Adaptive Predictive Control Using Neural Network for a Class of Pure-Feedback Systems in Discrete Time

Shuzhi Sam Ge, *Fellow, IEEE*, Chenguang Yang, *Student Member, IEEE*, and Tong Heng Lee, *Member, IEEE*

***Abstract*—In this paper, adaptive neural network (NN) control is investigated for a class of nonlinear pure-feedback discrete-time systems. By using prediction functions of future states, the pure-feedback system is transformed into an n-step-ahead predictor, based on which state feedback NN control is synthesized. Next, by investigating the relationship between outputs and states, the system is transformed into an input–output predictor model, and then, output feedback control is constructed. To overcome the difficulty of nonaffine appearance of the control input, implicit function theorem is exploited in the control design and NN is employed to approximate the unknown function in the control. In both state feedback and output feedback control, only a single NN is used and the controller singularity is completely avoided. The closed-loop system achieves semiglobal uniform ultimate boundedness (SGUUB) stability and the output tracking error is made within a neighborhood around zero. Simulation results are presented to show the effectiveness of the proposed control approach.**

***Index Terms*—Discrete-time system, neural network, pure-feedback system.**

I. Introduction

THE last decade has witnessed an ever increasing research in adaptive neural network (NN) control since the introduction of NN for identification and control of nonlinear dynamical systems [1]. In the literature of adaptive NN control, NN is mostly used as approximation models for the unknown nonlinearities. Through years of progress, adaptive NN control has been shown to be particularly useful for control of highly uncertain, nonlinear, and complex systems owing to NN's excellent function approximation ability, and much significant development has been achieved [2]–[4].

For continuous-time systems, much research has been carried out on adaptive NN control of affine nonlinear systems that are feedback linearizable. An indirect NN control was presented for the systems with unknown constant control gain in [5]. For systems with unknown functions as control gains, adaptive NN control has been constructed in [6], where a switch action is taken to avoid potential singularity problem. In [7], adaptive NN control via backstepping design was presented for a class of minimum phase nonlinear systems with known relative degree. In [8], the combination of NN and backstepping has been proposed for multiple-input–multiple-output (MIMO) nonlinear systems in block-triangular form.

In contrast to the large amount of work on affine systems, only a few results are available in adaptive NN control of nonaffine continuous-time systems, because of the lack of mathematical tools for nonaffine systems compared with affine systems, e.g., the feedback linearization used for many affine systems is not directly applicable to nonaffine systems. To control continuous-time nonaffine systems in normal form, adaptive NN control design using implicit function was first proposed in [9], and NN inverse control was studied in [10] to invert the nonlinear dynamics such that the resulting tracking error dynamics are almost linear. Based on implicit function theory, adaptive NN control using backstepping was constructed for two special classes of nonaffine pure-feedback systems [11]. However, to extend the control design to more general nonaffine pure-feedback form, one technical difficulty arises when NN is used to approximate the control u in backstepping design; u and $\dot{u}$ will be involved as inputs to NN. This will lead to a circular construction of the practical control as indicated in [12], in which the difficulty was solved by proposing an input-to-state stability (ISS) modular approach with implicit function theory used to ensure the existence of desired virtual controls.

Comparing to nonlinear continuous-time systems, adaptive control is less developed for nonlinear discrete-time systems. The same concepts in continuous time and discrete time may have different meanings. For example, the "relative degrees" defined for continuous-time and discrete-time systems have totally different physical explanations [13]. As a consequence, many elegant control schemes for continuous-time systems may be not suitable for discrete-time systems. For instance, Lyapunov design for nonlinear discrete-time systems becomes much more intractable than for continuous-time systems because the linearity property of the derivative of a Lyapunov function in continuous-time is not present in the difference of the Lyapunov function in discrete time [14]. However, there are still considerable advances in NN control for discrete-time systems [15]–[18]. For systems with general relative degree, multilayer NN control was studied through backpropagation [15] and for nonlinear discrete-time systems in normal form, NN control with filtered tracking error was proposed in [16].

Manuscript received April 2, 2007; revised September 29, 2007 and November 30, 2007; accepted January 10, 2008. First published August 15, 2008; current version published September 4, 2008.

The authors are with the Social Robotics Lab, Interactive Digital Media Institute (IDMI) and the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576, Singapore (e-mail: samge@nus.edu.sg; cgyang82@gmail.com; eleleeth@nus.edu.sg).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNN.2008.2000446

Noncausal problems will be encountered if we directly apply backstepping design to discrete-time systems in lower triangular form, because discrete-time systems are described by difference equations, which involve state variables at different instants. To solve the noncausal problem, coordinate-transformation-based backstepping, the approach that "looks ahead" and chooses the control law to force the states to acquire their desired values, was proposed in [19] for parameter-strict-feedback discrete-time systems, but it is not clear how to extend this technique to more general systems. To control more general strict-feedback discrete-time systems, system transformation using prediction functions of future states was studied in [17], in which adaptive NN backstepping design has been applied to the transformed system without noncausal problem.

Similarly to continuous-time systems, it is noticed that most of the results for controlling discrete-time systems are limited to affine systems that are feedback linearizable. To control nonaffine discrete-time systems, implicit function-theory-based adaptive NN control was first studied in [20] and it is further developed with multilayer neural network (MNN) for general nonlinear autoregressive moving average with eXogenous inputs (NARMAX) systems in [21]. A novel linearization based on the NN identified model was proposed in [22] and then NN control was designed with restriction on the control growth $\Delta u(k)$. In [23], the nonaffine discrete-time system was decomposed into a linear part and a nonlinear part, and the nonlinear part was compensated by using an additive NN control. This method was also adopted in [24], where multiple models with a switching action were used for control design.

In this paper, as an effort to further explore adaptive NN control of nonaffine systems in discrete time, we will study direct adaptive NN control of pure-feedback systems based on prediction approach, with implicit function theorem exploited to assert the existence of a desired control input.

The main contributions of the paper are as follows.

1) By states and outputs prediction, the pure-feedback discrete-time systems are transformed into an n-step predictor and then transformed into an input–output model. After transformation, both state feedback and output feedback controls are synthesized with employment of a single NN.
2) To solve the difficulty of nonaffine appearance of control input, implicit function theory is used to assert the existence of an ideal control.
3) The proposed NN control achieves SGUUB stability and the bounds of the closed-loop signals are explicitly given.

Throughout this paper, the following notations are used.

- Z_0^+ stands for nonnegative integers.
- $\|\cdot\|$ denotes the Euclidean norm of vectors and induced norm of matrices.
- $A := B$ means that B is defined as A.
- $[\,]^T$ represents the transpose of a vector or a matrix.
- $\mathbf{0}_{[\mathbf{p}]}$ stands for p-dimension zero vector.
- W^* and $\hat{W}(k)$ denote the ideal neural weight and the estimate of neural weight at the kth step, respectively. Let $\tilde{W}(k) = \hat{W}(k) - W^*$ denote the estimate error.

II. Problem Formulation and Preliminaries

A. System Representation

Consider the following single-input–single-output (SISO) discrete-time systems in pure-feedback form:

$$\begin{cases} \xi_i(k+1) = f_i\left(\bar{\xi}_i(k), \xi_{i+1}(k)\right) \\ i = 1,2,\ldots,n-1, \ \ n \geq 2 \\ \xi_n(k+1) = f_n\left(\bar{\xi}_n(k), u(k), d(k)\right) \\ y(k) = \xi_1(k) \end{cases} \tag{1}$$

where $\bar{\xi}_i(k) = [\xi_1(k), \xi_2(k), \ldots, \xi_i(k)]^T$, $j = 1, 2, \ldots, n$, are system states, $f_i(\cdot,\cdot) : R^{i+1} \to R$ and $f_n(\cdot,\cdot,\cdot) : R^{n+2} \to R$ are unknown nonlinear functions, $u(k) \in R$ and $y(k) \in R$ are system input and output, respectively, and $d(k)$ denotes the external disturbance, which is bounded by a constant $\bar{d}$, i.e., $|d(k)| \leq \bar{d}$.

Assumption 1: System functions $f_i(\cdot,\cdot) : R^i \times R \to R$, $i = 1, 2, \ldots, n-1$, and $f_n(\cdot,\cdot,0)$ in (1) are continuous with respect to all the arguments and continuously differentiable with respect to the second argument.

Assumption 2: There exist constants $\bar{g}_j > \underline{g}_j > 0$ such that $\underline{g}_j \leq |g_{1,j}(\cdot)| \leq \bar{g}_j$, $j = 1, 2, \ldots, n$, where $g_{1,i}(\cdot) = \partial f_i(\bar{\xi}_i(k), \xi_{i+1}(k))/\partial \xi_{i+1}(k)$, $i = 1, 2, \ldots, n-1$, and $g_{1,n}(\cdot) = \partial f_n(\bar{\xi}_n(k), u(k), d(k))/\partial u(k)$.

This assumption implies that the partial derivatives, $g_{1,j}(\cdot)$, $j = 1, 2, \ldots, n$, are strictly either positive or negative. Without losing generality, it is assumed that the signs of the partial derivatives are all positive. Let us introduce the notations: $\underline{g} = \Pi_{j=1}^n \underline{g}_j$ and $\bar{g} = \Pi_{j=1}^n \bar{g}_j$, which are to be used in the control design later.

Assumption 3: The system functions $f_i(\cdot, 0)$, $i = 1, 2, \ldots, n-1$, and $f_n(\cdot, 0, \cdot)$ are Lipschitz functions.

The control objective is to synthesize an NN control $u(k)$ for system (1) such that all signals in the closed-loop systems are bounded and the output $y(k)$ tracks a bounded reference trajectory $y_d(k) \in \Omega_{yd}$, where Ω_{yd} is a compact set.

Remark 1: The nonaffine pure-feedback form described in (1) includes a large class of systems. Actually, many cascaded physical systems that can be expressed in lower triangular form fall into this category, e.g., direct current (dc) motor system [25], coupled tank system [26], aircraft flight control system [27], Duffing oscillator [28], continuous stirred tank reactor (CSTR) system [29], etc.

B. HONN Approximation

There are many well-developed approaches used to approximate an unknown function. Artificial neural networks (ANNs) are one of the most frequently employed approximation methods due to the fact that ANNs are shown to be capable of universally approximating any unknown function to arbitrary precision [30]–[32]. Similar to biological neural networks, ANNs consist of massive simple processing units that correspond to biological neurons. With the highly parallel structure, ANNs are of powerful computing ability and intelligence of learning and adaptation with respect to fresh and unknown data. Higher order neural network (HONN) is a kind of linearly parametrized neural network (LPNN), and because

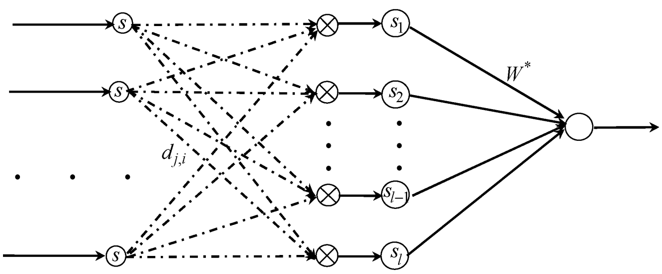


Fig. 1. HONN structure.

of its higher order interaction between neurons, HONN is of strong capacity and can approximate any continuous function to any desired accuracy over a compact set [4]. The structure of HONN is expressed as follows:

$$\phi(W,z) = W^T S(z) \quad W, S(z) \in R^l$$
$$S(z) = [s_1(z), s_2(z), \ldots, s_l(z)]^T \tag{2}$$
$$s_i(z) = \prod_{j\in I_i} [s(z_j)]^{d_j(i)}, \qquad i = 1,2,\ldots,l \tag{3}$$

where $z \in \Omega_z \subset R^m$ is the input to HONN, l is the NN nodes number, $\{I_1, I_2, \ldots, I_l\}$ is a collection of l not-ordered subsets of $\{1,2,\ldots,m\}$, specified by the designer, $d_{j,i}$'s are prescribed nonnegative integers, W is an adjustable synaptic weight vector, and $s(z_j)$ is a monotonically increasing and differentiable sigmoidal function. In this paper, it is chosen as a hyperbolic tangent function, i.e., $s(z_j) = (e^{z_j} - e^{-z_j})/(e^{z_j} + e^{-z_j})$. The HONN structure is shown in Fig. 1, where the dashed lines mean they may be connected or unconnected depending on I_i, and $d_{j,i}$ is the power of the $s(\cdot)$ function.

For a smooth function $\varphi(z)$ over a compact set $\Omega_z \subset R^m$, given a small constant real number $\mu^* > 0$, if l is sufficiently large, there exists a set of ideal bounded weights W^* such that

$$\max|\varphi(z) - \phi(W^*, z)| < \mu(z), \qquad |\mu(z)| < \mu^*. \tag{4}$$

From the universal approximation results for neural networks [33], it is known that the constant μ^* can be made arbitrarily small by increasing the NN nodes number l.

Lemma 1 [18]: Consider the basis functions of HONN (2) with z being the input vector. The following properties hold:

$$\left|\lambda_{\max}\left[S(z)S^T(z)\right]\right| < l, \qquad S^T(z)S(z) < l \tag{5}$$

where $\lambda_{\max}(M)$ denotes the maximal magnitude eigenvalue of M.

C. Preliminaries

The following lemmas and definitions will be used for control design and stability analysis in the remainder of this paper.

Definition 1 [18]: The future state variables of a discrete-time control system is said to be semidetermined future states (SDFS) at time instant k, if it can be determined based on the available system information up to time instant k, and controls up to time instant $k-1$ under the assumption that the dynamics of the plant and the disturbance are known.

Definition 2: Let U be an open subset of R^{i+1}. A mapping $f(\omega) : U \to R$ is said to be Lipschitz on U, if there exists a positive constant L such that

$$|f(\omega_a) - f(\omega_b)| \le L\|\omega_a - \omega_b\|$$

for all $(\omega_a, \omega_b) \in U$.

Definition 3 [34]: A trajectory $x(k)$ of the closed-loop system is said to be semiglobally uniformly ultimately bounded (SGUUB), if for any *a priori* given compact set, there exists a feedback control, a bound $\mu \ge 0$, and a number $N(\mu, x_0)$, such that the trajectory of the closed-loop system starting from the compact satisfy $\|x(k)\| \le \mu$ for all $k \ge k_0 + N$.

Lemma 2 [35]: Let $f \in C^r[R^k \times R^n]$ with $f(a,b) = \mathbf{0}_{[n]}$ and rank $[D_f(a,b)] = n$ where $D_f(a,b) = (\partial f(x,y)/\partial y)|_{(x,y)=(a,b)}$ is an $n \times n$ matrix. Then, there exists a neighborhood A of a in R^k and a unique C^r function $g : A \to R^n$ such that $g(a) = b$ and $f(x, g(x)) = \mathbf{0}_{[n]}$, for all $x \in A$.

Lemma 3: Under Assumptions 1–3, the states, and input of system (1) satisfy

$$\|\bar{\xi}_n(k)\| \le C_1 \max_{k\le j\le k+n-1}\{|y(j)|\} + C_2$$
$$|u(k)| \le C_3 \max_{k\le j\le k+n}\{|y(j)|\} + C_4 \tag{6}$$

where C_1, C_2, C_3, and C_4 are some finite constants.

Proof: See Appendix I. ■

Lemma 4: Let $V(k) = \sum_{i=1}^m V_i(k)$, where $V_i(k) \ge 0, \forall k \in Z_0^+$, $i = 1,2,\ldots,m$. If the following inequality holds:

$$V(k+1) \le \sum_{i=1}^m c_i(k)V_i(k) + b(k) \tag{7}$$

where $|c_i(k)| \le \bar{c} < 1$ and $|b(k)| \le \bar{b}$, then we have

$$V(k) \le V(0) + \frac{\bar{b}}{1-\bar{c}} \qquad \forall k \in Z_0^+$$
$$\lim_{k\to\infty}\sup\{V(k)\} \le \frac{\bar{b}}{1-\bar{c}}. \tag{8}$$

Proof: See Appendix II. ■

Corollary 1: Let $V(k) = \sum_{i=1}^m V_i(k)$, where $V_i(k) \ge 0$, $\forall k \in Z_0^+$. If the following inequality holds:

$$V(k+1) \le \sum_{i=1}^m c_i(k_1)V_i(k_1) + b(k_1),$$
$$k_1 = k-n+1, \qquad k \ge n-1, \qquad n \ge 1 \tag{9}$$

where $|c_i(k)| \le \bar{c} < 1$ and $|b(k)| \le \bar{b}$, then we have

$$V(k) \le \bar{V}(0) + \frac{\bar{b}}{1-\bar{c}}, \qquad k \ge n-1$$
$$\lim_{k\to\infty}\sup\{V(k)\} \le \frac{\bar{b}}{1-\bar{c}} \tag{10}$$

where $\bar{V}(0) = \max_{0\le j\le n-1}\{V(j)\}$.

Proof: See Appendix III. ■

Motivated by the result in continuous time [8], we have the following lemma.

Lemma 5: Define a positive–definite function $V(k) = V_1(k) + V_2(k)$ for system (1), with $V_1(k)$ and $V_2(k)$ given by

$$V_1 = a_e e^2(k)$$
$$V_2 = a_W \tilde{W}^T(k)\tilde{W}(k)$$

where $e(k) = y(k) - y_d(k)$, $y_d(k) \in \Omega_{yd}$, is output tracking error, $W^* \in R^l$ and $\hat{W}(k) \in R^l$ are ideal NN weight and its estimation, $\tilde{W}(k) = \hat{W}(k) - W^*$ is the estimate error, and a_e and a_W are some positive constants. If the following inequality holds:

$$V(k+1) \le c_1(k)V_1(k_1) + c_2(k)V_2(k_1) + b(k), \quad k_1 = k - n + 1, \quad k \ge n-1 \tag{11}$$

where $|c_i(k)| < \bar{c} < 1$, $i = 1, 2$, and $|b(k)| < \bar{b}$, then given any initial compact set defined by

$$\begin{aligned}\Omega_0 =& \Omega_{\xi_0} \times \Omega_{\hat{W}_0}\\ =& \left\{\bar{\xi}_n(0) \middle| \left\|\bar{\xi}_n(0)\right\| \le C_1 C_{e0} + C_1 \max\{|y_d(i)|\} + C_2\right\}\\ &\times \left\{\hat{W}(i) \middle| \left\|\hat{W}(i)\right\| \le \|W^*\| + C_{\tilde{W}0}\right\}, \\ & \qquad i = 0, 1, \ldots, n-1\end{aligned}$$

where C_1 and C_2 are the coefficients introduced in Lemma 3, C_{e0} and $C_{\tilde{W}0}$ are defined as

$$C_{e0} = \max_{0 \le i \le n-1}\{|e(i)|\} \quad C_{\tilde{W}0} = \max_{0 \le i \le n-1}\left\{\left\|\tilde{W}(i)\right\|\right\}. \tag{12}$$

Then, we have the following conclusions.

1) The states $\bar{\xi}_n(k)$ and the NN weight estimate $\hat{W}(k)$ remain in the compact set defined by

$$\begin{aligned}\Omega =& \Omega_{\xi} \times \Omega_{\hat{W}}\\ =& \left\{\bar{\xi}_n(k) \middle| \left\|\bar{\xi}_n(k)\right\| \le C_1 \sup_{y_d(k) \in \Omega_{yd}}\{|y_d(k)|\} \right.\\ & \left. + C_1 c_{e\max} + C_2\right\}\\ & \times \left\{\hat{W}(k) \middle| \|\hat{W}(k)\| \le \|W^*\| + c_{\tilde{W}\max}\right\}.\end{aligned}$$

2) The states $\bar{\xi}_n(k)$ and the NN weight estimate $\hat{W}(k)$ will eventually converge to the compact set defined by

$$\begin{aligned}\Omega_s =& \Omega_{\xi_s} \times \Omega_{\hat{W}_s}\\ =& \left\{\bar{\xi}_n(k) \middle| \left\|\bar{\xi}_n(k)\right\| \le C_1 \sup_{y_d(k) \in \Omega_{yd}}\{|y_d(k)|\} + C_1 c_{es} + C_2\right\}\\ &\times \left\{\hat{W}(k) \middle| \|\hat{W}(k)\| \le \|W^*\| + c_{\tilde{W}s}\right\}\end{aligned} \tag{13}$$

where constants

$$\begin{aligned}c_{e\max} &= \sqrt{\frac{1}{a_e}\left(C_0 + \frac{\bar{b}}{1-\bar{c}}\right)}\\ c_{\tilde{W}\max} &= \sqrt{\frac{1}{a_W}\left(C_0 + \frac{\bar{b}}{1-\bar{c}}\right)}\\ c_{es} &= \sqrt{\frac{\bar{b}}{a_e(1-\bar{c})}}\end{aligned} \tag{14}$$

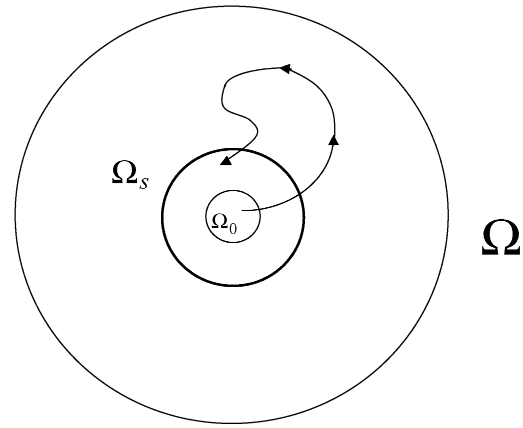


Fig. 2. Three compact sets (continuous-time case is documented in [8]).

$$\begin{aligned}c_{\tilde{W}s} &= \sqrt{\frac{\bar{b}}{a_W(1-\bar{c})}}\\ C_0 &= a_e C_{e0}^2 + a_W C_{\tilde{W}0}^2.\end{aligned} \tag{15}$$

Proof: See Appendix IV. ■

Remark 2: The three compact sets defined in Lemma 5, the initial compact set Ω_0, the bounding compact set Ω, and the steady-state compact set Ω_s, are illustrated in Fig. 2. It is noted in (14) and (15) that the size of Ω_0 (C_0) only affects the bounding compact set Ω but not affects the steady-state compact set Ω_s.

Remark 3: According to Lemma 5, given any initial condition Ω_0, if there is a control that guarantees the validness of (11) on the bounding compact set Ω, then the closed-loop signals are SGUUB in accordance with Definition 3.

III. Transformation for State Feedback Control

In this section, we will show that the future states $\bar{\xi}_i(k + n - i)$ in system (1), $i = 1, 2, \ldots, n-1$, are SDFS, and then, the system is transformed into an n-step predictor, which only involves current states and input.

A. Future States Prediction

It is noted in system (1) that among the future states at the $(k+1)$th step, only the last state $\xi_n(k+1)$ depends on the control input, while other $(n-1)$ states are independent of $u(k)$. Therefore, they can be predicted at the kth step provided that the system dynamics are known exactly. This implies that these states are SDFS. The prediction functions of one step ahead states are as follows:

$$\bar{\xi}_i(k+1) = \begin{bmatrix}\xi_1(k+1)\\ \vdots\\ \xi_i(k+1)\end{bmatrix} = \begin{bmatrix}\phi_{1,1}\left(\bar{\xi}_2(k)\right)\\ \vdots\\ \phi_{1,i}\left(\bar{\xi}_{i+1}(k)\right)\end{bmatrix}, \qquad i = 1, 2, \ldots, n-1 \tag{16}$$

where

$$\phi_{1,i}\left(\bar{\xi}_{i+1}(k)\right) = f_i\left(\bar{\xi}_i(k), \xi_{i+1}(k)\right), \qquad i = 1, 2, \ldots, n-1$$

For convenience, (16) is written as vector functions

$$\bar{\xi}_i(k+1) = \Phi_{1,i}\left(\bar{\xi}_{i+1}(k)\right), \qquad i = 1, 2, \ldots, n-1. \quad (17)$$

According to Assumption 2, it can be checked that

$$\frac{\partial \phi_{1,i}\left(\bar{\xi}_{i+1}(k)\right)}{\partial \xi_{i+1}(k)} = g_{1,i}(\cdot) > 0. \quad (18)$$

Moving one step forward in (16) and using the predicted states vector in (17), we see that the first $(n-2)$ states at the $(k+2)$th step are still independent of control $u(k)$, and thus, they are SDFS

$$\begin{aligned}\bar{\xi}_i(k+2) &= \begin{bmatrix} \xi_1(k+2) \\ \vdots \\ \xi_i(k+2) \end{bmatrix} = \begin{bmatrix} \phi_{1,1}\left(\bar{\xi}_2(k+1)\right) \\ \vdots \\ \phi_{1,i}\left(\bar{\xi}_{i+1}(k+1)\right) \end{bmatrix} \\ &= \begin{bmatrix} \phi_{1,1}\left(\Phi_{1,2}\left(\bar{\xi}_3(k)\right)\right) \\ \vdots \\ \phi_{1,i}\left(\Phi_{1,i+1}\left(\bar{\xi}_{i+2}(k)\right)\right) \end{bmatrix} \\ &= \begin{bmatrix} \phi_{2,1}\left(\bar{\xi}_3(k)\right) \\ \vdots \\ \phi_{2,i}\left(\bar{\xi}_{i+2}(k)\right) \end{bmatrix}, \qquad i = 1, 2, \ldots, n-2 \quad (19)\end{aligned}$$

where

$$\phi_{2,i}\left(\bar{\xi}_{i+2}(k)\right) = \phi_{1,i}\left(\Phi_{1,i+1}\left(\xi_{i+2}(k)\right)\right), \qquad i = 1, 2, \ldots, n-2.$$

Similar to the notation in (17), the above vector functions are denoted as

$$\bar{\xi}_i(k+2) = \Phi_{2,i}\left(\bar{\xi}_{i+2}(k)\right), \qquad i = 1, 2, \ldots, n-2. \quad (20)$$

Continuing the procedure above iteratively, after $(n-2)$ steps, it is noted that the first state at the $(k+n-1)$th step can be predicted by the states at the kth step as follows:

$$\xi_1(k+n-1) = \phi_{1,1}\left(\Phi_{n-2,2}\left(\bar{\xi}_n(k)\right)\right) := \phi_{n-1,1}\left(\bar{\xi}_n(k)\right) \quad (21)$$

where vector-valued functions $\Phi_{j,i}(\bar{\xi}_{j+i}(k))$, $j = 3, 4, \ldots, n-2$, $i = 1, 2, \ldots n-j$, are defined consistently via the above procedure. Then, we see that $\xi_1(k+n-1)$ is still an SDFS.

For consistency, we denote

$$\bar{\xi}_1(k+n-1) = \phi_{n-1,1}\left(\bar{\xi}_n(k)\right) := \Phi_{n-1,1}\left(\bar{\xi}_n(k)\right). \quad (22)$$

B. The n-Step-Ahead Predictor Form

If the dynamics of systems (1) is exactly known, it has been shown from (17)–(22) that the future states $\bar{\xi}_i(k+n-i)$, $i = 1, 2, \ldots, n-1$, are SDFS and can be obtained by the prediction functions $\Phi_{n-i,i}(\bar{\xi}_n(k))$, which are functions of current states.

Substituting these predicted future states into system (1), it is obtained

$$\begin{cases} \xi_1(k+n) = \phi_{1,1}\left(\Phi_{n-1,1}\left(\bar{\xi}_n(k)\right), \xi_2(k+n-1)\right) \\ \xi_2(k+n-1) = \phi_{1,2}\left(\Phi_{n-2,2}\left(\bar{\xi}_n(k)\right), \xi_3(k+n-2)\right) \\ \qquad \vdots \\ \xi_n(k+1) = \phi_{1,n}\left(\bar{\xi}_n(k), u(k), d(k)\right) \\ y(k+n) = \xi_1(k+n) \end{cases} \quad (23)$$

where $\phi_{1,n}(\bar{\xi}_n(k), u(k), d(k))$ is defined in the following for continuity:

$$\phi_{1,n}\left(\bar{\xi}_n(k), u(k), d(k)\right) = f_n\left(\bar{\xi}_n(k), u(k), d(k)\right). \quad (24)$$

Remark 4: To facilitate the control design, we consider combining the n equations in (23) together rather than applying backstepping to (23) directly as in [17], where n NNs are required to generate a control input.

Replacing $\xi_2(k+n-1)$ in the first equation of (23) with the right-hand side of the second equation yields

$$\begin{aligned}\xi_1(k+n) = \phi_{1,1}\big(&\Phi_{n-1,1}\left(\bar{\xi}_n(k)\right), \\ &\phi_{1,2}\left(\Phi_{n-2,2}\left(\bar{\xi}_n(k)\right), \xi_3(k+n-2)\right)\big).\end{aligned}$$

Continuing to iteratively replace $\xi_j(k+n-j+1)$ in the above equation with the right-hand side of the jth equation in (23), $j = 3, 4, \ldots, n-1$, until $u(k)$ appears at the last step, we obtain

$$y(k+n) = \xi_1(k+n) = \phi\left(\bar{\xi}_n(k), u(k), d(k)\right) \quad (25)$$

where

$$\begin{aligned}\phi&\left(\bar{\xi}_n(k), u(k), d(k)\right) \\ &= \phi_{1,1}\Bigg(\Phi_{n-1,1}\left(\bar{\xi}_n(k)\right), \\ &\qquad \phi_{1,2}\Bigg(\Phi_{n-2,2}\left(\bar{\xi}_n(k)\right), \\ &\qquad\qquad \phi_{1,3}\left(\ldots, \phi_{1,n}\left(\bar{\xi}_n(k), u(k), d(k)\right)\ldots\right)\Bigg)\Bigg).\end{aligned} \quad (26)$$

Now the original pure-feedback system (1) is transformed into the n-step-ahead predictor.

IV. State Feedback Adaptive NN Control

The n-step-ahead predictor function (25) can be written as

$$\begin{aligned}y(k+n) &= \phi\left(\bar{\xi}_n(k), u(k), d(k)\right) \\ &= \phi_s\left(\bar{\xi}_n(k), u(k)\right) + d_s(k)\end{aligned} \quad (27)$$

where

$$\begin{aligned}\phi_s\left(\bar{\xi}_n(k), u(k)\right) &= \phi\left(\bar{\xi}_n(k), u(k), 0\right) \\ d_s(k) &= \phi\left(\bar{\xi}_n(k), u(k), d(k)\right) - \phi\left(\bar{\xi}_n(k), u(k), 0\right).\end{aligned}$$

According to Assumption 3, there exists some finite constant L_d such that

$$|d_s(k)| \le L_d\,|d(k)| \le L_d \bar{d} := \bar{d}_s. \tag{28}$$

For $\phi_s(\bar{\xi}_n(k), u(k))$, from (18) and (23), it is easy to show that

$$\begin{aligned} 0 < \underline{g} < & \frac{\partial \phi_s\left(\bar{\xi}_n(k), u(k)\right)}{\partial u(k)} \\ &= g_{1,1}(\cdot) g_{1,2}(\cdot) \ldots g_{1,n}(\cdot) := g_s(\cdot) < \bar{g}. \end{aligned} \tag{29}$$

Denote $e(k) = y(k) - y_d(k)$ and we have

$$e(k+n) = \phi_s\left(\bar{\xi}_n(k), u(k)\right) - y_d(k+n) + d_s(k). \tag{30}$$

From (29), it is clear that

$$\frac{\partial \phi_s\left(\bar{\xi}_n(k), u(k)\right) - y_d(k+n)}{\partial u(k)} = g_s(\cdot) > 0.$$

According to Lemma 2, there exists a continuous ideal control input $u_s^*(z(k))$ such that

$$\begin{aligned} &\phi_s\left(\bar{\xi}_n(k), u_s^*\left(z(k)\right)\right) - y_d(k+n) = 0 \\ &z(k) = \left[\bar{\xi}_n^T(k), y_d(k+n)\right]^T \in \Omega_z \in R^{n+1}. \end{aligned} \tag{31}$$

Substituting this ideal control $u_s^*(z(k))$ into (30) results in $e(k+n) = d_s(k)$, $\forall k \in Z_0^+$. This means that the ideal control $u_s^*(z(k))$ is an n-step deadbeat control because, after n steps, we have $y(k+n) = y_d(k+n)$, if $d_s(k) = 0$. It is known that $d_s(k)$ is bounded, then $y(k)$ will be bounded. According to Lemma 3, the ideal control input $u_s^*(z(k))$ is bounded. From Section II-B, there exists an NN with ideal weight vector $W_s^* \in R^{l_s}$ such that $u_s^*(z(k))$ can be approximated in the following manner:

$$\begin{aligned} u_{nn}^*\left(z(k)\right) &= W_s^{*T} S\left(z(k)\right), \;\; S\left(z(k)\right) \in R^{l_s} \\ u_s^*\left(z(k)\right) &= u_{nn}^*\left(z(k)\right) + \mu\left(z(k)\right), \; \forall z(k) \in \Omega_z \end{aligned} \tag{32}$$

where $\Omega_z = \Omega_\xi \times \Omega_{yd}$ and $\mu(z(k))$ is the NN function approximation error that can be made arbitrary small by increasing NN neurons number l_s.

Consider the following control with an adaptive HONN as an approximator of $u_s^*(z(k))$:

$$\begin{aligned} u(k) &= \frac{\eta_s(k)}{\bar{g}} e(k) + \hat{u}_s\left(z(k)\right) \\ \hat{u}_s\left(z(k)\right) &= \hat{W}_s^T(k) S\left(z(k)\right) \end{aligned} \tag{33}$$

where $|\eta_s(k)| \le \bar{\eta}_s < 1$ is a scaling factor and $\hat{W}_s(k)$ is the estimate of ideal NN weight W_s^* and it is updated by the adaptation law

$$\begin{aligned} \hat{W}_s(k+1) = \hat{W}_s(k_1) - \gamma_s S\left(z(k_1)\right) e(k+1) - \sigma_s \hat{W}_s(k_1), \\ k_1 = k - n + 1 \end{aligned} \tag{34}$$

where $0 < \sigma_s < 1$ and $\gamma_s > 0$ are NN tuning parameters to be chosen.

Remark 5: It should be noted that the update law (34) is presented at the $(k+1)$th step, and the weight $\hat{W}_s(k+1)$ is obtained using information at the $(k+1)$th step. On the other hand, the control (33) employing $\hat{W}_s(k)$ is presented at the kth step and it only involves information at kth step.

Theorem 1: The closed-loop adaptive system consisting of the plant (1), the adaptive NN control (33), and the NN adaptation law (34) achieves SGUUB stability, provided that Assumptions 1–3 hold, and the design parameters $0 < \sigma_s < 1$, $0 < \bar{\eta}_s < 1$, and γ_s are suitably chosen such that

$$2\gamma_s \bar{g} l_s + \bar{\eta}_s \bar{g} + \bar{\eta}_s < 1. \tag{35}$$

Furthermore, the tracking error and the NN weight estimation error are ultimately bounded as

$$\lim_{k\to\infty} \sup \left\{ |e(k)|^2 + \frac{\bar{g}}{\gamma_s} \|\tilde{W}_s(k)\|^2 \right\} \le \frac{\bar{b}}{1-\bar{c}}$$

where

$$\begin{aligned} \bar{b} &= \frac{\bar{g}}{\bar{\eta}_s} \mu_s^{*2} + 2\frac{\bar{g}}{\gamma_s} \sigma_s \|W_s^*\|^2 \\ \bar{c} &= \max\left\{\bar{\eta}_s, (1-2\sigma_s)\right\} \\ \mu_s^* &= \mu^* + \frac{\bar{d}_s}{\underline{g}} \end{aligned} \tag{36}$$

and μ^* is the NN approximation bound defined in (4).

Proof: Adding and subtracting $\phi_s(\bar{\xi}_n(k), u_s^*(k))$ on the right-hand side of (30) leads to

$$\begin{aligned} e(k+n) &= \phi_s\left(\bar{\xi}_n(k), u(k)\right) - \phi_s\left(\bar{\xi}_n(k), u_s^*\left(z(k)\right)\right) + d_s(k) \\ &= g_s\left(\bar{\xi}_n(k), u^c(k)\right)\left(u(k) - u_s^*\left(z(k)\right)\right) + d_s(k) \end{aligned} \tag{37}$$

where $u^c(k) \in [\min\{u_s^*(z(k)), u(k)\}, \max\{u_s^*(z(k)), u(k)\}]$ and the last equality is obtained by using mean value theorem. For convenience, denote

$$g_s(k) = g_s\left(\bar{\xi}_n(k), u^c(k)\right) \quad S(k) = S\left(z(k)\right).$$

Combining (32), (33), and (37) yields

$$\begin{aligned} e(k+1) = \eta_s(k) \frac{g_s(k_1)}{\bar{g}} e(k_1) + g_s(k_1) \tilde{W}_s^T(k_1) S(k_1) \\ - g_s(k_1) \mu\left(z(k_1)\right) + d_s(k_1) \end{aligned} \tag{38}$$

where $\tilde{W}_s(k) = \hat{W}_s(k) - W_s^*$ is the NN weight estimation error.

First, we assume that the NN approximation ability is never violated such that (38) always holds, while we will show that it is indeed the case if initially the NN approximation range is constructed to cover a specified compact set, and the so-called circular argument in the literature does not apply here in this very proof. Choose a positive–definite function $V(k)$ as

$$\begin{aligned} V(k) &= V_1(k) + V_2(k) \\ V_1(k) &= e^2(k) \\ V_2(k) &= \frac{\bar{g}}{\gamma_s} \tilde{W}_s^T(k) \tilde{W}_s(k). \end{aligned} \tag{39}$$

It can be derived from (34) that

$$\tilde{W}_s(k+1) = \tilde{W}_s(k_1) - \gamma_s S(k_1) e(k+1) - \sigma_s \hat{W}_s(k_1). \tag{40}$$

From (38), it can be derived that

$$\tilde{W}^T(k_1)S(k_1)e(k+1) = \frac{e^2(k+1)}{g_s(k_1)} - \frac{\eta_s(k)}{\bar{g}}e(k_1)e(k+1) + e(k+1)\mu_s(k_1)$$

where

$$\mu_s(k_1) = \mu\left(z(k_1)\right) - \frac{d_s(k_1)}{g_s(k_1)}.$$

Noting the facts

$$\begin{aligned}
&0 < g_s(k_1) < \bar{g} \quad S^T(k)S(k) \le l_s \quad |\mu_s(k_1)| \le \mu_s^* \\
&2\tilde{W}_s^T(k_1)\hat{W}_s(k_1) = \tilde{W}_s^T(k_1)\tilde{W}_s(k_1) + \|\hat{W}_s(k)\|^2 - \|W_s^*\|^2 \\
&2\sigma_s\hat{W}_s^T(k_1)S(k_1)e(k+1) \le \gamma_s l_s e^2(k+1) + \frac{\sigma_s^2}{\gamma_s}\|\hat{W}_s(k_1)\|^2 \\
&\quad -2e(k+1)\mu_s(k_1) \le \bar{\eta}_s e^2(k+1) + \frac{\mu_s^2(k_1)}{\bar{\eta}_s} \\
&2\eta_s(k)e(k_1)e(k+1) \le \bar{\eta}_s e^2(k_1) + \bar{\eta}_s e^2(k+1)
\end{aligned}$$

we have the following inequality from (40):

$$\begin{aligned}
&V_2(k+1) \\
&= \frac{\bar{g}}{\gamma_s}\tilde{W}_s^T(k+1)\tilde{W}_s(k+1) \\
&= \frac{\bar{g}}{\gamma_s}\Big[\tilde{W}_s^T(k_1)\tilde{W}_s(k_1) + \gamma_s^2 S^T(k_1)S(k_1)e^2(k+1) \\
&\qquad + \sigma_s^2\|\hat{W}_s(k_1)\|^2 - 2\gamma_s\tilde{W}_s^T(k_1)S(k_1)e(k+1) \\
&\qquad - 2\sigma_s\tilde{W}_s^T(k_1)\hat{W}_s(k_1) \\
&\qquad + 2\gamma_s\sigma_s\hat{W}_s^T(k_1)S(k_1)e(k+1)\Big] \\
&\le \frac{\bar{g}}{\gamma_s}\tilde{W}_s^T(k_1)\tilde{W}_s(k_1) + \gamma_s l_s\bar{g}e^2(k+1) \\
&\quad + \frac{\bar{g}}{\gamma_s}\sigma_s^2\|\hat{W}_s(k_1)\|^2 - 2\frac{\bar{g}}{g_s(k_1)}e^2(k+1) \\
&\quad - 2\eta_s(k)e(k_1)e(k+1) - 2\bar{g}\mu_s(k_1)e(k+1) \\
&\quad - 2\frac{\bar{g}}{\gamma_s}\sigma_s\left(\tilde{W}_s^T(k_1)\tilde{W}_s(k_1) + \|\hat{W}_s(k)\|^2 - \|W_s^*\|^2\right) \\
&\quad + 2\bar{g}\sigma_s\hat{W}_s(k_1)S(k_1)e(k+1) \\
&\le \frac{\bar{g}}{\gamma_s}(1-2\sigma_s)\tilde{W}_s^T(k_1)\tilde{W}_s(k_1) + \bar{\eta}_s e^2(k_1) + \frac{\bar{g}}{\bar{\eta}_s}\mu_s^{*2} \\
&\quad + 2\frac{\bar{g}}{\gamma_s}\sigma_s\|W_s^*\|^2 + (2\gamma_s\bar{g}l_s + \bar{\eta}_s\bar{g} + \bar{\eta}_s - 2)e^2(k+1) \\
&\quad - 2\frac{\bar{g}}{\gamma_s}\sigma_s(1-\sigma_s)\|\hat{W}_s(k_1)\|^2.
\end{aligned} \tag{41}$$

Combining with

$$V_1(k+1) = e^2(k+1) \tag{42}$$

yields

$$\begin{aligned}
V(k+1) &= V_1(k+1) + V_2(k+1) \\
&\le \frac{\bar{g}}{\gamma_s}(1-2\sigma_s)\tilde{W}_s^T(k_1)\tilde{W}_s(k_1) + \bar{\eta}_s e^2(k_1) \\
&\quad + \frac{\bar{g}}{\bar{\eta}_s}\mu_s^{*2} + 2\frac{\bar{g}}{\gamma_s}\sigma_s\|W_s^*\|^2 \\
&\quad + (2\gamma_s\bar{g}l_s + \bar{\eta}_s\bar{g} + \bar{\eta}_s - 1)e^2(k+1) \\
&= \bar{\eta}_s V_1(k_1) + (1-2\sigma_s)V_2(k_1) + \bar{b} \\
&\quad + (2\gamma_s\bar{g}l_s + \bar{\eta}_s\bar{g} + \bar{\eta}_s - 1)e^2(k+1)
\end{aligned} \tag{43}$$

where

$$\bar{b} = \frac{\bar{g}}{\bar{\eta}_s}\mu_s^{*2} + 2\frac{\bar{g}}{\gamma_s}\sigma_s\|W_s^*\|^2. \tag{44}$$

If the parameters are chosen such that the following inequality holds:

$$2\gamma_s\bar{g}l_s + \bar{\eta}_s\bar{g} + \bar{\eta}_s < 1$$

then (43) becomes

$$V(k+1) \le \bar{\eta}_s V_1(k_1) + (1-2\sigma_s)V_2(k_1) + \bar{b}. \tag{45}$$

Let $a_e = 1$, $a_W = \bar{g}/\gamma_s$, and $\bar{c} = \max\{\bar{\eta}_s, (1-2\sigma_s)\}$. Noting that $0 < \bar{\eta}_s < 1$ and $0 < \sigma_s < 1$ and applying Lemma 5, we obtain the bounds on states and NN weights vector. According to Lemma 3, the control input is also bounded.

Now we show that the validness of NN approximation indeed holds given any initial condition Ω_0, if the NN used in (33) is predesigned with approximation range covering a specified compact set.

From Remark 2, we see that the bounding compact set Ω is determined by initial condition Ω_0 and control parameters. Thus, given any initial condition Ω_0, because the bounding compact set $\Omega = \Omega_\xi \times \Omega_{\hat{W}}$ is determined, if NN is constructed such that its approximation range covers the determinant compact set $\Omega_z = \Omega_\xi \times \Omega_{yd}$, then NN approximation ability always holds. It implies that given any initial condition Ω_0, with employment of an NN whose approximation range is over corresponding Ω_z, the NN control (33) guarantees the boundedness of closed-loop signals. According to Definition 3, the closed-loop signals are SGUUB.

In addition, according to Corollary 1, it can be seen that the tracking error and the NN weight estimation error are ultimately bounded as

$$\lim_{k\to\infty}\sup\left\{|e(k)|^2 + \frac{\bar{g}}{\gamma_s}\|\tilde{W}_s(k)\|^2\right\} = \lim_{k\to\infty}\sup V(k) \le \frac{\bar{b}}{1-\bar{c}}$$

where $\bar{b}$ and $\bar{c}$ are defined in Theorem 1. This completes the proof. ■

V. Transformation for Output Feedback Control

To design output feedback control, in this section, we consider transforming the system into an input–output model.

A. Transformation to Input–Output Model

Let us rewrite the first equation of (1) as

$$\xi_1(k+1) - f_1\left(\xi_1(k), \xi_2(k)\right) = 0. \tag{46}$$

Noting Assumption 1 and according to Lemma 2, $\xi_2(k)$ can be seen as a function of $\xi_1(k+1)$ and $\xi_1(k)$, i.e.,

$$\begin{aligned}\xi_2(k) &= p_2'\left(\xi_1(k+1), \xi_1(k)\right)\\ &:= p_2\left(y(k+1), y(k)\right)\end{aligned} \tag{47}$$

where $p_2'(\cdot)$ is the implicit function asserted by Lemma 2. In the same manner, from the second equation of (1), $\xi_3(k)$ can be expressed as a function of $\xi_2(k+1)$, $\xi_2(k)$, and $\xi_1(k)$ as

$$\begin{aligned}\xi_3(k) &= p_3'\left(\xi_2(k+1), \xi_2(k), \xi_1(k)\right)\\ &= p_3'\left(p_2\left(y(k+2), y(k+1)\right), p_2\left(y(k+1), y(k)\right), y(k)\right)\\ &:= p_3\left(y(k+2), y(k+1), y(k)\right)\end{aligned} \tag{48}$$

where $p_3'(\cdot)$ is the implicit function asserted by Lemma 2. Continuing the same procedure, we can see that $\xi_i(k)$, $i = 2, 3, \cdots, n$, can be expressed as

$$\begin{aligned}\xi_i(k) &= p_i'\left(\xi_{i-1}(k+1), \xi_{i-1}(k), \xi_{i-2}(k), \cdots, \xi_1(k)\right)\\ &= p_i'\left(p_{i-1}\left(y(k+i-1), \cdots, y(k+1)\right),\right.\\ &\qquad p_{i-1}\left(y(k+i-2), \cdots, y(k)\right),\\ &\qquad \left. p_{i-2}\left(y(k+i-3), \cdots, y(k)\right), \cdots, y(k)\right)\\ &:= p_i\left(y(k+i-1), y(k+i-2), \cdots, y(k)\right)\end{aligned} \tag{49}$$

where $p_i'(\cdot)$ is the implicit function asserted by Lemma 2 and $p_i(\cdot)$, $i = 2, 3, \ldots, n$, are defined consistently. Then, it is easy to derive a vector function only dependent on outputs to express $\bar{\xi}_i(k)$ as

$$\begin{aligned}\bar{\xi}_i(k) &= \begin{bmatrix}\xi_1(k)\\ \xi_2(k)\\ \vdots\\ \xi_i(k)\end{bmatrix}\\ &= \begin{bmatrix} y(k)\\ p_2\left(y(k+1), y(k)\right)\\ \vdots\\ p_i\left(y(k+i-1), y(k+i-2), \cdots, y(k)\right)\end{bmatrix}\\ &:= P_i\left(y(k+i-1), y(k+i-2), \cdots, y(k)\right),\\ &\qquad i = 1, 2, \ldots, n.\end{aligned} \tag{50}$$

Now let us rewrite the equations in system (1) as follows:

$$\begin{cases}\xi_1(k+n) = f_1\left(\bar{\xi}_1(k+n-1), \xi_2(k+n-1)\right)\\ \xi_2(k+n-1) = f_2\left(\bar{\xi}_2(k+n-2), \xi_3(k+n-2)\right)\\ \qquad\vdots\\ \xi_{n-1}(k+2) = f_{n-1}\left(\bar{\xi}_{n-1}(k+1), \xi_n(k+1)\right)\\ \xi_n(k+1) = f_n\left(\bar{\xi}_n(k), u(k), d(k)\right)\\ y(k) = \xi_1(k).\end{cases} \tag{51}$$

Then, replacing $\xi_2(k+n-1)$ in the first equation of (51) with the right-hand side of the second equation yields

$$\begin{aligned}\xi_1(k+n) &= f_1\left(\bar{\xi}_1(k+n-1), f_2\left(\bar{\xi}_2(k+n-2), \xi_3(k+n-2)\right)\right)\\ &:= \psi_{2,1}\left(\bar{\xi}_1(k+n-1), \bar{\xi}_2(k+n-2), \xi_3(k+n-2)\right).\end{aligned} \tag{52}$$

Using the chain rule of derivative, we will have

$$\frac{\partial \psi_{2,1}(\cdot)}{\partial \xi_3(k+n-2)} = g_{1,1}(\cdot) g_{1,2}(\cdot) := g_{2,1}(\cdot). \tag{53}$$

Continuing to replace $\xi_j(k+n-j+1)$ in (52) with the right-hand side of the jth equation in (51), $j = 3, 4, \ldots, n-1$, we have

$$\begin{aligned}\xi_1(k+n) &= \psi_{j-1,1}\left(\bar{\xi}_1(k+n-1), \bar{\xi}_2(k+n-2), \ldots,\right.\\ &\qquad \bar{\xi}_{j-1}(k+n-j),\\ &\qquad \left. f_j\left(\bar{\xi}_j(k+n-j), \xi_{j+1}(k+n-j)\right)\right)\\ &:= \psi_{j,1}\left(\bar{\xi}_1(k+n-1), \bar{\xi}_2(k+n-2), \ldots,\right.\\ &\qquad \left. \bar{\xi}_j(k+n-j), \xi_{j+1}(k+n-j)\right)\end{aligned} \tag{54}$$

where $\psi_{j,1}(\cdot)$, $j = 3, 4, \ldots, n-1$, are defined recursively. Similarly, we have

$$\frac{\partial \psi_{j,1}(\cdot)}{\partial \xi_{j+1}(k+n-j)} = g_{j-1,1}(\cdot) g_{1,j}(\cdot) := g_{j,1}(\cdot) \tag{55}$$

where $g_{j,1}(\cdot)$, $j = 3, 4, \ldots, n-1$, are also defined recursively. Continuing the substitution until control $u(k)$ appears on the right-hand side of (54), we have

$$\begin{aligned}y(k+n) &= \psi_{n-1,1}\left(\bar{\xi}_1(k+n-1), \bar{\xi}_2(k+n-2), \ldots,\right.\\ &\qquad \left. \bar{\xi}_{n-1}(k+1), f_n\left(\bar{\xi}_n(k), u(k), d(k)\right)\right)\\ &:= \psi_{n,1}\left(\bar{\xi}_1(k+n-1), \bar{\xi}_2(k+n-2), \ldots,\right.\\ &\qquad \left. \bar{\xi}_n(k), u(k), d(k)\right).\end{aligned} \tag{56}$$

In the same manner, we have

$$\frac{\partial \psi_{n,1}(\cdot)}{\partial u(k)} = g_{n-1,1}(\cdot) g_{1,n}(\cdot) := g_{n,1}(\cdot). \tag{57}$$

From the definition of vector functions $P_i(\cdot)$, (56) can be further written as

$$\begin{aligned}y(k+n) &= \psi_{n,1}\left(P_1\left(y(k+n-1)\right),\right.\\ &\qquad P_2\left(y(k+n-1), y(k+n-2)\right), \ldots,\\ &\qquad P_{n-1}\left(y(k+n-1), \ldots, y(k+1)\right),\\ &\qquad \left. P_n\left(y(k+n-1), \ldots, y(k)\right), u(k), d(k)\right)\\ &:= f\left(y(k+n-1), y(k+n-2), \ldots, y(k), u(k), d(k)\right).\end{aligned} \tag{58}$$

Accordingly, we have

$$\frac{\partial f(\cdot)}{\partial u(k)} = g_{n,1}(\cdot) = \Pi_{i=1}^{n} g_{1,i}(\cdot) := g_o(\cdot), \qquad \underline{g} \leq |g(\cdot)| \leq \bar{g}. \tag{59}$$

B. Future Outputs Prediction

Control design based on (58) is not straightforward due to the existence of future outputs. Hence, let us consider applying output prediction approach [18]. For convenience, we define

$$\begin{aligned}\underline{y}(k) &= \left[y(k), y(k-1), \ldots, y(k-n+1)\right]^T\\ \underline{u}(k) &= \left[u(k), \ldots, u(k-n+2)\right]^T.\end{aligned}$$

Moving back $(n-1)$ steps in (58), we obtain

$$\begin{aligned} y(k+1) &= f\,(y(k),\cdots,y(k-n+1), \\ &\qquad u(k-n+1), d(k-n+1)) \\ &:= F_1\left(\underline{y}(k), u(k-n+1), d(k-n+1)\right). \end{aligned} \quad (60)$$

It means that the output $y(k+1)$ is a function of current and past outputs $y(k),\cdots,y(k-n+1)$ and past input $u(k-n+1)$ and disturbance $d(k-n+1)$.

Moving one step forward, we obtain the following equation from (60):

$$y(k+2) = F_1\left(\underline{y}(k+1), u(k-n+2), d(k-n+2)\right). \quad (61)$$

Substituting (60) into (62), $y(k+2)$ becomes a function of $\underline{y}(k)$ and $u(k-n+2), u(k-n+1)$ and disturbance $d(k-n+1), d(k-n+2)$. Define

$$\begin{aligned} y(k+2) = F_2\big(&\underline{y}(k), u(k-n+2), u(k-n+1), \\ &d(k-n+2), d(k-n+1)\big). \end{aligned} \quad (62)$$

If we continue to substitute recursively, it is easy to prove that $y(k+n-1)$ is a function of $\underline{y}(k)$, $\underline{u}(k-1)$, and $d(k-1),\cdots,d(k-n+1)$, as expressed in the following:

$$y(k+n-1) = F_{n-1}\left(\underline{y}(k), \underline{u}(k-1), d(k-1),\cdots,d(k-n+1)\right). \quad (63)$$

Moving one step ahead in (63), we have the following:

$$y(k+n) = F_{n-1}\left(\underline{y}(k+1), \underline{u}(k), d(k),\cdots,d(k-n+2)\right). \quad (64)$$

Substituting (60) into (64), and introducing the following definition:

$$\begin{aligned} \underline{z}(k) &= \left[\underline{y}^T(k), \underline{u}^T(k-1)\right]^T \\ \underline{d}(k) &= \left[d(k), d(k-1),\ldots,d(k-n+1)\right]^T \end{aligned}$$

we can see that on the right-hand side of (64), there will be no future outputs and the control input $u(k)$ appears. Then, we see that $y(k+n)$ is a function of $\underline{z}(k)$, $u(k)$, and $\underline{d}(k)$. It is defined as

$$y(k+n) = F_n\left(\underline{z}(k), u(k), \underline{d}(k)\right). \quad (65)$$

It is easy to check that $F_n(\cdot)$ is a continuous function and it is continuously differentiable over $u(k)$. Therefore, (65) can be expressed by the mean value theorem [36] as

$$y(k+n) = \phi_o\left(\underline{z}(k), u(k)\right) + d_o(k) \quad (66)$$

where

$$\begin{aligned} \phi_o(\underline{z}(k), u(k)) &= F_n\left(\underline{z}(k), u(k), \mathbf{0}_{[n]}\right) \\ d_o(k) &= F_n(\underline{z}(k), u(k), \underline{d}(k)) - F_n\left(\underline{z}(k), u(k), \mathbf{0}_{[n]}\right). \end{aligned}$$

According to Assumption 3, there exists a constant L_m such that

$$\begin{aligned} |d_o(k)| &= \left|F_n\left(\underline{z}(k), u(k), \underline{d}(k)\right) - F_n\left(\underline{z}(k), u(k), \mathbf{0}_{[n]}\right)\right| \\ &\le L_m\,|d(k)| + L_m\,|d(k-1)| + \ldots + L_m\,|d(k-n+1)| \\ &\le nL_m\bar{d} := \bar{d}_o. \end{aligned} \quad (67)$$

VI. Output Feedback Adaptive NN Control

The dynamics of the tracking error $e(k) = y(k) - y_d(k)$ is given by

$$e(k+n) = \phi_o\left(\underline{z}(k), u(k)\right) - y_d(k+n) + d_o(k). \quad (68)$$

It is trivial to show that

$$\frac{\partial\left(\phi_o\left(\underline{z}(k), u(k)\right) - y_d(k+n)\right)}{\partial u(k)} > 0$$

therefore, there exists ideal control input $u_o^*(\bar{z}(k))$ satisfying

$$\begin{aligned} &\phi_o\left(\underline{z}(k), u_o^*\left(\bar{z}(k)\right)\right) - y_d(k+n) = 0 \\ &\bar{z}(k) = \left[\underline{z}^T(k), y_d(k+n)\right]^T \in \Omega_{\bar{z}} \subset R^{2n} \end{aligned} \quad (69)$$

where $\Omega_{\bar{z}}$ is a compact set corresponding to Ω_ξ and Ω_{yd}. Using the ideal control $u_o^*(\bar{z}(k))$, we will have $e(k) = 0$ after n steps if $d(k) = 0$. It implies that the ideal control $u_o^*(\bar{z}(k))$ is an n-step deadbeat control. According to Lemma 3, the ideal control $u_o^*(\bar{z}(k))$ is bounded.

As mentioned in Section II-B, there exist an ideal NN weights vector $W_o^* \in R^{l_o}$, such that $u_o^*(\bar{z}(k))$ can be approximated by an HONN as follows:

$$\begin{aligned} u_{nn}^*\left(\bar{z}(k)\right) &= W_o^{*T} S\left(\bar{z}(k)\right), \qquad S\left(\bar{z}(k)\right) \in R^{l_o} \\ u_o^*\left(\bar{z}(k)\right) &= u_{nn}^*\left(\bar{z}(k)\right) + \mu\left(\bar{z}(k)\right) \qquad \forall \bar{z} \in \Omega_{\bar{z}} \end{aligned} \quad (70)$$

where $\mu(\bar{z}(k))$ is the NN approximation error. Consider using an online adaptive HONN as to approximate $u_o^*(\bar{z}(k))$. Then, the output feedback adaptive NN control is given as

$$\begin{aligned} u(k) &= \frac{\eta_o(k)}{\bar{g}} e(k) + \hat{u}_o(k) \\ \hat{u}_o(k) &= \hat{W}_o^T(k) S\left(\bar{z}(k)\right) \end{aligned} \quad (71)$$

where $|\eta_o(k)| \le \bar{\eta}_o < 1$ is a scaling parameter to be specified and the NN weights vector is updated by the following adaptation law:

$$\hat{W}_o(k+1) = \hat{W}_o(k_1) - \gamma_o S\left(\bar{z}(k_1)\right) e(k+1) - \sigma_o \hat{W}_o(k_1), \quad k_1 = k-n+1 \quad (72)$$

where $0 < \sigma_o < 1$ and $\gamma_o > 0$ are NN tuning parameters to be chosen.

Theorem 2: Consider the adaptive closed-loop system consisting of the system (1), adaptive NN control (71), and NN adaptation law (72). Under Assumptions 1–3, and with design parameters $0 < \sigma_o < 1$, $0 < \bar{\eta}_o < 1$, and γ_o satisfying

$$2\gamma_o \bar{g} l_o + \bar{\eta}_o \bar{g} + \bar{\eta}_o < 1 \tag{73}$$

the closed-loop system is SGUUB stable and the tracking error and NN weight estimation error will eventually be bounded as

$$\lim_{k\to\infty} \sup \left\{ |e(k)|^2 + \frac{\bar{g}}{\gamma_o} \|\tilde{W}_o(k)\|^2 \right\} \leq \frac{\bar{b}}{1-\bar{c}}$$

where

$$\begin{aligned} \bar{b} &= \frac{\bar{g}}{\bar{\eta}_o} \mu_o^{*2} + 2\frac{\bar{g}}{\gamma_o} \sigma_o \|W_o^*\|^2 \\ \bar{c} &= \max\{\bar{\eta}_o, (1-2\sigma_o)\} \\ \mu_c^* &= \mu^* + \frac{\bar{d}_o}{\bar{g}} \end{aligned} \tag{74}$$

and μ^* is the NN approximation error bound defined in (4).

Proof: It is similar to the proof of Theorem 1 and is thus omitted. ■

Remark 6: From Theorems 1 and 2, there is a tradeoff to make the ultimate bound of the output tracking error and the NN weights vector estimate error smaller. The NN approximation error bound μ^* can be made smaller by increasing NN nodes numbers l_s and l_o. However, in order to satisfy (35) and (73), we will need to choose smaller γ_s and γ_o, which will increase the ultimate bound. The method to avoid the tradeoff and to obtain an arbitrary small ultimate bound will be the topic for future research.

VII. Simulation Results

To demonstrate the effectiveness of the proposed NN control, the following CSTR system in [29] will be used for simulation:

$$\begin{cases} \dot{x}_1 = -x_1 + D_a(1-x_1)e^{\frac{x_2}{1+\frac{x_2}{\gamma}}} \\ \dot{x}_2 = -x_2 + BD_a(1-x_1)e^{\frac{x_2}{1+\frac{x_2}{\gamma}}} - \beta(x_2 - u) + d \\ y = x_1 \end{cases} \tag{75}$$

where x_1 is the concentration and x_2 is the temperature, $B = 21.5$, $\gamma = 28.5$, $D_a = 0.036$, and $\beta = 25.2$ are scalar parameters [29], and $d = \cos(t)\cos(\xi_1)$ is unmeasured disturbance. It is noted that in system (75), the state variable x_2 appears to be nonaffine. The control objective is to make the output y track a smooth reference signal y_d, which is generated by passing a discontinuous set-point step signal r with amplitude 0.4 ± 0.2 into the following linear model [34]:

$$\frac{y_d(s)}{r(s)} = \frac{\omega_n^2}{s^2 + 2\zeta_n \omega_n s + \omega_n^2} \tag{76}$$

where the natural frequency $\omega_n = 5.0$ rad/min and the damping ration $\zeta_n = 1.0$.

Denoting $\xi_1 = x_1$ and $\xi_2 = x_2$ and by using first-order Taylor expansion, the CSTR system (75) can be approximated by a discrete-time model as

$$\begin{cases} \xi_1(k+1) = f_1(\xi_1(k), \xi_2(k)) \\ \xi_2(k+1) = f_2(\xi_1(k), \xi_2(k), u(k)) + d(k) \\ y(k) = \xi_1(k) \end{cases} \tag{77}$$

where

$$f_1(\cdot) = \xi_1(k) + \left[-\xi_1(k) + D_a(1-\xi_1(k)) e^{\frac{\xi_2(k)}{1+\frac{\xi_2(k)}{\gamma}}} \right] T$$

$$f_2(\cdot) = \xi_2(k) + \left[-\xi_2(k) + BD_a(1-\xi_1(k)) e^{\frac{\xi_2(k)}{1+\frac{x_2(k)}{\gamma}}} \right. \left. - \beta(\xi_2(k) - u(k)) \right] T$$

with sampling period $T = 0.05$ and $d(k) = 0.05\cos(0.05k)\cos(\xi_1(k))$.

For system (77), it is obvious that Assumption 1 holds. Assumptions 2 and 3 are not strictly satisfied, but it is seen in the simulation results that practically the proposed controls still work well. Consider a operation ranges $0.02 < \xi_1(k) < 0.8$ and $0 < \xi_2(k) < 5$. It is easy to check that $0.18 < g_{1,1}(\cdot) < 0.13$ and $g_{2,1}(\cdot) = 1.26$ and the partial directives $\partial f_1/\partial \xi_1$, $\partial f_2/\partial \xi_1$, and $\partial f_2/\partial \xi_2$ are upper bounded in the operation range. In this operation range, we have $\bar{g} = 0.17$ such that $g_{1,1} g_{2,1} < \bar{g}$.

It should be noted that the discretized model (77) is only used for analysis. The simulation is carried out on original system (75).

A. State Feedback Control

The structure of NN control (33) is shown in Fig. 3(a) where the gain $K = \eta_s(k)/\bar{g}$ and the NN is constructed according to (2) and (3) with $l_s = 18$ neurons. For the control parameters, they can be specified as long as criteria in (35) is satisfied. First, we choose $\sigma_s = 0.01$ and $\bar{\eta}_s = 0.05$. The gain $\eta_s(k)$ can be chosen as an arbitrary sequence satisfying $|\eta_s(k)| \leq \bar{\eta}_s$. In the simulation, it is simply chosen as $\eta_s(k) = \bar{\eta}_s$ and we choose $\gamma_s = 0.08$

The simulation is carried out with the initial states $\bar{\xi}_2(0) = [0.1, 0.1]^T$, and for the initial weights vector $\hat{W}_s(j) \in R^{l_s}$, $j = -1, 0$, each element is selected as a standard uniform distributed random number divided by 10. The results are presented in Figs. 4(a), 5(a), and 6(a). Fig. 4(a) shows the output $y(k)$ and the reference signal $y_d(k)$. Fig. 5(a) illustrates the boundedness of the control input $u(k)$ and the norm of NN weights vector $\|\hat{W}_s(k)\|$ and Fig. 6(a) shows state $\xi_2(k)$. It can be seen that all the signals are bounded in the operation range.

B. Output Feedback Control

The structure of NN control (71) is shown in Fig. 3(b) where the gain $K = \eta_o(k)/\bar{g}$ and the NN is constructed according to (2) and (3) with $l_s = 30$ neurons. The initial system states are $\bar{\xi}_2(0) = [0.1, 0.1]^T$. The initial weight estimate $\hat{W}_o(0)$, $j = -1, 0$, is chosen in the same manner as that for state feedback control design. The design parameters are chosen as $\bar{\eta}_o(k) =$

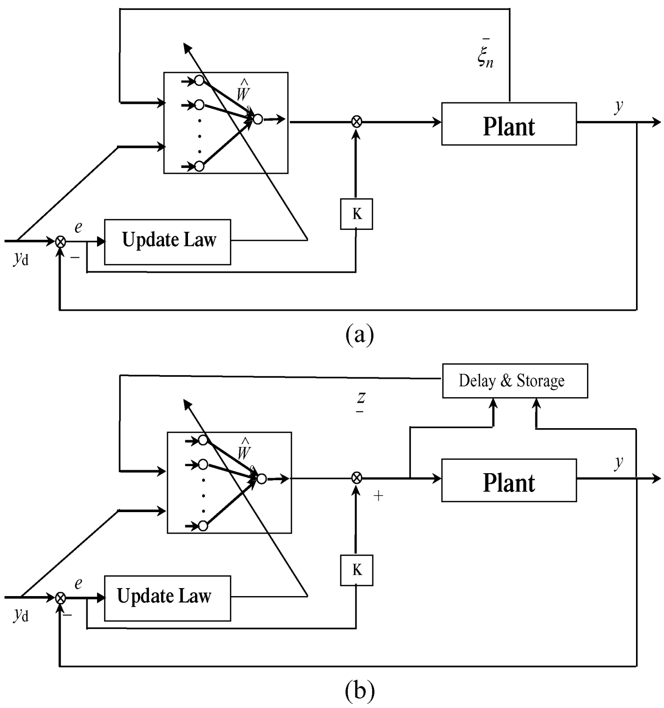


Fig. 3. NN control structure. (a) State feedback control structure; (b) output feedback control structure.

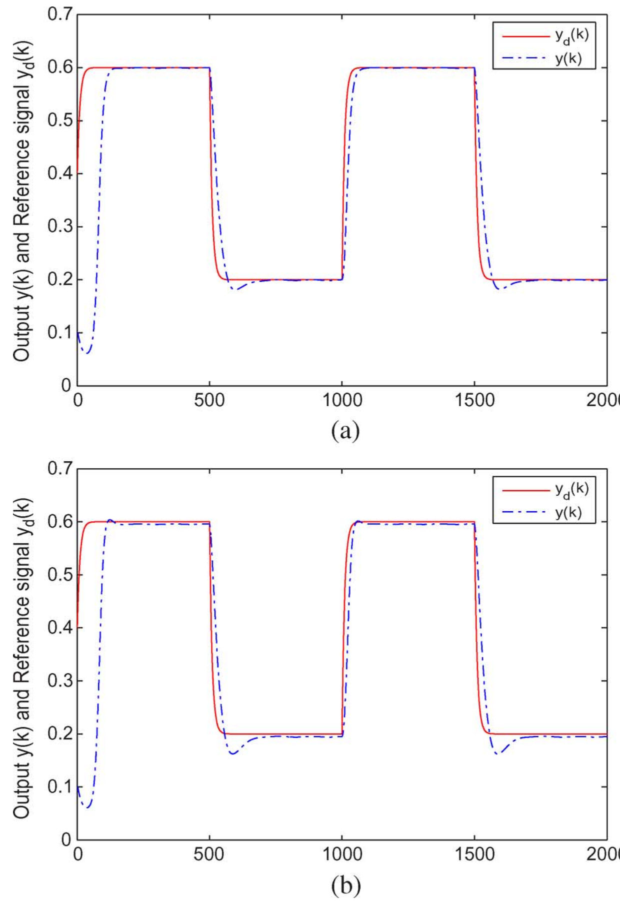


Fig. 4. System output and reference trajectory. (a) State feedback NN control; (b) output feedback NN control.

0.05, $\gamma_o = 0.06$, and $\sigma_o = 0.01$, which satisfy the criterion in (73) The simulation results are presented in Figs. 4(b), 5(b), and 6(b). Fig. 4(b) shows the output $y(k)$ and the reference signal $y_d(k)$. Fig. 5(b) illustrates the boundedness of the control input $u(k)$ and the norm of NN weight $\|\hat{W}_o(k)\|$, and Fig. 6(b) shows state $\xi_2(k)$.

C. NN Learning Performance

To demonstrate the NN learning performance, we define the following mean square errors (MSEs):

$$e_s(k) = \frac{1}{k}\sum_{k'=1}^{k}\left[\phi_s\left(\bar{\xi}_n(k'), \hat{u}_s\left(z(k')\right)\right) - y_d(k'+n)\right]^2$$
$$e_o(k) = \frac{1}{k}\sum_{k'=1}^{k}\left[\phi_o\left(\underline{z}(k'), \hat{u}_o\left(\bar{z}(k')\right)\right) - y_d(k'+n)\right]^2 \quad (78)$$

as measurement of NN learning error. According to (31) and (69), the smaller the NN approximation error $\hat{u}_s(k) - u_s^*(k)$ and $\hat{u}_o(k) - u_o^*(k)$ are, the smaller $e_s(k)$ and $e_o(k)$ are. If $\hat{u}_s(k) - u_s^*(k) = 0$ and $\hat{u}_o(k) - u_o^*(k) = 0$, we have $e_s(k) = 0$ and $e_o(k) = 0$.

The MSEs of state feedback and output feedback NN learning are demonstrated in Fig. 7(a) and (b). It is noted that the NN learning performance is satisfactory, i.e., the defined MSEs $e_s(k)$ and $e_o(k)$ are made to be bounded around zero.

D. Discussion and Comparison

From the previously presented simulation results, it is seen that all the signals are bounded in the operation range. According to Fig. 4, in the initial period of simulation, the system output does not track the reference trajectory very well. However, as the simulation time increases, the output tracking becomes much better. This is because the initial NN weights are set to be zero. Thus, NN has to be sufficiently trained before it is able to generate NN control outputs that would facilitate a good trajectory following.

It is seen from Figs. 4 and 7 that the steady-state error is very small and after the first rising of reference trajectory, the output tracks the rising and falling of reference trajectory quickly enough. The tracking performance is better at the rising edge of reference signal (no overshoot and shorter rising time) than at the falling edge (small overshoot and a bit longer rising time). The asymmetric performance is due to complicated nonlinearity of the plant.

To demonstrate the superiority over proportional–integral–derivative (PID) control, we compare the proposed output feedback NN control (71) with a standard PID control. In the simulation, the system initial condition is set to be $\bar{\xi}_2(0) = [0.1, 0.1]^T$ and the PID control is given in discretized manner as

$$u(k) = u(k-1) + K_p\left[e(k) - e(k-1)\right]$$
$$+K_i e(k) + K_d\left[e(k) - 2e(k-1) + e(k-2)\right]$$

where the parameters $K_P = 4$, $K_I = -0.2$, and $K_D = 1$ were found by trial and error to minimize the sum of squared output tracking errors.

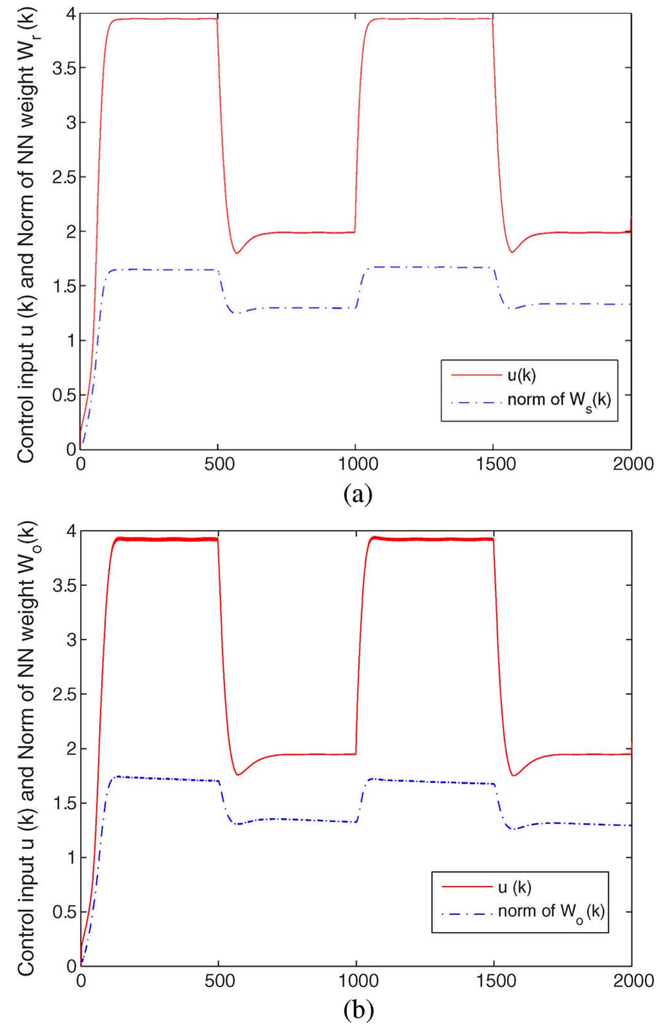


Fig. 5. Boundedness of control signal and NN weight. (a) State feedback NN control; (b) output feedback NN control.

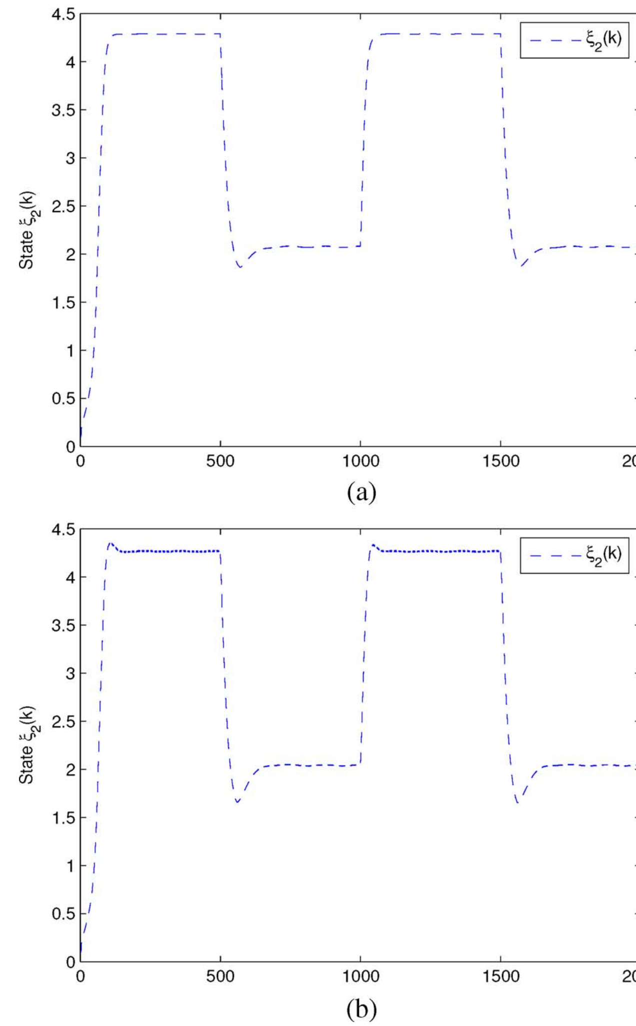


Fig. 6. State ξ_2. (a) State feedback NN control; (b) output feedback NN control.

The proposed output feedback adaptive NN control is further compared with the linear error observer-based NN inverse control constructed in [10], which is a continuous-time design for nonaffine system. The system's initial condition is also set to be $\bar{\xi}_2(0) = [0.1, 0.1]^T$. The dynamic compensator parameters used in the control are set to be $A_c = -0.86$, $B_c = -1.4$, $C_c = 0.1$, and $D_c = -0.75$. HONN with 45 neurons is used with the same initial condition as that for our proposed output feedback control. The design parameters are $\gamma_W = 35$, $Q_2 = I$, $\lambda_W = 0.01$, $\lambda_\Phi = 0.01$, and $\gamma_\Phi = 0.001$. The poles of the observer have been set to be five times faster than those of the closed-loop error system.

The comparison results are shown in Fig. 8, where it is very clear that the two NN-based controls perform much better than the PID control with respect to either tracking error or control effort, though NN-based controls respond not as quickly as PID control in the initial steps. This is because the two NN controls are based on online NN learning. From the tracking performance of the two NN-based controls in Fig. 8(a), it is seen that the inverse NN control has an obvious steady-state error while the steady-state error for our proposed output-feedback adaptive NN control is very small.

VIII. Conclusion

In this paper, we have studied adaptive NN control of nonlinear discrete-time systems in nonaffine pure-feedback form. By future state predictions, the system is transformed to an n-step predictor for state feedback control, and by future output predictions, the system is further transformed into a suitable input–output model for output feedback control. Implicit function theorem is exploited to identify the existence of an ideal control and NN is used to approximate the unknown function in the control. Both state feedback and output feedback controls only use one single NN and achieve SGUUB stability in the closed loop. The results in the paper can be further extended to other linearly parametrized approximator that also has the property in (5), such as radius basis function (RBF) NN.

Appendix I
Proof of Lemma 3

Proof: Consider the first equation in (1). According to the mean value theorem, it can be written as

$$\begin{aligned} y(k+1) &= f_1\left(\xi_1(k), \xi_2(k)\right) \\ &= f_1\left(y(k), 0\right) + g_{1,1}\left(y(k), \xi_2^c(k)\right)\xi_2(k) \end{aligned}$$

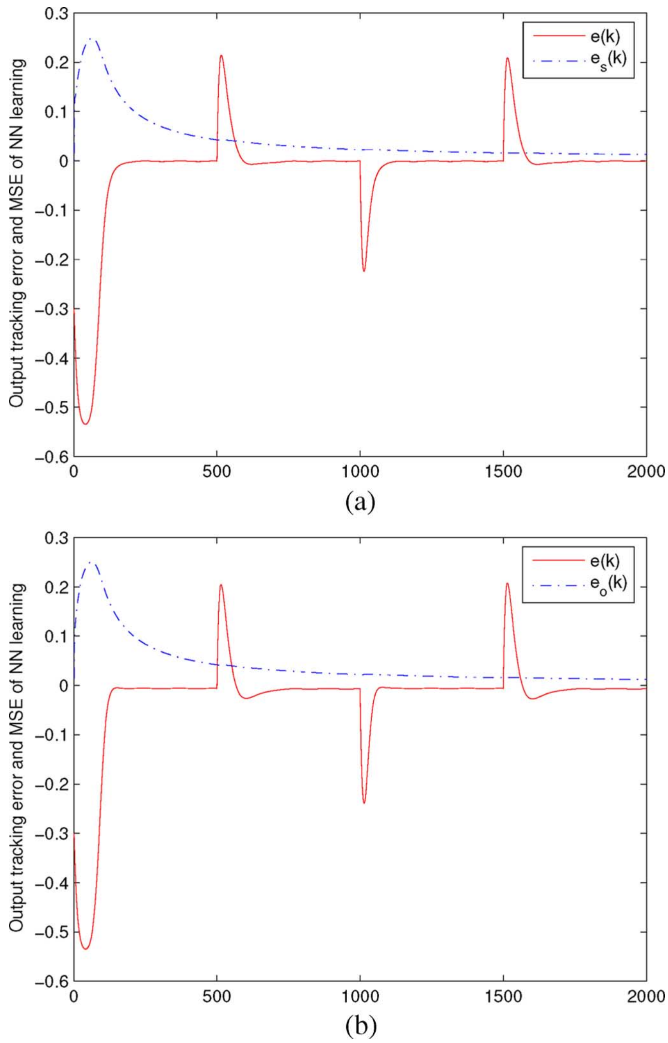


Fig. 7. Output tracking error and MSE of NN learning. (a) State feedback NN control; (b) output feedback NN control.

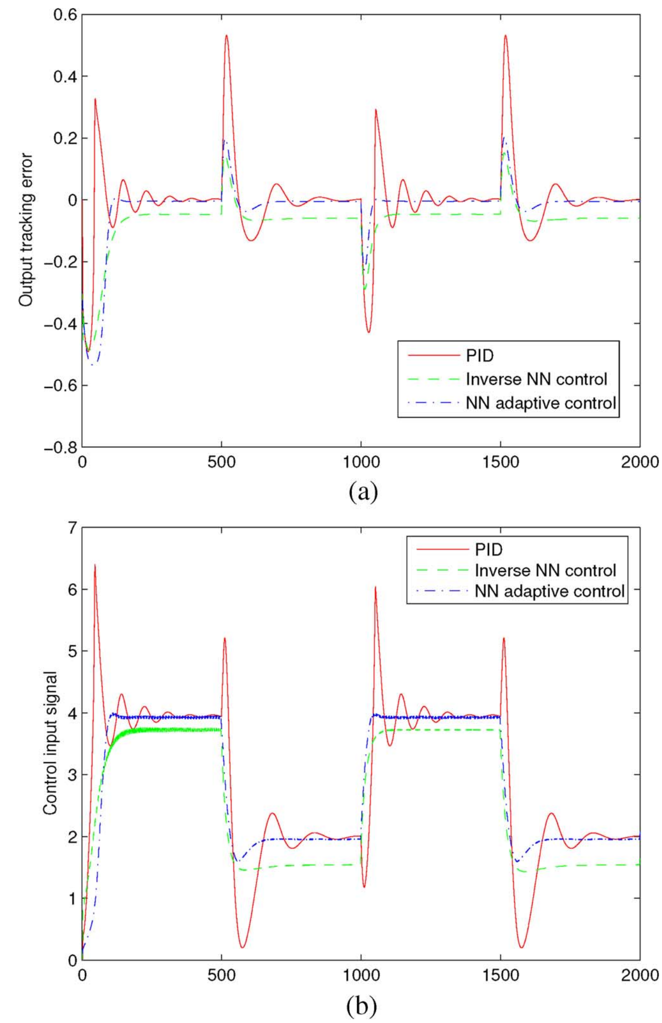


Fig. 8. Comparison of PID, NN inverse, and adaptive NN control. (a) Comparison of tracking errors; (b) comparison of control signals.

where $\xi_2^c(k) \in [\min\{0, \xi_2(k)\}, \max\{0, \xi_2(k)\}]$. According to Assumptions 2 and 3, we obtain

$$\begin{aligned}|\xi_2(k)| &= \left|\frac{y(k+1) - f_1(y(k), 0)}{g_{1,1}(y(k), \xi_2^c(k))}\right| \\ &\leq \frac{1}{\underline{g}_1}\left(|y(k+1)| + L_1|y(k)| + C_{1,0}\right) \\ &\leq C_{1,1} \max_{k \leq i \leq k+1}\{|y(i)|\} + C_{1,2}\end{aligned} \tag{79}$$

where L_1 is the Lipschitz constant, $C_{1,0} = |f_1(0,0)|$, $C_{1,1} = (1/\underline{g}_1)(1 + L_1)$, and $C_{1,2} = (1/\underline{g}_1)C_{1,0}$. Similarly, the second equation in (1) can be written as

$$\begin{aligned}\xi_2(k+1) &= f_2(y(k), \xi_2(k), \xi_3(k)) \\ &= f_2(y(k), \xi_2(k), 0) + g_{1,2}(y(k), \xi_2(k), \xi_3^c(k))\xi_3(k)\end{aligned}$$

where $\xi_3^c(k) \in [\min\{0, \xi_3(k)\}, \max\{0, \xi_3(k)\}]$. In the same manner, it can be shown that there are some finite constants L_2 and $C_{2,0}$ such that

$$\begin{aligned}|\xi_3(k)| &= \left|\frac{\xi_2(k+1) - f_2(y(k), \xi_2(k), 0)}{g_{1,2}(y(k), \xi_2(k), \xi_3^c(k))}\right| \\ &\leq \frac{1}{\underline{g}_2}\left(|\xi_2(k+1)| + L_2|y(k)| + L_2|\xi_2(k)| + C_{2,0}\right)\end{aligned} \tag{80}$$

where $C_{2,0} = |f_2(0,0,0)|$. Substituting (79) into (80) yields

$$|\xi_3(k)| \leq C_{2,1} \max_{k \leq i \leq k+2}\{|y(i)|\} + C_{2,2} \tag{81}$$

where $C_{2,1}$ and $C_{2,2}$ are some finite constants.

Following the previous procedure, one can easily show that

$$|\xi_j(k)| \leq C_{j-1,1} \max_{k \leq i \leq k+j-1}\{|y(i)|\} + C_{j-1,2}, \quad j = 2, 3 \ldots, n \tag{82}$$

where $C_{j,1}$ and $C_{j,2}$ are some finite constants. Then, it is easy to obtain

$$\|\bar{\xi}_n(k)\| \leq \sum_{i=1}^{n}|\xi_i(k)| \leq C_1 \max_{k \leq i \leq k+n-1}\{|y(i)|\} + C_2 \tag{83}$$

where

$$C_1 = 1 + \sum_{j=2}^{n} C_{j-1,1} \quad C_2 = \sum_{j=2}^{n} C_{j-1,2}. \tag{84}$$

From the last equation in (1), one has

$$\begin{aligned} |u(k)| &= \left| \frac{\xi_n(k+1) - f_n\left(\bar{\xi}_n(k), 0, d(k)\right)}{g_{1,n}\left(\bar{\xi}_n(k), u^c(k), d(k)\right)} \right| \\ &\leq \frac{1}{\underline{g}_n}\left(|\xi_n(k+1)| + L_n\left|\bar{\xi}_n(k)\right| + L_n\bar{d} + C_{n,0}\right) \end{aligned} \tag{85}$$

where $u^c(k) \in [\min\{0, u(k)\}, \max\{0, u(k)\}]$ and L_n and $C_{n,0}$ are some finite constants. Combining with (82) and (83), we have

$$|u(k)| \leq C_3 \max_{k \leq i \leq k+n} \{|y(i)|\} + C_4 \tag{86}$$

where C_3 and C_4 are some finite constants. ■

Appendix II
Proof of Lemma 4

Proof: First, let us consider the following inequality of $V(k) \geq 0$:

$$V(k+1) \leq c(k)V(k) + b(k), \qquad k \in Z_0^+ \tag{87}$$

where $|c(k)| \leq \bar{c} < 1$ and $|b(k)| \leq \bar{b}$. It is straightforward to show that

$$\begin{aligned} V(1) &\leq \bar{c}V(0) + \bar{b} \\ V(2) &\leq \bar{c}V(1) + \bar{b} \leq \bar{c}^2 V(0) + (\bar{c}+1)b \\ &\vdots \\ V(k) &\leq \bar{c}^k V(0) + \frac{1-\bar{c}^k}{1-\bar{c}}\bar{b} \leq V(0) + \frac{\bar{b}}{1-\bar{c}} \end{aligned}$$

and, furthermore

$$\lim_{k\to\infty} \sup\{V(k)\} \leq \lim_{k\to\infty} \bar{c}^k V(0) + \lim_{k\to\infty} \frac{1-\bar{c}^k}{1-\bar{c}}\bar{b} = \frac{\bar{b}}{1-\bar{c}}.$$

Now, if we choose $c(k) = \max\{c_i(k)\}$, $i = 1, 2, \ldots, m$, then the inequality (7) becomes (87) in Lemma 4. It is easy to see that (8) holds. ■

Appendix III
Proof of Corollary 1

Proof: Define $V_i^j(l) = V_i(ln + j)$ and $V^j(l) = \sum_{i=1}^{m} V_i^j(l)$, where $l \in Z_0^+$, $i = 1, 2, \ldots, m$, $j = 0, 1, \ldots, n-1$. It is obvious that $V^j(0) \leq \bar{V}(0)$. Then, from the definition, we have

$$\begin{aligned} V^j(l+1) &= \sum_{i=1}^{m} V_i^j(l+1) = \sum_{i=1}^{m} V_i\left((l+1)n + j\right) \\ &= V(ln + n + j). \end{aligned} \tag{88}$$

According to (9), it is easy to obtain

$$\begin{aligned} V(ln+n+j) &\leq \sum_{i=1}^{m} c_i(ln+j)V_i(ln+j) + b(ln+j) \\ &= \sum_{i=1}^{m} c_i^j(l)V_i^j(l) + b^j(l) \end{aligned} \tag{89}$$

where $c_i^j(l) = c_i(ln+j)$ and $b^j(l) = b(ln+j)$. Combining (88) and (89) results in

$$V^j(l+1) \leq \sum_{i=1}^{m} c_i^j(l)V_i^j(l) + b^j(l). \tag{90}$$

Noting that $|c_i^j(l)| \leq \bar{c}$ and $|b^j(l)| \leq \bar{b}$, we apply Lemma 4 to (90) and it results in

$$\begin{aligned} V^j(l) &\leq V^j(0) + \frac{\bar{b}}{1-\bar{c}} \\ &\leq \bar{V}(0) + \frac{\bar{b}}{1-\bar{c}} \qquad \forall l \in Z_0^+ \\ \lim_{l\to\infty} \sup\left\{V^j(l)\right\} &\leq \frac{\bar{b}}{1-\bar{c}}. \end{aligned} \tag{91}$$

It is obvious that $\forall k$, $k \geq n-1$, there exist $j = k(\mathrm{mod}\ n)$, $j \in \{0, 1, \ldots, n-1\}$, and $l = (k-j)/n$, such that we can obtain

$$\begin{aligned} V(k) &= \sum_{i=1}^{m} V_i(ln+j) = \sum_{i=1}^{m} V_i^j(l) \\ &= V^j(l) \leq \bar{V}(0) + \frac{\bar{b}}{1-\bar{c}}, \qquad k \geq n-1 \\ \lim_{k\to\infty} \sup\{V(k)\} &\leq \frac{\bar{b}}{1-\bar{c}}. \end{aligned} \tag{92}$$

■

Appendix IV
Proof of Lemma 5

Proof: Note that $\max_{0\leq i\leq n-1}\{V(i)\} \leq C_0$. From Corollary 1, we have the following:

$$V(k) \leq C_0 + \frac{\bar{b}}{1-\bar{c}} \quad \lim_{k\to\infty} \sup\{V(k)\} \leq \frac{\bar{b}}{1-\bar{c}}. \tag{93}$$

From the definition of $V(k)$, we have

$$\begin{aligned} e^2(k) &\leq \frac{1}{a_e}V(k) \\ \tilde{W}^T(k)\tilde{W}(k) &\leq \frac{1}{a_W}V(k). \end{aligned} \tag{94}$$

Combining (93) and (94), the following is obtained:

$$\begin{aligned} |e(k)| &\leq \sqrt{\frac{1}{a_e}\left(C_0 + \frac{\bar{b}}{1-\bar{c}}\right)} := c_{e\max} \\ \lim_{k\to\infty} \sup|e(k)| &\leq \sqrt{\frac{\bar{b}}{a_e(1-\bar{c})}} := c_{es} \\ \|\tilde{W}(k)\| &\leq \sqrt{\frac{1}{a_W}\left(C_0 + \frac{\bar{b}}{1-\bar{c}}\right)} := c_{\tilde{W}\max} \\ \lim_{k\to\infty} \sup\|\tilde{W}(k)\| &\leq \sqrt{\frac{\bar{b}}{a_W(1-\bar{c})}} := c_{\tilde{W}s}. \end{aligned} \tag{95}$$

Then, it is easy to show that

$$\begin{aligned}\left\|\bar{\xi}_n(k)\right\| &\leq C_1 \max_{k\leq i\leq k+n-1}\{|y(i)|\} + C_2\\ &\leq C_1 \sup_{y_d\in\Omega_{yd}}\{|y_d(k)|\} + C_1 c_{e\max} + C_2\\ \|\hat{W}(k)\| &\leq \|W^*\| + \|\tilde{W}(k)\| \leq \|W^*\| + c_{\tilde{W}}\end{aligned}$$

and

$$\begin{aligned}\lim_{k\to\infty}\sup\|\bar{\xi}_n(k)\| &\leq C_1 \sup_{y_d\in\Omega_{yd}}\{|y_d(k)|\} + C_1 c_{es} + C_2\\ \lim_{k\to\infty}\sup\|\hat{W}(k)\| &\leq \|W^*\| + \|\tilde{W}(k)\| \leq \|W^*\| + c_{\tilde{W}s}.\end{aligned}$$

■

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for constructive comments that helped to improve the quality and presentation of this paper.

REFERENCES

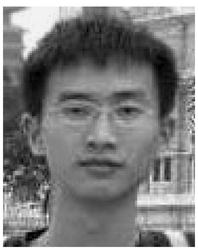
[1] K. S. Narendra and K. Parthasarathy, "Identification and control of dynamic systems using neural networks," *IEEE Trans. Neural Networks*, vol. 1, no. 1, pp. 4–27, Mar. 1990.
[2] A. U. Levin and K. S. Narendra, "Control of nonlinear dynamical systems using neural networks—Part II: Observability, identification, and control," *IEEE Trans. Neural Netw.*, vol. 7, no. 1, pp. 30–42, Jan. 1996.
[3] F. L. Lewis, S. Jagannathan, and A. Yesildirek, *Neural Network Control of Robot Manipulators and Nonlinear Systems*. London, U.K.: Taylor & Francis, 1999.
[4] S. S. Ge, C. C. Hang, T. H. Lee, and T. Zhang, *Stable Adaptive Neural Network Control*. Norwell, MA: Kluwer, 2001.
[5] M. M. Polycarpou and M. J. Mears, "Stable adaptive tracking of uncertain systems using nonlinearly parametrized on-line approximators," *Int. J. Control*, vol. 70, no. 3, pp. 363–384, 1998.
[6] A. Yesidirek and F. L. Lewis, "Feedback linearization using neural networks," *Automatica*, vol. 31, no. 11, pp. 1659–1664, 1995.
[7] Y. Zhang, P. Y. Peng, and Z. P. Jiang, "Stable neural controller design for unknown nonlinear systems using backstepping," *IEEE Trans. Neural Netw.*, vol. 11, no. 6, pp. 1347–1360, Nov. 2000.
[8] S. S. Ge and C. Wang, "Adaptive neural control of uncertain MIMO nonlinear systems," *IEEE Trans. Neural Netw.*, vol. 15, no. 3, pp. 674–692, May 2004.
[9] S. S. Ge, C. C. Hang, and T. Zhang, "Adaptive neural network control of nonlinear systems by state and output feedback," *IEEE Trans. Syst. Man Cybern. B, Cybern.*, vol. 29, no. 6, pp. 818–828, Dec. 1999.
[10] N. Hovakimyan, F. Nardi, and A. J. Calise, "A novel error observer-based adaptive output feedback approach for control of uncertain systems," *IEEE Trans. Autom. Control*, vol. 47, no. 8, pp. 1310–1314, Aug. 2002.
[11] S. S. Ge and C. Wang, "Adaptive NN control of uncertain nonlinear pure-feedback systems," *Automatica*, vol. 38, no. 4, pp. 671–682, 2002.
[12] C. Wang, D. J. Hill, S. S. Ge, and G. Chen, "An ISS-modular approach for adaptive neural control of pure-feedback systems," *Automatica*, vol. 42, no. 5, pp. 723–731, 2006.
[13] J. B. D. Cabrera and K. S. Narendra, "Issues in the application of neural networks for tracking based on inverse control," *IEEE Trans. Autom. Control*, vol. 44, no. 11, pp. 2007–2027, Nov. 1999.
[14] Y. Song and J. W. Grizzle, "Adaptive output-feedback control of a class of discrete-time nonlinear systems," in *Proc. Amer. Control Conf.*, 1993, pp. 1359–1364.
[15] F. C. Chen and H. K. Khalil, "Adaptive control of a class of nonlinear discrete-time systems using neural networks," *IEEE Trans. Autom. Control*, vol. 40, no. 5, pp. 791–801, May 1995.
[16] S. Jagannathan and F. L. Lewis, "Discrete-time neural net controller for a class of nonlinear dynamical systems," *IEEE Trans. Autom. Control*, vol. 41, no. 11, pp. 1693–1699, Nov. 1996.
[17] S. S. Ge, G. Y. Li, and T. H. Lee, "Adaptive NN control for a class of strick-feedback discrete-time nonlinear systems," *Automatica*, vol. 39, no. 5, pp. 807–819, 2003.
[18] S. S. Ge, T. H. Lee, G. Y. Li, and J. Zhang, "Adaptive NN control for a class of discrete-time non-linear systems," *Int. J. Control*, vol. 76, no. 4, pp. 334–354, 2003.
[19] P. C. Yeh and P. V. Kokotovic, "Adaptive control of a class of nonlinear discrete-time systems," *Int. J. Control*, vol. 2, no. 62, pp. 303–324, 1995.
[20] C. J. Goh and T. H. Lee, "Direct adaptive control of nonlinear systems via implicit function emulation," *Control Theory Adv. Technol.*, vol. 10, no. 3, pp. 539–552, 1994.
[21] S. S. Ge, J. Zhang, and T. H. Lee, "Adaptive MNN control for a class of non-affine NARMAX systems with disturbances," *Syst. Control Lett.*, vol. 53, no. 1, pp. 1–12, 2004.
[22] O. Adetona, E. Garcia, and L. H. Keel, "A new method for the control of discrete nonlinear dynamic systems using neural networks," *IEEE Trans. Neural Netw.*, vol. 11, no. 1, pp. 102–112, Jan. 2000.
[23] Q. M. Zhu and L. Guo, "Stable adaptive neurocontrol for nonlinear discrete-time systems," *IEEE Trans. Neural Netw.*, vol. 15, no. 3, pp. 653–662, May 2004.
[24] Y. Fu and T. Chai, "Nonlinear multivariable adaptive control using multiple models and neural networks," *Automatica*, vol. 43, no. 6, pp. 1101–1110, 2007.
[25] I. Kanellakopoulos, P. V. Kokotovic, and R. Marino, "An extended direct scheme for robust adaptive nonlinear control," *Automatica*, vol. 27, no. 2, pp. 247–255, 1991.
[26] N. K. Poulsen, B. Kouvaritakis, and M. Cannon, "Nonlinear constrained predictive control applied to a coupled-tanks apparatus," *Proc. Inst. Electr. Eng., Control Theory Appl.*, vol. 148, no. 1, pp. 17–24, 2001.
[27] L. R. Hunt and G. Meyer, "Stable inversion for nonlinear systems," *Automatica*, vol. 40, no. 1, pp. 1549–1554, 1997.
[28] G. C. X. Dong and L. Chen, "Adaptive control of the uncertain duffing oscillator," *Int. J. Bifurc. Chaos*, vol. 7, no. 7, pp. 1651–1658, 1997.
[29] T. D. Knapp, H. M. Budman, and G. Broderick, "Adaptive control of a CSTR with a neural network model," *J. Process Control*, vol. 11, pp. 53–68, 2001.
[30] A. R. Barron, "Universal approximation bounds for superposition for a sigmoidal function," *IEEE Trans. Inf. Theory*, vol. 39, no. 3, pp. 930–945, May 1993.
[31] T. P. Chen and H. Chen, "Approximation capability to functions of several variables, nonlinear functionals, and operators by radial basis function neural networks," *IEEE Trans. Neural Netw.*, vol. 6, no. 4, pp. 904–910, Jul. 1995.
[32] V. Cherkassky, D. Ghering, and F. Mulier, "Comparison of adaptive methods for function estimation from samples," *IEEE Trans. Neural Netw.*, vol. 7, no. 4, pp. 969–984, Jul. 1996.
[33] M. M. Gupta and D. H. Rao, *Neuro-Control Systems: Theory and Applications*. New York: IEEE Press, 1994.
[34] S. S. Ge, J. Zhang, and T. H. Lee, "Adaptive neural network control for a class of MIMO nonlinear systems with disturbances in discrete time," *IEEE Trans. Syst. Man Cybern. B, Cybern.*, vol. 34, no. 4, pp. 1630–1645, Aug. 2004.
[35] J. R. Munkres, *Analysis on Manifolds*. Reading, MA: Addison-Wesley, 1991.
[36] T. M. Apostol, *Mathematical Analysis*. Reading, MA: Addison-Wesley, 1974.

Shuzhi Sam Ge (S'90–M'92–SM'00–F'06) received the B.Sc. degree in control engineering from Beijing University of Aeronautics and Astronautics (BUAA), Beijing, China, in 1986 and the Ph.D. degree in robotics and the Diploma of Imperial College (DIC) from Imperial College of Science, Technology and Medicine, London, U.K., in 1993.

He is Founding Director of Social Robotics Lab, Interactive Digital Media Institute and Full Professor at the Department of Electrical and Computer Engineering, National University of Singapore, Singapore. He has (co)authored three books *Adaptive Neural Network Control of Robotic Manipulators* (Singapore: World Scientific, 1998), *Stable Adaptive Neural Network Control* (Norwell, MA: Kluwer, 2001), and *Switched Linear Systems: Control and Design* (New York: Springer-Verlag, 2005); edited a book *Autonomous Mobile Robots: Sensing, Control, Decision Making and Applications* (New York: Taylor & Francis, 2006); and published over 300 international journal and conference papers. His current research interests

include social robotics, multimedia fusion, adaptive control, and intelligent systems.

Dr. Ge has served/been serving as an Associate Editor for a number of flagship journals including the IEEE TRANSACTIONS ON AUTOMATIC CONTROL, the IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY, the IEEE TRANSACTIONS ON NEURAL NETWORKS, and *Automatica*. He also serves as an Editor of the Taylor & Francis Automation and Control Engineering Series. He is an elected member of Board of Governors, IEEE Control Systems Society. He provides technical consultation to industrial and government agencies. He is the Editor-in-Chief of the *International Journal of Social Robotics*.



Chenguang Yang (S'07) received the B.E. degree in measurement and control technologies and devices from College of Automation, Northwestern Polytechnical University, Xi'an, China, in 2005. Currently, he is working towards the Ph.D. degree at the Department of Electrical and Computer Engineering, the National University of Singapore, Singapore.

His current research interests lie in adaptive control, intelligent control, and control applications.

Tong Heng Lee (M'90) received the B.A. degree (with first class honors) in engineering tripos from Cambridge University, Cambridge, U.K., in 1980 and the Ph.D. degree in electrical engineering from Yale University, New Haven, CT, in 1987.

Currently, he is the Professor at the Department of Electrical and Computer Engineering, National University of Singapore, Singapore, where he is also Head of the Drives, Power and Control Systems Group. He has also coauthored three research monographs and holds four patents (two of which are in the technology area of adaptive systems, and the other two are in the area of intelligent mechatronics). His research interests are in the areas of adaptive systems, knowledge-based control, intelligent mechatronics, and computational intelligence.

Dr. Lee is an Associate Editor of *Control Engineering Practice*, the *International Journal of Systems Science*, the *Mechatronics Journal*, and the IEEE TRANSACTIONS IN SYSTEMS, MAN AND CYBERNETICS. He received the Cambridge University Charles Baker Prize in engineering.