_1^T \hat{W}_1 - p_1 \varphi_1^2 z_1^2 + \frac{\phi_1 d_1}{g_1} z_1$$

$$< z_1 z_2 - c_{10}^* z_1^2 - \frac{\sigma_1 \|\tilde{W}_1\|^2}{2} + \frac{\sigma_1 \|W_1^*\|^2}{2} + \frac{\varepsilon_1^{*2}}{4c_{11}} + \frac{d_1^{*2}}{4p_1 g_{10}^2} \tag{15}$$

where c_{10} is chosen such that $c_{10}^* \triangleq c_{10} - (g_{1d}/2g_{10}^2) > 0$.

Step 2. The derivative of $z_2 = x_2 - \alpha_1$ is

$$\dot{z}_2 = f_2(\overline{x}_2) + g_2(\overline{x}_2) x_3 + \phi_2(\overline{x}_2) d_2 - \dot{\alpha}_1 \tag{16}$$

From Eq. (9), it can be seen that α_1 is a function of x_1, x_d and $\hat{W}_1$. Thus, $\dot{\alpha}_1$ is given by

$$\dot{\alpha}_1 = \frac{\partial \alpha_1}{\partial x_1} \dot{x}_1 + \frac{\partial \alpha_1}{\partial x_d} \dot{x}_d + \frac{\partial \alpha_1}{\partial \hat{W}_1} \dot{\hat{W}}_1$$

$$= \frac{\partial \alpha_1}{\partial x_1} (g_1 x_2 + f_1 + \phi_1 d_1) + \psi_1 \tag{17}$$

where $\psi_1 = (\partial \alpha_1 / \partial x_d) \dot{x}_d + (\partial \alpha_1 / \partial \hat{W}_1)[\Gamma_1 (S_1(Z_1) z_1 - \sigma_1 \hat{W}_1)]$ is introduced as an intermediate variable, which is computable. Different from the $\dot{x}_{d1}$ in Step 1, $\dot{\alpha}_1$ is unknown because it contains both unknown nonlinear functions and a disturbance term. However, it does not matter as the unknown terms can be elegantly handled in the next step.

By viewing x_3 as a virtual control input to stabilize the (z_1, z_2)-subsystem, there exists a desired feedback control

$$\alpha_2^* = -z_1 - c_2 z_2 - \frac{1}{g_2}(f_2 - \dot{\alpha}_1) - \frac{\phi_2}{g_2} d_2$$

$$= -z_1 - c_2 z_2 - \frac{1}{g_2}\left[f_2 - \frac{\partial \alpha_1}{\partial x_1}(g_1 x_2 + f_1) - \psi_1\right]$$

$$+ \frac{\partial \alpha_1}{\partial x_1} \frac{\phi_1}{g_2} d_1 - \frac{\phi_2}{g_2} d_2 \tag{18}$$

where c_2 is a positive constant to be specified later.

Two types of uncertainties exist in Eq. (18). To deal with the uncertain nonlinear functions in Eq. (18), an RBF neural network $W_2^T S_2(Z_2)$ is employed to approximate the unknown nonlinearity denoted as $h_2(Z_2)$, i.e.

$$h_2(Z_2) \triangleq \frac{1}{g_2}\left[f_2 - \frac{\partial \alpha_1}{\partial x_1}(g_1 x_2 + f_1) - \psi_1\right]$$

$$= W_2^{*T} S_2(Z_2) + \varepsilon_2 \tag{19}$$

where $Z_2 \triangleq [\overline{x}_2^T, (\partial \alpha_1 / \partial x_1), \psi_1]^T \subset R^4$, W_2^* is the ideal constant weights, and $|\varepsilon_2| \leq \varepsilon_2^*$ is the approximation error with constant $\varepsilon_2^* > 0$. To counteract the uncertainty including the disturbance inputs d_1 and d_2, the nonlinear damping term $-p_{21}((\partial \alpha_1 / \partial x_1)\varphi_1)^2 z_2 - p_2 \varphi_2^2 z_2$ is introduced, where p_{21}, $p_2 > 0$ are design constants. Define the error variable $z_3 = x_3 - \alpha_2$. The virtual control α_2 is chosen as

$$\alpha_2 = -z_1 - c_2 z_2 - \hat{W}_2^T S_2(Z_2) - p_{21}\left(\frac{\partial \alpha_1}{\partial x_1}\varphi_1\right)^2 z_2 - p_2 \varphi_2^2 z_2 \tag{20}$$

Thus, the $\dot{z}_2$ equation becomes

$$\dot{z}_2 = g_2\left[z_3 + \alpha_2 + \frac{1}{g_2}\left(f_2 - \frac{\partial \alpha_1}{\partial x_1}(g_1 x_2 + f_1) - \psi_1\right)\right.$$

$$\left. - \frac{\partial \alpha_1}{\partial x_1} \frac{\phi_1}{g_2} d_1 + \frac{\phi_2}{g_2} d_2\right]$$

$$= g_2\left[z_3 - z_1 - c_2 z_2 - \tilde{W}_2^T S_2(Z_2) + \varepsilon_2\right.$$

$$- p_{21}\left(\frac{\partial \alpha_1}{\partial x_1}\varphi_1\right)^2 z_2 - p_2 \varphi_2^2 z_2$$

$$\left. - \frac{\partial \alpha_1}{\partial x_1} \frac{\phi_1}{g_2} d_1 + \frac{\phi_2}{g_2} d_2\right] \tag{21}$$

Consider the Lyapunov function candidate

$$V_2 = V_1 + \frac{1}{2g_2(\overline{x}_2)} z_2^2 + \frac{1}{2} \tilde{W}_2^T \Gamma_2^{-1} \tilde{W}_2 \tag{22}$$

where $\Gamma_2 = \Gamma_2^T > 0$ is an adaptation gain matrix. The derivative of V_2 is

$$\dot{V}_2 = \dot{V}_1 + \frac{z_2 \dot{z}_2}{g_2} - \frac{\dot{g}_2 z_2^2}{2g_2^2} + \tilde{W}_2^T \Gamma_2^{-1} \dot{\hat{W}}_2$$

$$= \dot{V}_1 - z_1 z_2 + z_2 z_3 - c_2 z_2^2 - \frac{\dot{g}_2}{2g_2^2} z_2^2 + z_2 \varepsilon_2$$

$$- \tilde{W}_2^T S_2(Z_2) z_2 + \tilde{W}_2^T \Gamma_2^{-1} \dot{\hat{W}}_2 - p_{21}\left(\frac{\partial \alpha_1}{\partial x_1}\varphi_1\right)^2 z_2^2$$

$$- p_2 \varphi_2^2 z_2^2 - \frac{\partial \alpha_1}{\partial x_1} \frac{\phi_1}{g_2} d_1 + \frac{\phi_2}{g_2} d_2 \tag{23}$$

Consider the following adaptation law

$$\dot{\hat{W}}_2 = \dot{\tilde{W}}_2 = \Gamma_2[S_2(Z_2)z_2 - \sigma_2\hat{W}_2] \quad (24)$$

where $\sigma_2 > 0$ is a small constant. Let $c_2 = c_{20} + c_{21}$, where c_{20} and $c_{21} > 0$. By using (15), (21) and (24), and with some completion of squares and straightforward derivation similar to those employed in Step 1, the derivative of V_2 becomes

$$\dot{V}_2 < z_2 z_3 - \sum_{k=1}^{2} c_{k0}^* z_k^2 - \sum_{k=1}^{2} \frac{\sigma_k\|\tilde{W}_k\|^2}{2} + \sum_{k=1}^{2} \frac{\sigma_k\|W_k^*\|^2}{2} + \sum_{k=1}^{2} \frac{\varepsilon_k^{*2}}{4c_{k1}} + \sum_{k=1}^{2} \frac{d_k^{*2}}{4p_k g_{k0}^2} + \frac{d_1^{*2}}{4p_{21}g_{20}^2} \quad (25)$$

where c_{20} is chosen such that $c_{20}^* \triangleq c_{20} - (g_{2d}/2g_{20}^2) > 0$.

Step i $(3 \leq i \leq n)$. The design procedures in Step i $(i = 3, \ldots, n)$ are very similar to the design in Step 2. Thus, we simplify the description in Step i for conciseness of presentation.

The derivative of $z_i = x_i - \alpha_{i-1}$ is $\dot{z}_i = f_i(\overline{x}_i) + g_i(\overline{x}_i)x_{i+1} + \phi_i(\overline{x}_i)d_i - \dot{\alpha}_{i-1}$ $(x_{n+1} \triangleq u)$, where

$$\dot{\alpha}_{i-1} = \sum_{k=1}^{i-1} \frac{\partial\alpha_{i-1}}{\partial x_k}(g_k x_{k+1} + f_k + \phi_i d_i) + \psi_{i-1} \quad (26)$$

with $\psi_{i-1} = \sum_{k=1}^{i-1}(\partial\alpha_{i-1}/\partial x_d)\dot{x}_d + \sum_{k=1}^{i-1}(\partial\alpha_{i-1}/\partial\hat{W}_k)[\Gamma_k(S_k(Z_k)z_k - \sigma_k\hat{W}_k)]$.

By viewing x_{i+1} as a virtual control input to stabilize the $(z_1, \ldots, z_i)$-subsystem, there exists a desired feedback control $\alpha_i^* = x_{i+1}$

$$\begin{aligned}\alpha_i^* &= -z_{i-1} - c_i z_i - \frac{1}{g_i(\overline{x}_i)}(f_i(\overline{x}_i) - \dot{\alpha}_{i-1}) - \frac{\phi_i}{g_i}d_i \\ &= -z_{i-1} - c_i z_i - \frac{1}{g_i}\left[f_i - \sum_{k=1}^{i-1}\frac{\partial\alpha_{i-1}}{\partial x_k}(g_k x_{k+1} + f_k) - \psi_{i-1}\right] + \sum_{k=1}^{i-1}\frac{\partial\alpha_{i-1}}{\partial x_k}\frac{\phi_k}{g_i}d_k - \frac{\phi_i}{g_i}d_i \end{aligned} \quad (27)$$

By employing an RBF neural network $W_i^T S_i(Z_i)$ to approximate the unknown function $h_i(Z_i)$ in Eq. (27), i.e.

$$\begin{aligned} h_i(Z_i) &\triangleq \frac{1}{g_i}\left[f_i - \sum_{k=1}^{i-1}\frac{\partial\alpha_{i-1}}{\partial x_k}(g_k x_{k+1} + f_k) - \psi_{i-1}\right] \\ &= W_i^{*T}S_i(Z_i) + \varepsilon_i \end{aligned} \quad (28)$$

with

$$Z_i \triangleq \left[\overline{x}_i^T, \frac{\partial\alpha_{i-1}}{\partial x_1}, \cdots, \frac{\partial\alpha_{i-1}}{\partial x_{i-1}}, \psi_{i-1}\right]^T \subset R^{2i} \quad (29)$$

and introducing the nonlinear damping term to cope with the disturbance term, the practical virtual control and the practical control u are chosen as

$$\alpha_i = -z_{i-1} - c_i z_i - \hat{W}_i^T S_i(Z_i) - \sum_{k=1}^{i-1} p_{ik}\left(\frac{\partial\alpha_{i-1}}{\partial x_k}\varphi_k\right)^2 z_i - p_i\varphi_i^2 z_i\,, \quad i = 3, \ldots, n-1 \quad (30)$$

and

$$\begin{aligned} u &= \alpha_n \\ &= -z_{n-1} - c_n z_n - \hat{W}_n^T S_n(Z_n) - \sum_{k=1}^{n-1} p_{nk}\left(\frac{\partial\alpha_{n-1}}{\partial x_k}\varphi_k\right)^2 z_n - p_n\varphi_n^2 z_n\,. \end{aligned} \quad (31)$$

Then, we have

$$\dot{z}_i = g_i\left[z_{i+1} - z_{i-1} - c_i z_i - \tilde{W}_i^T S_i(Z_i) + \varepsilon_i - \sum_{k=1}^{i-1} p_{ik}\left(\frac{\partial\alpha_{i-1}}{\partial x_k}\varphi_k\right)^2 z_i - p_i\varphi_i^2 z_i - \sum_{k=1}^{i-1}\frac{\partial\alpha_{i-1}}{\partial x_k}\frac{\phi_k}{g_i}d_k + \frac{\phi_i}{g_i}d_i\right], \quad i = 3, \ldots, n-1 \quad (32)$$

and

$$\dot{z}_i = g_i\Big[- z_{i-1} - c_i z_i - \tilde{W}_i^T S_i(Z_i) + \varepsilon_i - \sum_{k=1}^{i-1} p_{ik}\Big(\frac{\partial \alpha_{i-1}}{\partial x_k}\varphi_k\Big)^2 z_k - p_i\varphi_i^2 z_i - \sum_{k=1}^{i-1}\frac{\partial \alpha_{i-1}}{\partial x_k}\frac{\phi_k}{g_i} d_k + \frac{\phi_i}{g_i} d_i\Big], \quad i = n\,. \tag{33}$$

Consider the Lyapunov function candidate

$$V_i = V_{i-1} + \frac{1}{2g_i(\overline{x}_i)} z_i^2 + \frac{1}{2}\tilde{W}_i^T \Gamma_i^{-1}\tilde{W}_i\,. \tag{34}$$

Consider the following adaptation law

$$\dot{\hat{W}}_i = \dot{\tilde{W}}_i = \Gamma_i[S_i(Z_i)z_i - \sigma_i\hat{W}_i] \tag{35}$$

where $\sigma_i > 0$ is a small constant. Let $c_i = c_{i0} + c_{i1}$, where c_{i0} and $c_{i1} > 0$. Using (25), (32) and (35), and with some completion of squares and straightforward derivation similar to those employed in the former steps, the derivative of V_i becomes

$$\dot{V}_i < z_i z_{i+1} - \sum_{k=1}^{i} c_{k0}^* z_k^2 - \sum_{k=1}^{i}\frac{\sigma_k\|\tilde{W}_k\|^2}{2} + \sum_{k=1}^{i}\frac{\sigma_k\|W_k^*\|^2}{2} + \sum_{k=1}^{i}\frac{\varepsilon_k^{*2}}{4c_{k1}} + \sum_{k=1}^{i}\frac{{d_k}^2}{4p_k g_{k0}^2} + \sum_{l=1}^{i}\sum_{k=1}^{l-1}\frac{{d_k}^2}{4p_{lk}g_{l0}^2}\,, \quad i = 3,\ldots,n-1 \tag{36}$$

and

$$\dot{V}_i < - \sum_{k=1}^{i} c_{k0}^* z_k^2 - \sum_{k=1}^{i}\frac{\sigma_k\|\tilde{W}_k\|^2}{2} + \sum_{k=1}^{i}\frac{\sigma_k\|W_k^*\|^2}{2} + \sum_{k=1}^{i}\frac{\varepsilon_k^{*2}}{4c_{k1}} + \sum_{k=1}^{i}\frac{{d_k}^2}{4p_k g_{k0}^2} + \sum_{l=1}^{i}\sum_{k=1}^{l-1}\frac{{d_k}^2}{4p_{lk}g_{l0}^2}\,, \quad i = n \tag{37}$$

where c_{i0} is chosen such that $c_{i0}^* \overset{\triangle}{=} c_{i0} - (g_{id}/2g_{i0}^2) > 0$.

The following theorem shows the stability and control performance of the closed-loop adaptive system.

Theorem 1. *Consider the closed-loop system consisting of the plant (1), the reference model (2), the controller (31) and the NN weight updating laws (13), (24) and (35). Assume there exists sufficiently large compact sets* $\Omega_i \in R^{2i}$, $i = 1,\ldots,n$ *such that* $Z_i \in \Omega_i$ *for all* $t \geq 0$. *Then, for bounded initial conditions, all signals in the closed-loop system remain bounded, and the output tracking error* $y(t) - y_d(t)$ *converges to a neighborhood around zero by appropriately choosing design parameters.*

Proof. Let $\delta \overset{\triangle}{=} \sum_{k=1}^{n}(\sigma_k\|W_k^*\|^2/2) + \sum_{k=1}^{n}(\varepsilon_k^{*2}/4c_{k1}) + \sum_{k=1}^{n}(d_k^{*2}/4p_k g_{k0}^2) + \sum_{l=1}^{n}\sum_{k=1}^{l-1}(d_k^{*2}/4p_{lk}g_{l0}^2)$. If we choose c_{k0}^* such that $c_{k0}^* \geq (\gamma/2g_{k0})$, i.e. choose c_{k0} such that $c_{k0} > (\gamma/2g_{k0}) + (g_{kd}/2g_{k0}^2)$, $k = 1,\ldots,n$, where γ is a positive constant, and choose σ_k and Γ_k such that $\sigma_k \geq \gamma\lambda_{\max}\{\Gamma_k^{-1}\}$, $k = 1,\ldots,n$, then from (37) we have the following inequality

$$\dot{V}_n < - \sum_{k=1}^{n} c_{k0}^* z_k^2 - \sum_{k=1}^{n}\frac{\sigma_k\|\tilde{W}_k\|^2}{2} + \delta \leq - \sum_{k=1}^{n}\frac{\gamma}{2g_{k0}} z_k^2 - \sum_{k=1}^{n}\frac{\gamma\tilde{W}_k^T\Gamma_k^{-1}\tilde{W}_k}{2} + \delta \leq -\gamma\Big[\sum_{k=1}^{n}\frac{1}{2g_k} z_k^2 + \sum_{k=1}^{n}\frac{\tilde{W}_k^T\Gamma_k^{-1}\tilde{W}_k}{2}\Big] + \delta \leq -\gamma V_n + \delta \tag{38}$$

Thus, all z_i and $\hat{W}_i$ $(i = 1, \ldots, n)$ are uniformly ultimately bounded. Since $z_1 = x_1 - x_{d1}$ and x_{d1} are bounded, we have that x_1 is bounded. From $z_i = x_i - \alpha_{i-1}$, $i = 2, \ldots, n$, and the definitions of virtual controls α_i in (9), (20) and (30), we have that x_i, $i = 2, \ldots, n$ remain bounded. Using (31), we conclude that control u is also bounded. Thus, all the signals in the closed-loop system remain bounded.

Let $\rho \stackrel{\triangle}{=} \delta/\gamma > 0$, then (38) satisfies

$$0 \le V_n(t) < \rho + (V_n(0) - \rho)\exp(-\gamma t) \tag{39}$$

From (39), we have

$$\begin{aligned}\sum_{k=1}^{n} \frac{1}{2g_k} z_k^2 &< \rho + (V_n(0) - \rho)\exp(-\gamma t) \\ &< \rho + V_n(0)\exp(-\gamma t)\end{aligned} \tag{40}$$

Let $g^* = \max_{1\le i\le n}\{g_{i1}\}$. Then, we have

$$\frac{1}{2g^*}\sum_{k=1}^{n} z_k^2 \le \sum_{k=1}^{n} \frac{1}{2g_k} z_k^2 < \rho + V_n(0)\exp(-\gamma t) \tag{41}$$

that is,

$$\sum_{k=1}^{n} z_k^2 < 2g^*\rho + 2g^* V_n(0)\exp(-\gamma t) \tag{42}$$

which implies that given $\mu > \sqrt{2g^*\rho}$, there exists T such that for all $t \ge T$, the tracking error satisfies

$$\begin{aligned}|z_1(t)| &= |x_1(t) - x_{d1}(t)| \\ &= |y(t) - y_d(t)| < \mu\end{aligned} \tag{43}$$

where μ is the size of a residual set which depends on the NN approximation error ε_i, the disturbance inputs d_i, and controller parameters c_i, p_i, σ_i and Γ_i. ■

Remark 1. Accordingly, we have the following statements to make:

(i) increasing c_{i0} might lead to larger γ, and increasing c_{i1} and p_i will reduce δ, i.e. increasing c_i and p_i will lead to smaller μ; and

(ii) decreasing σ_i will help to reduce δ, and increasing the NN node number l_j will help to reduce ε_i^*, both of which will help to reduce the size of μ. However, increasing c_i and p_i will lead to a high gain control scheme, and a very small σ_i may not be enough to prevent the NN weight estimates from drifting to very large values in the presence of the NN approximation errors, where the large $\hat{W}_i$ might result in a variation of a high-gain control. Therefore, in practical applications, the design parameters should be adjusted carefully for achieving suitable transient performance and control action.

Remark 2. Note that for the control of uncertain strict-feedback system (1), the differentiation requirement becomes a technical problem due to the presence of two types of uncertainties. The derivative of virtual control α_{i-1} (26) involves the unknown nonlinear functions $f_1, \ldots, f_{i-1}(\cdot)$ and $g_1, \ldots, g_{i-1}(\cdot)$, as well as the disturbances $\phi_1 d_1, \ldots, \phi_{i-1}d_{i-1}$, and thus is unknown and unavailable for implementation. Since the derivative $\dot{\alpha}_{i-1}$ appears in the desired virtual control α_i^* (27), to cope with the two types of uncertainties in the desired virtual control, two techniques are employed. The RBF neural networks are used to approximate all the unknown nonlinearities $h_i(Z_i)$ in α_i^* (27). Here we do not take the neural weight estimates $\hat{W}_1, \ldots, \hat{W}_{i-1}$ as inputs to the RBF NNs, which will make the approximation computationally unacceptable and lead to curse of dimensionality [Haykin, 1999] with large number of neural weight estimates. Instead, we introduce intermediate variables $(\partial\alpha_{i-1}/\partial x_1), \ldots, (\partial\alpha_{i-1}/\partial x_{i-1})$ and ϕ_{i-1} as inputs to RBF NN $W_i^T S_i(Z_i)$, where the intermediate variables are available through the computation of system states $\overline{x}_i$ and neural weight estimates $\hat{W}_1, \ldots, \hat{W}_{i-1}$. Thus, the NN approximation can be implemented by using the minimal number of NN input variables.

To deal with the disturbance terms $\phi_i d_i$, there is no need to use norm operation which is commonly employed in the robust control design. The norm operation is in general not differentiable with respect to its arguments. We use nonlinear damping terms instead to counteract the disturbances. Hence, the differentiability problem in the control of disturbed strict-feedback system (1) is solved by combining backstepping methodology with adaptive neural design and nonlinear damping techniques.

4. Simulation Studies

The Brusselator model is a simplified model describing a certain set of chemical reactions. This model was introduced by Turing [1952] and studied in detail by Prigogine and his coworkers [Nicolis & Prigogine, 1977]. This model was named Brusselator because its originators worked in Brussels. It has become one of the most popular nonlinear oscillatory models of chemical kinetics, as well as one of the paradigms in the research of chaos.

The Brusselator model in dimensionless form is

$$\dot{x}_1 = A - (B+1)x_1 + x_1^2 x_2$$
$$\dot{x}_2 = Bx_1 - x_1^2 x_2$$

where x_1 and x_2 are the concentrations of the reaction intermediates; $A,\ B > 0$ are parameters describing the (constant) supply of "reservoir" chemicals.

As a simplified model depicting chemical reactions, the Brusselator model (44) is derived from partial differential equations (PDE) after a series of approximations [Nicolis & Prigogine, 1977]. Thus, there must exist modeling errors and other types of unknown nonlinearities in the practical chemical reactions. The controlled Brusselator with disturbance is assumed as

$$\begin{aligned} \dot{x}_1 &= A - (B+1)x_1 + x_1^2 x_2 + 0.7x_1^2 \cos(1.5t) \\ \dot{x}_2 &= Bx_1 - x_1^2 x_2 + (2 + \cos(x_1))u \\ y &= x_1 \end{aligned} \tag{44}$$

where $0.7x_1^2 \cos(1.5t)$ is the disturbance term with $d_1 = \cos(1.5t)$ and $\phi_1 = x_1^2$, the nonlinearities $f_1(x_1) = A - (B+1)x_1$, $g_1(x_1) = x_1^2$, $f_2(\overline{x}_2) = Bx_1 - x_1^2 x_2$, $g_2(\overline{x}_2) = 2 + \cos(x_1)$ are assumed unknown to the controller u. For clarity of presentation, we assume that $d_2 = 0$ and $\varphi_1 = \phi_1 = x_1^2$. It can be easily seen that the Brusselator is in strict-feedback form (1), under the assumption that $x_1 \neq 0$ (which is reasonable if x_1 does not reach zero in practice).

The control objective is to guarantee (i) all the signals in the closed-loop system remain bounded, and (ii) the output y follows the reference signal $y_d = 3 + \sin(0.5t)$. The robust adaptive NN controller is chosen according to (31) as follows

$$u = -z_1 - c_2 z_2 - \hat{W}_2^T S_2(Z_2) - p_{21} \frac{\partial \alpha_1}{\partial x_1} \varphi_1^2 z_2 \tag{45}$$

where $z_1 = x_1 - y_d$, $z_2 = x_2 - \alpha_1$ and $Z_2 = [x_1,\ x_2,\ \partial\alpha_1/\partial x_1,\ \psi_1]^T$ with

$$\begin{aligned} \alpha_1 &= -c_1 z_1 - \hat{W}_1^T S_1(Z_1) - p_1 \varphi_1^2 z_1\,, \\ Z_1 &= [x_1,\ \dot{y}_d]^T \\ \psi_1 &= \frac{\partial \alpha_1}{\partial y_d}\dot{y}_d + \frac{\partial \alpha_1}{\partial \dot{y}_d}\ddot{y}_d + \frac{\partial \alpha_1}{\partial \hat{W}_1}\dot{\hat{W}}_1 \end{aligned} \tag{46}$$

and NN weights $\hat{W}_1$ and $\hat{W}_2$ are updated by (13) and (24) correspondingly.

The selection of the centers and widths of RBF has a great influence on the performance of the adaptive neural controller. It has been indicated [Sanner & Slotine, 1992] that Gaussian RBF NNs arranged on a regular lattice on R^n can uniformly approximate sufficiently smooth functions on closed, bounded subsets.

Accordingly, in the following simulation studies, we select the centers and widths as: Neural networks $\hat{W}_1^T S_1(Z_1)$ contains 25 nodes (i.e. $l_1 = 25$), with centers μ_l $(l = 1, \ldots, l_1)$ evenly spaced in $[0,\ 5] \times [-2,\ -2]$, and widths $\eta_l = 1.5$ $(l = 1, \ldots, l_1)$. Neural networks $\hat{W}_2^T S_2(Z_2)$ contains 144 nodes (i.e. $l_2 = 144$), with centers μ_l $(l = 1, \ldots, l_2)$ evenly spaced in $[0,\ 5] \times [-4,\ 4] \times [-4,\ 4] \times [-4,\ 4]$, and widths $\eta_l = 2$ $(l = 1, \ldots, l_2)$. The design parameters of the above controller are $c_1 = 1.5$, $c_2 = 2$, $p_1 = 0.01$, $p_{21} = 0.01$, $\Gamma_1 = \Gamma_2 = \mathrm{diag}\{2.0\}$, $\sigma_1 = \sigma_2 = 0.2$. The initial weights $\hat{W}_1(0) = 0.0$, $\hat{W}_2(0) = 0.0$, the initial conditions $[x_1(0),\ x_2(0)]^T = [2.7,\ 1]^T$.

Figures 1–4 show the simulation results of applying controller (45) to the uncertain Brusselator model (44) for tracking desired signal y_d. From Figs. 1 and 2, we can see that fairly good tracking performance is obtained (the solid line). The boundedness of NN weights $\hat{W}_1$, $\hat{W}_2$ and control signal u are shown in Figs. 3 and 4 respectively.

In comparison, two more simulation studies are conducted to further verify the effectiveness of the robust adaptive NN controller. Firstly, we remove the nonlinear damping terms in the control (45) and the virtual control α_1 (46). In this case, we obtain an adaptive NN controller. It can be seen from Figs. 1 and 2 (the dashed line) that the output tracking performance becomes worse compared with the solid lines in Figs. 1 and 2. Secondly, we turn off the adaptation of neural networks, i.e. let $\hat{W}_1(t) = 0$, $\hat{W}_1(t) = 0$, $\forall t \geq 0$. The controller (45) becomes a PD-like controller. It can be seen from the dashdotted lines in Figs. 1 and 2 that the output

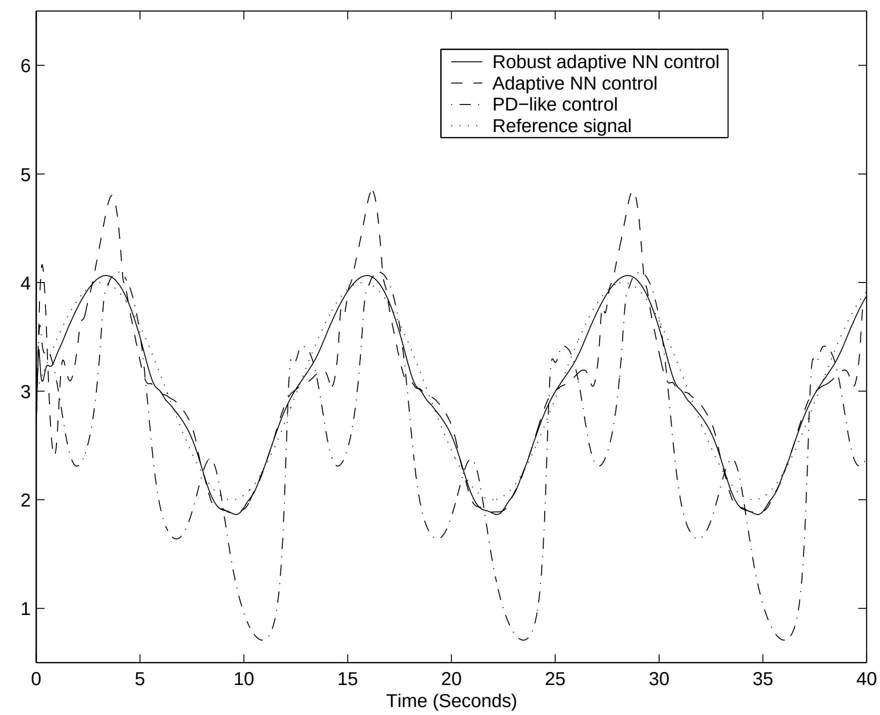

Fig. 1. Output tracking performance.

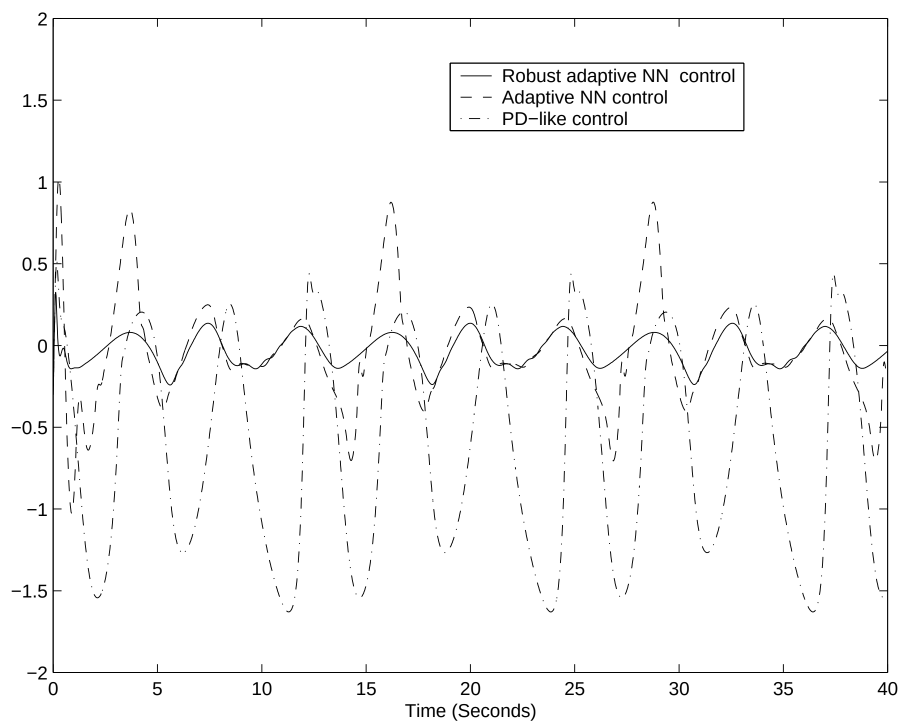

Fig. 2. Tracking errors $y - y_d$.

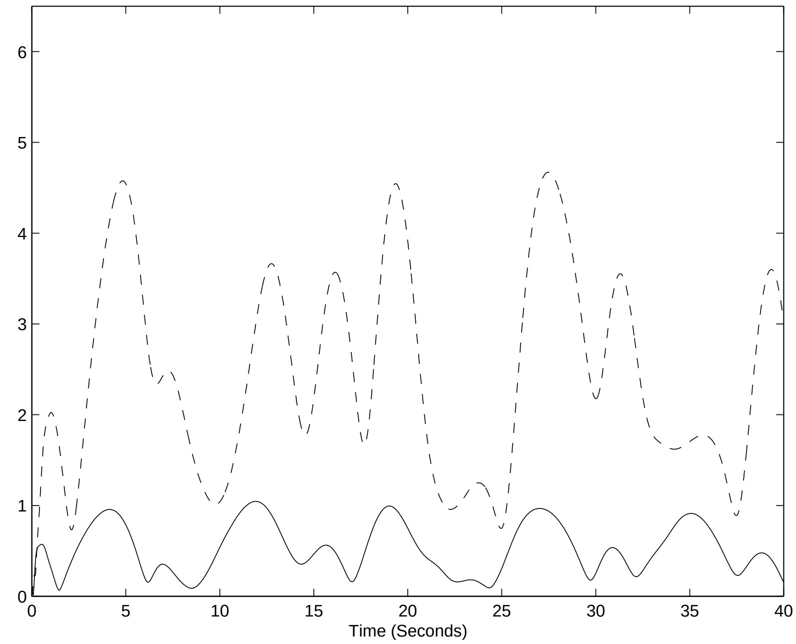

Fig. 3. L_2 norms of the NN weights: $\hat{W}_1$ (solid line) and $\hat{W}_2$ (dashed line).

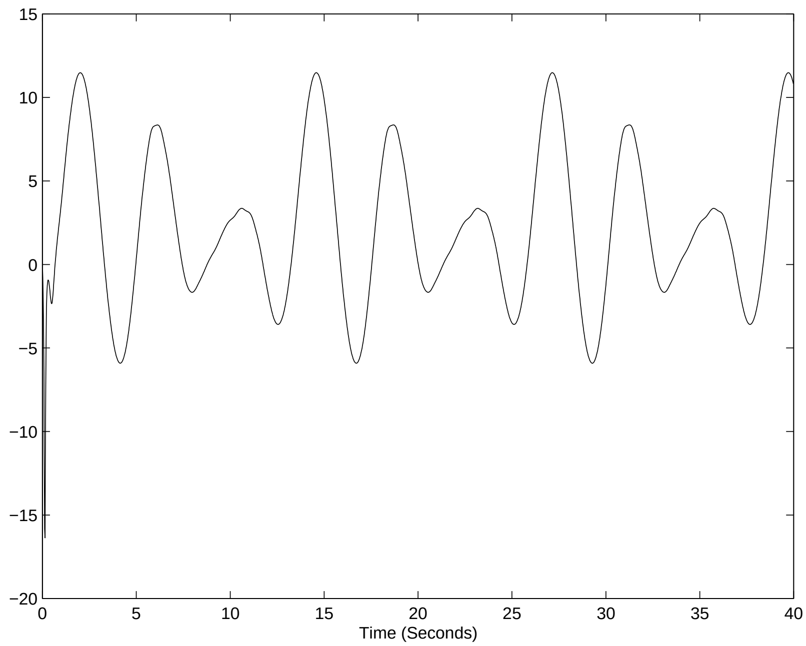

Fig. 4. Boundedness of the control u.

tracking performance becomes much worse without NN adaptation.

5. Conclusion

In this paper, a robust adaptive neural control scheme is presented for a class of uncertain chaotic systems in the disturbed strict-feedback form, with both unknown nonlinearities and uncertain disturbances. To cope with the two types of uncertainties, backstepping methodology is combined with adaptive neural design and nonlinear damping technique. A smooth adaptive neural controller is presented, where nonlinear damping terms are used to counteract the disturbances. Simulation results demonstrate that the proposed controllers can be applied to uncertain chaotic systems in practical situations.

References

Arslan, G. & Basar, T. [2001] "Disturbance attenuating controller design for strict-feedback systems with structurally unknown dynamics," *Automatica* **37**, 1175–1188.

Chen, F. C. & Liu, C. C. [1994] "Adaptively controlling nonlinear continuous-time systems using multilayer neural networks," *IEEE Trans. Automat. Cont.* **39**(6), 1306–1310.

Chen, G. (ed.) [1999] *Controlling Chaos and Bifurcations in Engineering Systems* (CRC Press, Boca Raton).

Chua, L. O., Komuro, M. & Matsumoto, T. [1986] "The double scroll family parts I and II," *IEEE Trans. Circuits Syst. Part I* **33**(11), 1073–1118.

Duffing, G. [1918] *Erzwungene Schwingungen Bei Veränderlicher Eigenfrequenz und ihre Technische Bedeutung* Vieweg, Braunschweig.

Fitzhugh, R. [1961] "Impulses and physiological states in theoretical models of nerve membrane," *Biophys. J.* **1**, 445–466.

Fradkov, A. L. & Pogromsky, A. Yu. [1996] "Speed gradient control of chaotic systems," *IEEE Trans. Circuits Syst. Part I* **43**, 907–913.

Freeman, R. A., Krstic, M. & Kokotovic, P. V. [1998] "Robustness of adaptive nonlinear control to bounded uncertainties," *Automatica* **34**, 1227–1230.

Ge, S. S., Lee, T. H. & Harris, C. J. [1998] *Adaptive Neural Network Control of Robotic Manipulators* (World Scientific, Singapore).

Ge, S. S., Wang, C. & Lee, T. H. [2000] "Adaptive backstepping control of a class of chaotic systems," *Int. J. Bifurcation and Chaos* **10**(5), 1149–1156.

Ge, S. S., Hang, C. C., Lee, T. H. & Zhang, T. [2001] *Stable Adaptive Neural Network Control* (Kluwer Academic Publishers, Norwell, USA).

Ge, S. S. & Wang, C. [2002] "Direct adaptive NN control of a class of nonlinear systems," *IEEE Trans. Neural Networks* **13**(1), 214–221.

Gong, J. & Yao, B. [2001] "Neural network adaptive robust control of nonlinear systems in semi-strict feedback form," *Automatica* **37**, 1149–1160.

Haykin, S. [1999] *Neural Networks*: *A Comprehensive Foundation*, 2nd edition (Prentice-Hall, NJ).

Jiang, Z. P. & Praly, L. [1998] "Design of robust adaptive controllers for nonlinear systems with dynamic uncertainties," *Automatica* **34**, 825–840.

Kanellakopoulos, I., Kokotović, P. & Morse, A. [1991] "Systematic design of adaptive controllers for feedback linearizable systems," *IEEE Trans. Automat. Contr.* **36**, 1241–1253.

Krstić, M., Kanellakopoulos, I. & Kokotović, P. [1992] "Adaptive nonlinear control without overparametrization," *Syst. Contr. Lett.* **19**, 177–185.

Krstić, M., Kanellakopoulos, I. & Kokotović, P. [1995] *Nonlinear and Adaptive Control Design* (John Wiley, NY).

Lewis, F. L., Jagannathan, S. & Yeildirek, A. [1999] *Neural Network Control of Robot Manipulators and Nonlinear Systems* (Taylor & Francis, London).

Lewis, F. L., Yeildirek, A. & Liu, K. [2000] "Robust backstepping control of induction motors using neural networks," *IEEE Trans. Neural Networks* **11**(5), 1178–1187.

Lorenz, E. N. [1963] "Deterministic non-periodic flow," *J. Atmos. Sci.* **20**, 130–141.

Nicolis, G. & Prigogine, I. [1977] *Self-organization in Non-equilibrium Systems* (Wiley, NY).

Pan, Z. & Basar, T. [1998] "Adaptive controller design for tracking and disturbance attenuation in parametric strict-feedback nonlinear systems," *IEEE Trans. Automat. Contr.* **43**(8), 1066–1083.

Polycarpou, M. M. & Ioannou, P. A. [1992] "Modeling, identification and stable adaptive control of continuous-time nonlinear dynamical systems using neural networks," *Proc. American Control Conf.*, Boston, MA, pp. 36–40.

Polycarpou, M. M. & Ioannou, P. A. [1996] "Robust adaptive nonlinear control design," *Automatica* **32**, 423–427.

Polycarpou, M. M. & Mears, M. J. [1998] "Stable adaptive tracking of uncertain systems using nonlinearly parametrized on-line approximators," *Int. J. Contr.* **70**(3), 363–384.

Qin, H., Zhang, H. & Chen, G. [1999] "Neural-network based adaptive control of uncertain chaotic systems," *14th World Congress of IFAC*, Beijing, China, pp. 318–321.

Rössler, O. E. [1976] "An equation for continuous chaos," *Phys. Letts.* **A57**, 397–398.

Rovithakis, G. A. & Christodoulou, M. A. [1994] "Adaptive control of unknown plants using dynamical neural networks," *IEEE Trans. Syst. Man Cybern.* **24**, 400–412.

Sanner, R. M. & Slotine, J. E. [1992] "Gaussian networks for direct adaptive control," *IEEE Trans. Neural Networks* **3**(6), 837–863.

Spooner, J. T. & Passino, K. M. [1996] "Stable adaptive control using fuzzy systems and neural networks," *IEEE Trans. Fuzzy Syst.* **4**(3), 339–359.

Turing, A. M. [1952] "The chemical basis of morphogenesis," *Philos. Trans. R. Soc.* **B237**(641), 37–72.

van der Pol, B. [1927] "Forced oscillations in a circuit with nonlinear resistance (reception with reactive triode)," *Philos. Mag.* **7**, 65–80.

Yesidirek, A. & Lewis, F. L. [1995] "Feedback linearization using neural networks," *Automatica* **31**, 1659–1664.