Adaptive Neural Network Control for Strict-Feedback Nonlinear Systems With Unknown Control Direction

Y. Q. Jin, S. X. Wang, and Z. C. Xiao

Naval Aeronautical Engineering Institute, Yantai
China 264001

jyq301@yahoo.com.cn

Abstract

Adaptive neural network control was presented for a class of strict-feedback nonlinear systems with unknown the sign of control coefficients. A systematic procedure, which relaxes some rigorous restrictions on the plants in the literature at present stage, was developed based on Nussbaum-type function and RBF neural networks. The developed control scheme guarantees global stability of the closed-loop systems. Finally, numerical simulation study was presented to demonstrate the effectiveness of the proposed method.

Keyword: nonlinear system; neural networks; adaptive control

I. Introduction

Recently, interest in adaptive control of nonlinear system has been ever increasing, and many significant developments have achieved. In order to guarantee the global stability, some restrictions on the plants had to be made such as matching condition, or growth conditions on system nonlinear. To overcome these restrictions, a recursive design procedure called adaptive backstepping design was developed in [1]. Adaptive backstepping control has been studied for certain class of strict-feedback nonlinear systems [2]-[5]. Several adaptive control systems have been proposed for parametric strict-feedback systems with unknown control coefficients but with known signs (either positive or negative) in [3], [6].

When there is no a prior knowledge about the signs of control coefficients, adaptive control of such systems becomes much more difficult. Up to now, there are mainly two ways to address the problem. One way is to incorporate the technique of Nussbaum-type gain into the control design [5], [7]. Another way is to directly estimate unknown parameters involved in the control directions [8]-[9]. Most of these results only can be applied to first-order systems or second-order systems.

Motivated by previous works on unknown control direction system, we successfully incorporate the technique of Nussbaum-type gain into backstepping design and propose an adaptive neural network control scheme. The proposed method avoids the controller singularity problem and overparameterization problem encountered in many results. The main contributions of this paper lie in:

i) backstepping design techniques and the Nussbaum-type functions are incorporated to solve the problem of the completely unknown control direction;
ii) no growth restriction are imposed on system nonlinearities;
iii) the resulting adaptive control is smooth. Computer explosion problem, and overparameterization problem are reduced.
iv)

The paper is organized as follows. Formulation and preliminaries of our adaptive control problem of uncertain nonlinear system is given in Section II. An adaptive neural control design procedure is presented in Section III. Simulation study and conclusion are presented in Section IV and V, respectively.

II. Problem Formulation and Preliminaries

Consider the following uncertain nonlinear system

$$\begin{aligned} \dot{x}_1 &= f_1(x_1) + g_1(x_1)x_2 \\ \dot{x}_i &= f_i(\bar{x}_i) + g_i(\bar{x}_i)x_{i+1},\ 2 \le i \le n-1, \\ \dot{x}_n &= f_n(\bar{x}_n) + g_n(\bar{x}_n)u \end{aligned} \tag{1}$$

where $\bar{x}_i = [x_1 \ \cdots \ x_i]^T$, $2 \le i \le n$, $u \in R$ are the state variable and system input respectively, $f_i(\bar{x}_i), g_i(\bar{x}_i)$ are uncertain smooth functions.

Assumption 1. $f_i(0) = 0$, $1 \le i \le n$.

Assumption 2. Function $g_i(\bar{x}_i)$ and their signs are unknown, and there exist constants g_{i0} and known smooth functions $\bar{g}_i(\bar{x}_i)$ such that $0 < g_{i0} \le |g_i(\bar{x}_i)| \le \bar{g}_i(\bar{x}_i)$, $\forall \bar{x}_i \in R^i$.

The RBF neural network as a kind of linear parametrized neural networks is found wide applications in control system design because of its nice approximation properties.

Assumption 3. Function vector $\Delta : \Omega \mapsto R$, Ω belongs to a sub-compact set of R. $\forall\ \varepsilon > 0$, there always exist a Gaussian base function vector $\varphi : R \mapsto R^l$ and an optimal weight matrix $W^* \in R^{l\times 1}$ such that

$$\Delta = W^{*T}\varphi + \varepsilon, \forall x \in \Omega, \tag{2}$$

where $\varphi = \left[\exp\left(\frac{-|\varsigma - \mu_1|^2}{\sigma_1^2} \right) \ \cdots \ \exp\left(\frac{-|\varsigma - \mu_l|^2}{\sigma_l^2} \right) \right]^T$, $\mu_i, i = 1,2,...,l$ is the center, l is the number of hidden layer nodes, $\sigma_i, i = 1,2,...,l$ is the affect size, ς is the input of RBF NN, and ε is the construction error of NN.

Nussbaum-type gains have been effectively used in controller design in solving the difficulty of unknown control directions.

Any continuous function $v(k): R \to R$ is a Nussbaum-type function if it has the following properties

$$\begin{aligned} &\lim_{s\to\infty} \sup \int^s v(k)dk = \infty \\ &\lim_{s\to\infty} \inf \int^s v(k)dk = -\infty \end{aligned}. \tag{3}$$

For example, the continuous functions $k^2 \cos(k)$, and $k^2 \sin(k)$. Through this paper, we choose $v(k) = k^2 \cos(k)$.

Lemma 1. [5] Let $V(\cdot)$ and $k(\cdot)$ be smooth functions defined on $\left[0, t_f\right)$ with $V(t) \geq 0$, $\forall t \in \left[0, t_f\right)$, $v(\cdot)$ be an even smooth Nussbaum-type function, and θ_0 be a nonzero constant. If the following inequality holds

$$V(t) \leq e^{-c_1 t} \int_0^t \left(\theta_0 v\left(k(\tau)\right) + 1\right) \dot{k}(\tau) e^{c_1 \tau} d\tau + c_0,\ \forall t \in \left[0, t_f\right), \tag{4}$$

where constant $c_1 > 0$, c is some suitable constant, then $\int_0^t \left(\theta_0 v\left(k(\tau)\right) + 1\right) \dot{k}(\tau) d\tau$, $V(t)$, and $k(t)$ must be bounded on $\left[0, t_f\right)$.

In this paper, an adaptive neural network controller is design for the regulation problem. The method can be extended to the tracking problem easily.

III. Adaptive Control Design

The design procedure consists of n step, at the i th step, $1 \leq i \leq n-1$, the state variable x_{i+1} is viewed as the fictitious control. Firstly, an ideal fictitious signal α_i is designed. Then, the actual control u is given at the n th step, which completes the design.

Step 1: Defining the new state variable $z_1 = x_1$, which satisfies

$$\dot{z}_1 = f_1(z_1) + g_1(z_1) x_2. \tag{5}$$

We choose the following ideal fictitious control signal α_1 to stabilize (5)

$$\begin{aligned} &\alpha_1 = v(k_1)\left[z_1 + \hat{W}_1^T \varphi_1(z_1)\right] \\ &\dot{k}_1 = z_1^2 + \hat{W}_1^T \varphi_1(z_1) z_1 \\ &\dot{\hat{W}}_1 = \Gamma_1 \varphi_1(z_1) z_1 - \Gamma_1 \sigma_1 \hat{W}_1 \end{aligned}, \tag{6}$$

where $v(\cdot)$ is an smooth Nussbaum-type function, small constant $\sigma_1 > 0$ is to introduce the σ-modification for the closed-loop system. Other symbols will be explained later.

Choose Lyapunov function as

$$V_1 = \frac{1}{2\left|g_1(z_1)\right|} z_1^2 + \frac{1}{2} tr\left(\tilde{W}_1^T \Gamma_1^{-1} \tilde{W}_1\right), \tag{7}$$

where $\Gamma_1 = \Gamma_1^T > 0$. Here, it should be emphasized that $\left|g_1(z_1)\right|$ is only required for analytical purposes, its true value is not necessarily known.

Viewing as $x_2 = \alpha_1$, the time derivative of V_1 along (5) is given by

$$\begin{aligned}\dot{V}_1 &= \frac{1}{\left|g_1(z_1)\right|} z_1\left(f_1(z_1) + g_1(z_1)x_2\right) + \tilde{W}_1^T \Gamma_1^{-1}\dot{\tilde{W}}_1 \\ &= z_1\left(\frac{f_1(z_1)}{\left|g_1(z_1)\right|} + \theta_1 x_2\right) + \tilde{W}_1^T \Gamma_1^{-1}\dot{\tilde{W}}_1\end{aligned}, \quad (8)$$

where $\theta_1 = 1$ or -1 is unknown constant. Apparently, $\frac{f_1(z_1)}{\left|g_1(z_1)\right|}$ is continuous and can be approximated by RBF neural networks to arbitrary any accuracy as

$$\Delta_1(z_1) \triangleq \frac{f_1(z_1)}{\left|g_1(z_1)\right|} = W_1^{*T}\varphi_1(z_1) + \varepsilon_1(z_1), \quad (9)$$

where $\left|\varepsilon_1(z_1)\right| < \varepsilon_1^*$ is the approximation error. $W_1^* \in R^{l\times 1}$ are unknown ideal constant weights. Here, we use its estimate $\hat{W}_1$ instead to form the control law as (6). The weight estimation error is $\tilde{W}_1 = \hat{W}_1 - W_1^*$.

Substituting (6) and (9) into (8), we have

$$\begin{aligned}\dot{V}_1 &= z_1\left(W_1^{*T}\varphi_1(z_1) + \varepsilon_1(z_1) + \theta_1 x_2\right) + \tilde{W}_1^T \Gamma_1^{-1}\dot{\tilde{W}}_1 \\ &= z_1\theta_1\alpha_1 + \left(W_1^{*T}\varphi_1(z_1) + \varepsilon_1(z_1)\right)z_1 + \tilde{W}_1^T \Gamma_1^{-1}\dot{\hat{W}}_1 \\ &= -z_1^2 + \left(\theta_1 v(k_1) + 1\right)\left(z_1^2 + \hat{W}_1^T\varphi_1(z_1)z_1\right) \\ &\quad -\tilde{W}_1^T\varphi_1(z_1)z_1 + z_1\varepsilon_1(z_1) + \tilde{W}_1^T \Gamma_1^{-1}\dot{\hat{W}}_1 \\ &= -z_1^2 + \left(\theta_1 v(k_1) + 1\right)\dot{k}_1 - \sigma_1\tilde{W}_1^T\hat{W}_1 + z_1\varepsilon_1(z_1)\end{aligned}. \quad (10)$$

Since $-\sigma_1\tilde{W}_1^T\hat{W}_1 \le \frac{\sigma_1}{2}\left\|W_1^*\right\|^2 - \frac{\sigma_1}{2}\left\|\tilde{W}_1\right\|^2$, $-\frac{1}{4}z_1^2 + \left|z_1\right|\varepsilon_1^* \le \varepsilon_1^{*2}$, the following inequality can be obtained

$$\begin{aligned}\dot{V}_1 &\le -\frac{3}{4}z_1^2 - \frac{\sigma_1}{2}\left\|\tilde{W}_1\right\|^2 + \left(\theta_1 v(k_1) + 1\right)\dot{k}_1 \\ &\quad + \frac{\sigma_1}{2}\left\|W_1^*\right\|^2 + \varepsilon_1^{*2} \\ &= -\frac{3}{4}z_1^2 - \frac{\sigma_1}{2}\left\|\tilde{W}_1\right\|^2 + \left(\theta_1 v(k_1) + 1\right)\dot{k}_1 + c_1 \\ &\le -\lambda_1 V_1 + + \left(\theta_1 v(k_1) + 1\right)\dot{k}_1 + c_1\end{aligned}, \quad (11)$$

where constant $\lambda_1 = \min\left\{3/4, \sigma_1 / \lambda_{\max}\left(\Gamma_1^{-1}\right)\right\}$, $c_1 = \frac{\sigma_1}{2}\left\|W_1^*\right\|^2 + \varepsilon_1^{*2}$. Multiplying (11) by $e^{\lambda_1 t}$, it becomes

$$\frac{d}{dt}\left(V_1 e^{\lambda_1 t}\right) \le c_1 e^{\lambda_1 t} + \left(\theta_1 v(k_1) + 1\right)\dot{k}_1 e^{\lambda_1 t}. \quad (12)$$

Integrating (12), we have

$$\begin{aligned} V_1 &\le \rho_1 + \left[V_1(0) - \rho_1\right] e^{-\lambda_1 t} \\ &\quad + e^{-\lambda_1 t}\int_0^t \left(\theta_1 v(k_1) + 1\right)\dot{k}_1 e^{\lambda_1 \tau} d\tau \\ &\le \rho_1 + V_1(0) e^{-\lambda_1 t} \\ &\quad + e^{-\lambda_1 t}\int_0^t \left(\theta_1 v(k_1) + 1\right)\dot{k}_1 e^{\lambda_1 \tau} d\tau \end{aligned}, \tag{13}$$

here $\rho_1 = c_1/\lambda_1$. Applying Lemma 1, we can conclude that V_1 and $k_1(t)$, hence $z_1(t)$ and $\hat{W}_1$ are all bounded on $\left[0, t_f\right)$. As an immediate result, $z_1(t)$ is square integrable and $\dot{z}_1(t)$ is bounded, both on $[0, \infty)$. Therefore, using Barbalat's lemma, regulation of $z_1(t)$ can be concluded, i,e., $\lim_{t\to\infty} z_1(t) = 0$. However, x_2 is not the actual control, hence there exists a difference between x_2 and α_1, which is defined as

$$z_2 = x_2 - \alpha_1\left(z_1, k_1, \hat{W}_1\right). \tag{14}$$

Accordingly, expression (11) should be modified as

$$\dot{V}_1 \le -\lambda_1 V_1 + \left(\theta_1 v(k_1) + 1\right)\dot{k}_1 + c_1 + \theta_1 z_1 z_2, \tag{15}$$

and the undesired effects of z_2 on $\dot{V}_1$ should be controlled at the next step using the conventional backstepping design procedure. However, the situation is somewhat different here. At the next step, there exist another Nussbaum-type gain in the expression of $\dot{V}_2$, it seem difficult to conclude any stability results [7]. So, some methods must be adopted to resolve this problem.

To proceed with our design procedure, we observe form (11) and (15) that

$$\begin{aligned} \dot{V}_1 &\le -\frac{1}{2} z_1^2 - \frac{\sigma_1}{2}\left\|\tilde{W}_1\right\|^2 + \left(\theta_1 v(k_1) + 1\right)\dot{k}_1 \\ &\quad + c_1 + \theta_1 z_2^2 - \left(\frac{1}{2} z_1 - \theta_1 z_2\right)^2 \\ &\le -\frac{1}{2} z_1^2 - \frac{\sigma_1}{2}\left\|\tilde{W}_1\right\|^2 + \left(\theta_1 v(k_1) + 1\right)\dot{k}_1 \\ &\quad + c_1 + \theta_1 z_2^2 \end{aligned}. \tag{16}$$

Thus, at step 2, if $z_2(t)$ can be regulated such that it is square integrable, then according to Lemma 1, regulation of $z_1(t)$ can also be achieved.

Step i $(2 \le i \le n-1)$: The time derivative of z_i is given by

$$\dot{z}_i = f_i(\bar{x}_i) + g_i(\bar{x}_i) x_{i+1} - \dot{\alpha}_{i-1}. \tag{17}$$

Since α_{i-1} is a function of $\bar{x}_{i-1}$, k_{i-1}, $\hat{W}_1$, $\cdots$, and $\hat{W}_{i-1}$, $\dot{\alpha}_{i-1}$ can be expressed as

$$\dot{\alpha}_{i-1} = \sum_{j=1}^{i-1} \frac{\partial \alpha_{i-1}}{\partial x_j}\left(f_j(\bar{x}_j) + g_j(\bar{x}_j) x_{j+1}\right) + \zeta_{i-1}, \tag{18}$$

where $\zeta_{i-1} = \frac{\partial \alpha_{i-1}}{\partial k_{i-1}}\dot{k}_{i-1} + \sum_{j=1}^{i-1}\frac{\partial \alpha_{i-1}}{\partial \hat{W}_j}\dot{\hat{W}}_j$ is computable. Then, (17) can be expressed as

$$\dot{z}_i = \bar{f}_i(\bar{x}_i, \zeta_{i-1}) + g_i(\bar{x}_i)x_{i+1}, \tag{19}$$

here, we assume that $\bar{f}_i(\bar{x}_i, \zeta_{i-1}) = f_i(\bar{x}_i) - \dot{\alpha}_{i-1}$ is unknown smooth function. We choose the following ideal fictitious control α_i

$$\begin{aligned} \alpha_i &= v(k_i)\left[z_i + \hat{W}_i^T \varphi_i(Z_i)\right] \\ \dot{k}_i &= z_i^2 + \hat{W}_i^T \varphi_i(Z_i) z_i \qquad , \\ \dot{\hat{W}}_i &= \Gamma_i \varphi_i(Z_i) z_i - \Gamma_i \sigma_i \hat{W}_i \end{aligned} \tag{20}$$

where $v(\cdot)$ is an smooth Nussbaum-type function, small constant $\sigma_i > 0$ is to introduce the σ-modification for the closed-loop system, $\Gamma_i = \Gamma_i^T > 0$ is constant matrix.

Considering the following Lyapunov function candidate

$$V_i = \frac{1}{2|g_i(\bar{x}_i)|} z_i^2 + \frac{1}{2} tr\left(\tilde{W}_i^T \Gamma_i^{-1} \tilde{W}_i\right). \tag{21}$$

Its time derivative along (19) and (20) is

$$\dot{V}_i = z_i \left(\frac{\bar{f}_i(\bar{x}_i, \zeta_{i-1})}{|g_i(\bar{x}_i)|} + \theta_i x_{i+1} \right) + \tilde{W}_i^T \Gamma_i^{-1} \dot{\hat{W}}_i, \tag{22}$$

where $\theta_i = 1$ or -1 is unknown constant. RBF neural network is used to approximate continuous function $\frac{\bar{f}_i(\bar{x}_i, \zeta_{i-1})}{|g_i(\bar{x}_i)|}$. Assuming that

$$\Delta_i(Z_i) \triangleq \frac{\bar{f}_i(\bar{x}_i, \zeta_{i-1})}{|g_i(\bar{x}_i)|} = W_i^{*T} \varphi_i(Z_i) + \varepsilon_i(Z_i), \tag{23}$$

where $Z_i = \left[\bar{x}_i^T, \zeta_{i-1}\right]^T$ is input vector of RBF neural network, $|\varepsilon_i(Z_i)| < \varepsilon_i^*$ is the approximation error, $W_i^* \in R^{l\times 1}$ are unknown ideal constant weights. $\hat{W}_i$ is the estimating value of W_i^* in (20). The weight estimation error is $\tilde{W}_i = \hat{W}_i - W_i^*$. Using the same procedure as step 1, we have

$$\dot{V}_i \le -\lambda_i V_i + \left(\theta_i v(k_i) + 1\right)\dot{k}_i + c_i + \theta_i z_{i+1}^2 \tag{24}$$

where constant $\lambda_i = \min\left\{1/2, \sigma_i / \lambda_{\max}\left(\Gamma_i^{-1}\right)\right\}$, $c_i = \frac{\sigma_i}{2}\left\|W_i^*\right\|^2 + \varepsilon_i^{*2}$.

According to Lemma 1, if z_{i+1} can be regulated such that it is square integrable, then V_i and $k_i(t)$, hence $z_i(t)$ and $\hat{W}_i$ are all bounded on $\left[0, t_f\right)$. Furthermore, $z_i(t)$ is square integrable.

Step n: This is the final step, since the actual control u appears in the derivative of z_n as given in

$$\dot{z}_n = f_n(\bar{x}_n) + g_n(\bar{x}_n)u - \dot{\alpha}_{n-1}. \tag{25}$$

Since α_{n-1} is a function of $\bar{x}_{n-1}$, k_{n-1}, $\hat{W}_1$, $\cdots$, and $\hat{W}_{n-1}$, $\dot{\alpha}_{n-1}$ can be expressed as

$$\dot{\alpha}_{n-1} = \sum_{j=1}^{n-1} \frac{\partial \alpha_{n-1}}{\partial x_j} \left(f_j\left(\overline{x}_j\right) + g_j\left(\overline{x}_j\right) x_{j+1} \right) + \zeta_{n-1}, \tag{26}$$

where $\zeta_{n-1} = \frac{\partial \alpha_{n-1}}{\partial k_{n-1}} \dot{k}_{n-1} + \sum_{j=1}^{n-1} \frac{\partial \alpha_{n-1}}{\partial \hat{W}_j} \dot{\hat{W}}_j$ is computable. Then, (25) can be expressed as

$$\dot{z}_n = \overline{f}_n\left(\overline{x}_n, \zeta_{n-1}\right) + g_n\left(\overline{x}_n\right) u, \tag{27}$$

here, we assume $\overline{f}_n\left(\overline{x}_n, \zeta_{n-1}\right) = f_n\left(\overline{x}_n\right) - \dot{\alpha}_{n-1}$ is unknown smooth function.

We now design the following actual adaptive control

$$\begin{aligned} u &= v\left(k_n\right)\left[z_n + \hat{W}_n^T \varphi_n\left(Z_n\right) \right] \\ \dot{k}_n &= z_n^2 + \hat{W}_n^T \varphi_n\left(Z_n\right) z_n \qquad , \\ \dot{\hat{W}}_n &= \Gamma_n \varphi_n\left(Z_n\right) z_n - \Gamma_n \sigma_n \hat{W}_n \end{aligned} \tag{28}$$

and consider the following Lyapunov function candidate

$$V_n = \frac{1}{2\left|g_n\left(\overline{x}_n\right)\right|} z_n^2 + \frac{1}{2} tr\left(\tilde{W}_n^T \Gamma_n^{-1} \tilde{W}_n \right), \tag{29}$$

where $\Gamma_n = \Gamma_n^T > 0$ is constant matrix. Its time derivative is

$$\dot{V}_n = z_n \left(\frac{\overline{f}_n\left(\overline{x}_n, \zeta_{n-1}\right)}{\left|g_n\left(\overline{x}_n\right)\right|} + \theta_n u \right) + \tilde{W}_n^T \Gamma_n^{-1} \dot{\tilde{W}}_n. \tag{30}$$

Assuming that

$$\Delta_n\left(Z_n\right) \triangleq \frac{\overline{f}_n\left(\overline{x}_n, \zeta_{n-1}\right)}{\left|g_n\left(\overline{x}_n\right)\right|} = W_n^{*T} \varphi_n\left(Z_n\right) + \varepsilon_n\left(Z_n\right), \tag{31}$$

where $Z_n = \left[\overline{x}_n^{\,T}, \zeta_{n-1} \right]^T$ is input vector of RBF neural network, $\left|\varepsilon_n\left(Z_n\right)\right| < \varepsilon_n^*$ is the approximation error, $W_n^* \in R^{l\times 1}$ are unknown ideal constant weights. $\hat{W}_n$ is the estimating value of W_n^* in (28). The weight estimation error is $\tilde{W}_n = \hat{W}_n - W_n^*$. Using the same procedure as step 1, we have

$$\dot{V}_n \le -\lambda_n V_n + \left(\theta_n v\left(k_n\right) + 1\right) \dot{k}_n + c_n, \tag{32}$$

where constant $\lambda_n = \min\left\{ 3/4, \sigma_n / \lambda_{\max}\left(\Gamma_n^{-1}\right) \right\}$, $c_n = \frac{\sigma_n}{2} \left\| W_n^* \right\|^2 + \varepsilon_n^{*2}$. (32) and λ_n are different with (24) and λ_i because here u is the actual control.

So far the design procedure is complete. Applying Lemma 1 to (32), we conclude that V_n and k_n are bounded; furthermore, z_n is square integrable, all on $\left[0, t_f\right)$. Applying Lemma 1 $n-1$ times, it can be shown from the above design procedure that V_i and k_i, $1 \le i \le n-1$, and all estimates $\hat{W}_i$, $1 \le i \le n$, and in turn α_i, $1 \le i \le n-1$ and the original state x_i, $1 \le i \le n$, are also bounded on $\left[0, t_f\right)$. Therefore, no finite time escape phenomenon may occur and $t_f = \infty$. As an immediate result, $u, \dot{x}_i, \dot{z}_i$ and $\ddot{x}_i$, $1 \le i \le n$, are also bounded on $\left[0, \infty\right)$. Thus, using Barbalat's lemma, we can conclude that $\lim_{t\to\infty} x_1\left(t\right) = \lim_{t\to\infty} z_1\left(t\right) = 0$, $\lim_{t\to\infty} \dot{x}_1\left(t\right) = 0$, $1 \le i \le n$. Furthermore, since $f_i\left(0\right) = 0$, $1 \le i \le n$, it can also shown from (1) that $\lim_{t\to\infty} x_i\left(t\right) = 0$, $2 \le i \le n$. Then, we can obtain the following results.

Theorem. Suppose that the proposed design procedure is applied to system (1), then for all initial conditions, uniform boundedness of all signals in the resulting closed-loop system is guaranteed; furthermore, regulation of the state $x(t)$ is achieved, $\lim_{t\to\infty} x(t) = 0$.

IV. Illustrative Example

Consider the following second-order nonlinear system

$$\begin{cases} \dot{x}_1 = 3x_1^2 + \sin(-6x_2) - 6x_2 \\ \dot{x}_2 = \sin(-6x_2) + 6u \end{cases}$$

where the dynamic of the system is unknown. We choose the initial condition $[x_1(0), x_2(0)]^T = [-2, 2]^T$. The controller is designed using the above design procedure. In the simulation, the first RBF neural network contain 9 hidden nodes, and the second RBF neural network contain 36 hidden nodes. The design parameters of the controller are $\Gamma_1 = \Gamma_2 = 0.5$, $\sigma_1 = \sigma_2 = 0.1$. We choose the initial weights $\hat{W}_1 = \hat{W}_2 = 0$. Figs.1-3 depict the simulation results.

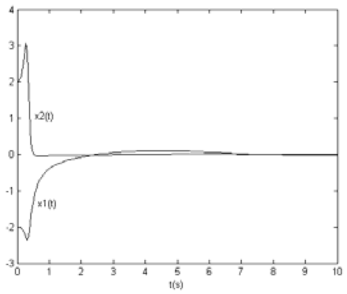

Fig. 1. State variables x_1 and x_2

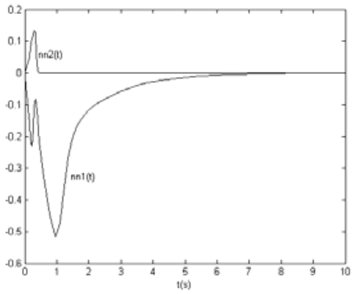

Fig. 2. Neural networks output

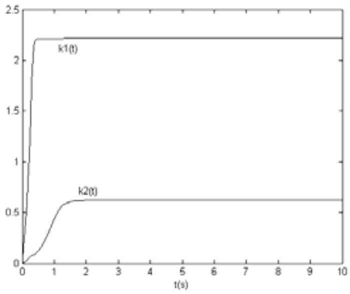

Fig. 3. Adaptive variables k_1 and k_2

V. Conclusion

An adaptive neural-based control has been proposed for a class of strict-feedback nonlinear systems with unknown control direction. It has been shown that Nussbaum-type gains can be incorporated in the backstepping design to counteract the lack of a prior knowledge of control directions. The proposed method avoids the controller singularity problem and overparameterization problem encountered in many results. Global stability results for the resulting closed-loop system have been established.

References

[1] M. Krstic, I. Kanellakopoulos, P. Kokotovic, *Nonlinear and adaptive control design*. Wiley–Interscience Publication, 1995, pp.81-105.

[2] M. M. Polycarpou, P. A. Ioannou, “Stable adaptive neural control scheme for nonlinear systems,” in IEEE Translations Automatic Control, vol. 3, 1996, pp. 447-451.

[3] T. Zhang, S. S. Ge, C. C. Hang, “Adaptive neural network control for strict-feedback nonlinear systems using backstepping design,” in Automatica, vol. 12, 2000, pp. 835-1846.

[4] S. S. Ge, C. Wang, “Robust Adaptive neural control for a class of perturbed strict feedback nonlinear Systems,” in IEEE Translations on Neural Networks, vol.6, 2002, pp. 1409-1419.

[5] S. S. Ge, F. Hong, H. L. Lee, “Adaptive neural control of nonlinear time-delay systems with unknown virtual control coefficients,” in IEEE Translations on System, Man, and Cybernetics, vol.1, 2004, pp. 499-516.
[6] A. Yesildirek, F. L. Lewis, “Feedback linearization using neural networks,” in Automatica, vol. 11, 1995, pp. 1659-1664.
[7] X. D. Ye, J. P. Jiang, “Adaptive nonlinear design without a priori knowledge of control directions,” in IEEE Translations on Automatic Control, vol. 11, 1998, pp. 1617-1621.
[8] B.Brogliato, R. Lozano, “Adaptive control of first order nonlinear systems with reduced knowledge of the plant parameters,” in IEEE Translations on Automatic Control, vol. 11, 1994, pp. 1764-1768.
[9] J. Kaloust, Z. Qu, “Continuous robust control design for nonlinear uncertain systems without a priori knowledge of control direction,” in IEEE Translations on Automatic Control, vol. 2, 1995, pp. 276-282.

Y. Q. Jin was born in Hebei, China, in 1977. Now he is a doctoral candidate of Naval Aeronautical Engineering Institute of China. His current research interests are in the area of nonlinear control, and neural network theory.

S. X. Wang was born in Jiangsu, China, in 1972. Now he is a lecturer of Naval Aeronautical Engineering Institute of China. His current research interests are control theory, system simulation.

Z. C. Xiao was born in Hubei, China, in 1977. Now he is a lecturer of Naval Aeronautical Engineering Institute of China. His research interests are nonlinear control, and virtual reality.