Neural Adaptive Backstepping Control of a Robotic Manipulator With Prescribed Performance Constraint

Qing Guo, *Member, IEEE*, Yi Zhang, *Member, IEEE*, Branko G. Celler, *Fellow, IEEE*, and Steven W. Su, *Senior Member, IEEE*

***Abstract*— This paper presents an adaptive neural network (NN) control of a two-degree-of-freedom manipulator driven by an electrohydraulic actuator. To restrict the system output in a prescribed performance constraint, a weighted performance function is designed to guarantee the dynamic and steady tracking errors of joint angle in a required accuracy. Then, a radial-basis-function NN is constructed to train the unknown model dynamics of a manipulator by traditional backstepping control (TBC) and obtain the preliminary estimated model, which can replace the preknown dynamics in the backstepping iteration. Furthermore, an adaptive estimation law is adopted to self-tune every trained-node weight, and the estimated model is online optimized to enhance the robustness of the NN controller. The effectiveness of the proposed control is verified by comparative simulation and experimental results with Proportional–integral-derivative and TBC methods.**

***Index Terms*— Adaptive estimation law, adaptive neural network (NN) control, prescribed performance constraint (PPC), two-degree-of-freedom (Two-DOF), manipulator, weighted performance function.**

I. Introduction

A MANIPULATOR is a typical robotic plant that is widely used in industrial, aeronautics, and astronautics engineering, as it has a superior load-lifting capability and can replace the human operation in high-risk environment. Generally, a manipulator has two types of driven modes, i.e., full-actuated joints [1], [2] and underactuated joints [3]. The latter is more difficult than the former, since both kinematic and dynamic constraints are integrated in the underactuated motion system, which cannot directly adopt conventional Euler–Lagrange method. Thus, model complexity [4] and uncertainty degrade the controller robustness. To handle model constraint and system uncertainty, neural network (NN) control is popular to be used in discrete nonlinear systems [5], hypersonic flight vehicle [6], mobile manipulators [7], and fixed-base manipulators [8]. Karakaşoğlu *et al.* [9] originated a supervised learning scheme that employs a simple distributed updating rule for the online identification and decentralized adaptive control. Subsequently, Man *et al.* [10] proposed a robust adaptive sliding mode control with a radial-basis-function neural network (RBFNN) for rigid robotic manipulators to achieve the robustness and asymptotic error convergence. Yang *et al.* [11] investigated the teleoperated robot systems and presented many novel control techniques, such as RBFNN, wave variable, and variable gain control to eliminate the negative effects caused by communication delays and dynamics uncertainties existing in robot systems and human operators. Cao *et al.* [12] presented an approximate optimal control integrating with NN to realize the path control of underactuated snake robots. Chen [13] proposed a robust adaptive control based on a dynamic neural-fuzzy structure in a fixed-base manipulator to avoid the problems of overfitting and underfitting existed in the trained network. Subsequently, Wai and Muthusamy [14] presented a fuzzy-NN controller with backstepping and sliding mode to improve the position tracking performance of a two-link robotic manipulator driven by a dc servomotor. Patiño *et al.* [15] proposed a feedback adaptive neurocontroller for PUMA-560 robot, which combines feedforward NNs with adaptive and robust control techniques. The advantage of this neurocontroller is that the parameter adaptation is faster than that in the case where the learning capability of the full NN is used for the adaptation task. Yue *et al.* [16] presented an NN with terminal sliding mode control used in wheeled mobile robots. In this reference, the uncertain ground friction model is identified according to the required performance. Then, He *et al.* [17], [18] proposed an adaptive NN control to estimate the unknown modeling uncertainty and environmental

Manuscript received September 8, 2017; revised January 18, 2018 and April 16, 2018; accepted June 27, 2018. Date of publication August 30, 2018; date of current version December 10, 2019. This work was supported in part by the National Natural Science Foundation of China under Grant 61801094, Grant 61522105, Grant 51775089, Grant 61305092, and Grant 51205045, in part by the Fundamental Research Funds for the Central Universities, China, under Grant ZYGX2015J118 and Grant ZYGX2016J160, in part by the Open Foundation of the State Key Laboratory of Fluid Power and Mechatronic Systems under Grant GZKF-201515, and in part by the China Post-Doctoral Science Foundation funded project under Grant 2017M612950. *(Qing Guo and Yi Zhang contributed equally to this work.) (Corresponding author: Yi Zhang.)*

Q. Guo is with the School of Aeronautics and Astronautics, University of Electronic Science and Technology of China, Chengdu 611731, China, and also with the State Key Laboratory of Fluid Power and Mechatronic Systems, Zhejiang University, Hangzhou 310027, China (e-mail: guoqinguestc@uestc.edu.cn).

Y. Zhang is with the Key Laboratory for NeuroInformation, Ministry of Education, School of Life Science and Technology, the Center for Information in BioMedicine, and the School of Aeronautics and Astronautics, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: yi.zhang@uestc.edu.cn).

B. G. Celler is with the School of Electrical Engineering and Telecommunications, University of New South Wales, Sydney, NSW 2052, Australia (e-mail: b.celler@unsw.edu.au).

S. W. Su is with the Faculty of Engineering and Information Technology, University of Technology Sydney, Ultimo, NSW 2007, Australia (e-mail: steven.su@uts.edu.au).

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

Digital Object Identifier 10.1109/TNNLS.2018.2854699

disturbance. Chen *et al.* [19] presented an adaptive neural control strategy for multiple-input-multiple-output nonlinear systems to handle the nonsymmetric input nonlinearity and the constrained states. Meanwhile, Sun *et al.* [20] employed an adaptive NN in a flexible robotic manipulator to suppress vibrations. Yang and Liu [21] adopted adaptive RBFNN control to improve the position tracking performance of the coupled motor drives system. Dutta *et al.* [22] presented a single-network adaptive critic-based controller for continuous-time systems with unknown dynamics in a policy iteration framework. This control algorithm is verified in a commercial robotic manipulator experiment.

In order to constrain the state and output of nonlinear systems, Tee *et al.* [23], [24] proposed barrier Lyapunov function (BLF) to restrict system states in a prescribed constraint domain. Then, Ren *et al.* [25] adopted BLF integrated with an adaptive neural controller to improve the dynamic behavior of strict-feedback systems. Subsequently, He *et al.* [26] employed BLF to restrict the tracking error in the arbitrary accuracy of the robotic manipulator and improved the robustness of the NN controller. Meanwhile, he also adopted adaptive neural control and distributed parameter control in the robotic manipulator [27]. He *et al.* [17], [27], [28] have successfully proposed novel adaptive NN controllers (ANNCs) for the robots with constraints, where the stability of the closed-loop system is also proved. Then, Yang *et al.* [29] constructed a prescribed tracking performance requirement function integrated with BLF to guarantee both transient and steady-state tracking performance of the dual-arm robot. Zhang [30] studied the multiloop integral controllability of a multiple-input-single-output system to guarantee decentralized unconditional stability under control-loop failure as well as to achieve offset-free tracking performance. Guo *et al.* [31] derived an observer bandwidth constraint to compromise between the dynamic performance and the maximal load capability of the electro-hydraulic system. However, the input constraint is not considered in control design, which may result in larger control magnitude than the traditional controller. In practice, perhaps the input saturation is a serious problem, which should be avoided in most physical systems. Thus, the designed controller has to compromise between the dynamic response performance and the input saturation, especially the initial dynamic response with large state error.

On the other hand, to guarantee the tracking error with convergence rate no less than a prespecified value, Bechlioulis and Rovithakis [32] proposed an adaptive control with prescribed performance constraint (PPC) to overcome the loss of controllability issue with input saturation. Subsequently, Zhang *et al.* [33] adopted PPC to restrict the angle of attack of hypersonic vehicle, which can be more easily applied to handle both static and time-varying constraints than the BLF-based methods. The advantage of PPC is that it converts the original constrained system into an equivalent one without constraints by a weighted performance function. All the system states in the closed loop are uniformly ultimately bounded while the prescribed output constraints are held. Hence, the differences of two mentioned constraint holding techniques are summarized as follows.

1) The PPC is constructed by an adjustable weighted performance function [32], which is positive and monotonically decreasing, while the BLF is a clear logarithm function form [23].
2) Both techniques are suitable for both time-invariant output/state constraint and the time-varying constraint [23], [29]. The function derivation for the time-varying constraint of PPC is more convenient than the logarithm derivation of BLF. Nevertheless, for time-invariant constraint, the difference of structural complexity can be neglected.
3) Considering the physical control saturation, the output constraint of PPC needs low control magnitude by the appropriate design of weighted performance function.
4) The initial output error of PPC can be allowed in a larger scope and the negative effect to the system stability is not relatively sensitive.

Thanks to the research development of motion control of NN-based manipulator control [34], [35], the study is supplied valuable intention. In this paper, inspired by the adaptive NN control proposed in [28] and the prescribed tracking performance function in [29], an adaptive NN control is used in the two-degree-of-freedom (Two-DOF) manipulator driven by electrohydraulic actuators (EHAs). Different from these references, the EHA model is considered in the robotic systems, which implies that the model order is increased from two to three. In this condition, model uncertainties caused by the mechanical structure are more obvious than that of without actuator model, which will decline the output performance of a manipulator. Furthermore, to the best of our knowledge, the robotic manipulator has not been driven by an EHA with PPC technique until now. Simultaneously, an RBFNN is adopted to train the unknown model dynamics emerged in backstepping iteration. Furthermore, considering the parametric uncertainty existed in the manipulator model, an adaptive estimation law is designed to self-tune every trained-node weights of the RBFNN to enhance the proposed controller robustness. The comparison simulations and experimental results with the other two common controllers have verified the effectiveness of the proposed controller in terms of the tracking angle performance and the control current output by the servo valve of an EHA.

The remainder of this paper is organized as follows. The manipulator plant is described in Section II. The adaptive NN control is designed in Section III, including PPC, traditional backstepping control design, the model training by RBFNN, and the adaptive estimation law of node weights. The simulation and experimental results demonstrated on the joint motion of the Two-DOF robotic manipulator are given in Sections IV and V, respectively. Finally, the conclusion is drawn in Section VI.

II. Plant Description

A Two-DOF robotic manipulator is comprised of an upper arm, a forearm, a disc load, and a fixed torso, as shown as in Fig. 1. The shoulder and elbow joints can be driven to rotation by two EHAs.

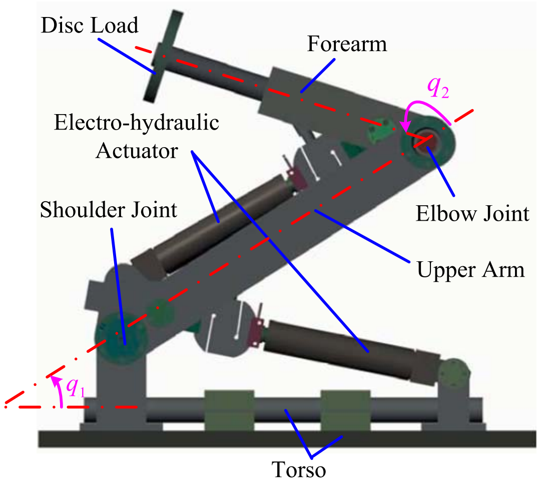

Fig. 1. Single-rod EHA control mechanism.

In the pressure loop of this EHA, the load pressure p_L of the hydraulic cylinder is controlled by the spool position x_v of a servo valve. Since the cutoff frequency of a servo valve is far greater than the control system bandwidth, the valve dynamics can be neglected in model construction [36] as follows:

$$x_v = K_{sv} u \tag{1}$$

where K_{sv} and u are the gain and control current of the servo valve, respectively.

Then, the load pressure p_L output by the hydraulic cylinder [37] is given by

$$\frac{p_L}{x_v} = \frac{a_1 s + a_0}{s + b_0} \tag{2}$$

where a_1, a_0, and b_0 are the hydraulic model parameters.

Additionally, the driven joint torque [38] can be computed by

$$\tau_i = p_{Li} A_p l_i(q_i), \quad i = 1, 2 \tag{3}$$

where $l_i(q_i)$ $(i = 1, 2)$ are the two dynamic force arms, and A_p is the cylinder ram areas.

According to triangle geometry rule, the corresponding force arm $l_i(q_i)$ is computed by

$$\begin{cases} c_i(q_i) = \sqrt{\kappa_i^2 + \varsigma_i^2 - 2\kappa_i \varsigma_i \cos(q_i + \epsilon_i)} \\ l_i(q_i) = \kappa_i \sin\left(\arccos\left(\left(\kappa_i^2 + c_i^2 - \varsigma_i^2\right)/2\kappa_i c_i\right)\right) \end{cases} \tag{4}$$

where $c_i(q_i)$ is the dynamic cylinder length, κ_i and ς are the installing locations of two cylinders, and ϵ_i is the bias angle of cylinder for $i = 1, 2$.

Definition 1: For convenient expression, two operators "$\odot$" and "$\oslash$" are defined as follows:

$$\begin{aligned} \beta \odot \gamma &= [\beta_1, \beta_2, \ldots, \beta_n]^T \odot [\gamma_1, \gamma_2, \ldots, \gamma_n]^T \\ &= [\beta_1\gamma_1, \beta_2\gamma_2, \ldots, \beta_n\gamma_n]^T \\ \beta \oslash \gamma &= [\beta_1, \beta_2, \ldots, \beta_n]^T \oslash [\gamma_1, \gamma_2, \ldots, \gamma_n]^T \\ &= [\beta_1/\gamma_1, \beta_2/\gamma_2, \ldots, \beta_n/\gamma_n]^T. \end{aligned}$$

The dynamic model of a Two-DOF robotic manipulator [39] is described as follows:

$$H(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + f_v(q) \odot l(q) = \tau \tag{5}$$

$$\begin{aligned} H(q) &= \begin{bmatrix} n_1 + n_2 + 2n_3\cos q_2 & \star \\ n_2 + n_3 \cos q_2 & n_2 \end{bmatrix} \\ C(q,\dot{q}) &= \begin{bmatrix} -n_3\dot{q}_2 \sin q_2 & -n_3(\dot{q}_1 + \dot{q}_2)\sin q_2 \\ n_3\dot{q}_1 \sin q_2 & 0 \end{bmatrix} \\ G(q) &= \begin{bmatrix} n_4 g\cos q_1 + n_5 g\cos(q_1+q_2) \\ n_5 g \cos(q_1+q_2) \end{bmatrix} \end{aligned} \tag{6}$$

$$\begin{cases} n_1 &= m_1 L_{c2} + m_2 L_1 + m_f L_1^2 \\ n_2 &= m_2 L_{c2}^2 + I_2 + m_f L_2^2 \\ n_3 &= m_2 L_1 L_{c2} + m_f L_2^2 \\ n_4 &= m_1 L_{c2} + m_2 L_1 + m_f L_1 \\ n_5 &= m_2 L_{c2} + m_f L_2 \end{cases} \tag{7}$$

where $q = [q_1, q_2]^T$, $\dot{q}$, and $\ddot{q}$ are the angular position, velocity, and acceleration, $\tau = [\tau_1, \tau_2]^T$ are the joint torque provided by two EHAs, $f_v(q) = \mu[\mathrm{sgn}(\dot{c}_1(q_1)), \mathrm{sgn}(\dot{c}_2(q_2))]^T$ is the viscous resistance, μ is the viscous coefficient of hydraulic oil, $H(q)$ is the symmetric positive definite inertia matrix, "$\star$" denotes the same transpose element of H, $C(q,\dot{q})\dot{q}$ is the Coriolis force element, $G(q)$ is the gravitational force, m_1 is the upper arm mass, including cylinder, m_2 is the forearm mass, m_f is the disc load mass, L_i is the corresponding link length, L_{c1} is the distance from shoulder joint to the center of mass of upper arm, L_{c2} is the distance from elbow joint to the center of mass of forearm, I_i is the moment of inertia rotating respective center of mass of i link, and g is the gravity constant.

From (1)–(5), if these states are defined as $x_1 = [x_{11}, x_{21}]^T = [q_1, q_2]^T$, $x_2 = [x_{12}, x_{22}]^T = [\dot{q}_1, \dot{q}_2]^T$, and $x_3 = [x_{13}, x_{23}]^T = [p_{L1}, p_{L2}]^T$, then the state-space model of this manipulator system is given by

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = H^{-1}(x_1)(x_3 \odot l(x_1)A_p - f_v(x_1,x_2) \odot l(x_1) \\ \qquad -C(x_1,x_2)x_2 - G(x_1)) \\ \dot{x}_3 = v - b_0 x_3 \end{cases} \tag{8}$$

where $v = K_{sv}(a_1\dot{u} + a_0\, u)$ is considered to be the indirect control variable of this manipulator system.

Remark 1: Without loss of generality, some model parameters in (8), such as n_i $(i = 1, \ldots, 5)$ and μ, are usually uncertain or inaccurate, which can be handled by NN learning in control design.

Property 1 [39]: Since $H(q)$ is symmetric positive definite matrix, the matrix $\dot{H}(q) - 2C(q,\dot{q})$ is skew symmetric.

Problem 1: Considering the dynamic model (8) of a manipulator with parametric uncertainty mentioned in *Remark 1*, an ANNC is designed to guarantee the output state x_1 tracking the demand input x_{1d}. Furthermore, the tracking error $x_1 - x_{1d}$ is restricted in a prescribed performance.

III. Control Design

The proposed controller is designed based on the TBC method together with the output constraint holding technique and the node weights adaptive estimation law of the RBFNN-trained model.

A. Prescribed Performance Constraint

Different from the BLF, the tracking error constraint is realized by a designed weighted performance function $\rho(t)$, which can guarantee the control variable u not beyond required saturation.

The two tracking errors of this manipulator are defined as

$$e_i(t) = x_{i1}(t) - x_{i1d}(t), \quad i = 1, 2. \tag{9}$$

If x_{i1} is constrained in $x_{i1}(t) \in (x_{i1\min}, x_{i1\max})$, and x_{i1d} has a definite boundary $x_{i1d\min} \leq x_{i1d} \leq x_{i1d\max}$, then

$$\underline{k}_i < e_i(t) < \bar{k}_i \quad i = 1, 2 \tag{10}$$

where $\underline{k}_i = x_{i1\min} - x_{i1d\max} < 0$ and $\bar{k}_i = x_{i1\max} - x_{i1d\min} > 0$ are two constants.

Definition 2 [40]: A smooth function $\rho(t) = (\rho_i(0) - \rho_i(\infty))e^{-\lambda t} + \rho_i(\infty)$ is called a weighted performance function if: 1) $\rho(t)$ is positive and monotonically decreasing;

2) $\lim_{x\to\infty} \rho\ (t) = \rho_\infty > 0$; and 3) $\rho\ (\infty) < \rho\ (0) < 1$.

Lemma 1 [32]: If a weighted performance function $\rho(t)$ is designed such that

$$\underline{k}_i < e_i(t)/\rho_i(t) < \bar{k}_i \tag{11}$$

then $e_i(t)$ is constrained in $(\underline{k}_i, \bar{k}_i)$.

In fact, if $e_i(t) \geq 0$, then $e_i(t) \leq e_i(t)/\rho_i(t) < \bar{k}_i$. On the other hand, if $e_i(t) < 0$, then $\underline{k}_i < e_i(t)/\rho_i(t) < e_i(t)$.

Thus from *Lemma 1*, the PPC $\underline{k}_i\rho_i(t) < e_i(t) < \bar{k}_i\rho_i(t)$ can be used to define the system state errors as follows:

$$z_{i1}(t) = T^{-1}\left(\frac{e_i(t)}{\rho_i(t)}\right) = \ln\left(\frac{\bar{k}_i(\underline{k}_i - e_i/\rho_i)}{\underline{k}_i(\bar{k}_i - e_i/\rho_i)}\right), \quad i = 1, 2 \tag{12}$$

where $T(\cdot)$ is a smooth function and $T^{-1}(\cdot)$ is its inverse function.

Theorem 1 [33]: The smooth function $T(\cdot)$ is monotonically increasing and holds the following properties:

$$\underline{k}_i < T(z_{i1}) < \bar{k}_i \qquad T(0) = 0$$
$$\lim_{z_{i1}\to-\infty} T(z_{i1}) = \underline{k}_i \qquad \lim_{z_{i1}\to+\infty} T(z_{i1}) = \bar{k}_i.$$

Proof: From (11), the inverse function of z_{i1} is described as

$$T(z_{i1}) = \frac{e_i(t)}{\rho_i(t)} = \frac{\underline{k}_i\bar{k}_i(e^{z_{i1}} - 1)}{\underline{k}_i e^{z_{i1}} - \bar{k}_i}. \tag{13}$$

Then, the derivative of $T(z_{i1})$ is given by

$$\frac{dT}{dz_{i1}} = \frac{\underline{k}_i(\underline{k}_i - \bar{k}_i)e^{z_{i1}}}{\bar{k}_i\left(\frac{\underline{k}_i}{\bar{k}_i}e^{z_{i1}} - 1\right)^2}. \tag{14}$$

Since $\underline{k}_i < 0$ and $\bar{k}_i > 0$, $dT/dz_{i1} > 0$. Hence, $T(z_{i1})$ is monotonically increasing. Furthermore, due to the PPC $\underline{k}_i\rho_i(t) < e_i(t) < \bar{k}_i\rho_i(t)$, $\underline{k}_i < T(z_{i1}) < \bar{k}_i$ is held. As $z_{i1} \to \pm\infty$, $T(z_{i1})$ approaches its up and down boundaries $\bar{k}_i$ and $\underline{k}_i$, respectively. If $z_{i1} = 0$ is substituted into (12), then $T(0) = 0$. ■

Remark 2: From *Theorem 1*, the PPC $\underline{k}_i\rho_i(t) < e_i(t) < \bar{k}_i\rho_i(t)$ of the tracking error $e_i(t)$ can be transformed into the system state error arbitrarily varying on the scale $-\infty < z_{i1}(t) < +\infty$.

B. Traditional Backstepping Controller

The other two state errors are defined as follows:

$$\begin{aligned} z_{i2} &= x_{i2} - \alpha_{i1} \\ z_{i3} &= x_{i3} - \alpha_{i2} \end{aligned} \qquad i = 1, 2 \tag{15}$$

where α_{ij} is the virtual control variable [41] emerged in the backstepping control design.

For convenient derivation, many 2-D vectors are defined as $z_j = [z_{1j}, z_{2j}]^T$ $(j = 1, 2, 3)$, $\alpha_j = [\alpha_{1j}, \alpha_{2j}]^T$ $(j = 1, 2)$, $e = [e_1, e_2]^T$, and $\rho = [\rho_1, \rho_2]^T$. Then, the candidate Lyapunov functions of (8) are given by

$$\begin{cases} V_1 = z_1^T z_1/2 \\ V_2 = V_1 + z_2^T H(x_1)z_2/2 \\ V_3 = V_2 + z_3^T z_3/2. \end{cases} \tag{16}$$

Step 1: From (11), the derivative of z_1 is given by

$$\dot{z}_1 = R(x_2 - \dot{x}_{1d} - e \odot \dot{\rho} \oslash \rho) \tag{17}$$

where $R = \text{diag}(r_1, r_2) \in \mathbb{R}^{2\times 2}$

$$r_i = \frac{\partial T^{-1}}{\partial\ (e_i/\rho_i)}\frac{1}{\rho_i} = \frac{\bar{k}_i - \underline{k}_i}{(\bar{k}_i - e_i/\rho_i)(e_i/\rho_i - \underline{k}_i)\rho_i}, \quad i = 1, 2.$$

If the virtual control α_1 is designed as

$$\alpha_1 = \dot{x}_{1d} + e \odot \dot{\rho} \oslash \rho - R^{-1}C_1 z_1 \tag{18}$$

where $C_1 = \text{diag}(c_{11}, c_{21}) \in \mathbb{R}^{2\times 2}$ is a positive definite constant matrix, then the derivative of V_1 in (16) is given by

$$\dot{V}_1 = -z_1^T C_1 z_1 + z_1^T R z_2. \tag{19}$$

Step 2: Substituting *Property 1* into the derivative of V_2, we obtain that

$$\begin{aligned} \dot{V}_2 &= \dot{V}_1 + z_2^T H(x_1)\dot{z}_2 + z_2^T \dot{H}(x_1)z_2/2 \\ &= -z_1^T C_1 z_1 + z_2^T(Rz_1 + z_3 \odot lA_p + \alpha_2 \odot lA_p \\ &\quad - H\dot{\alpha}_1 - f_\upsilon \odot l - C\alpha_1 - G). \end{aligned} \tag{20}$$

If the virtual control α_2 is designed as

$$\alpha_2 = A_p^{-1}(-Rz_1 + H\dot{\alpha}_1 + f_\upsilon \odot l + C\alpha_1 + G - C_2 z_2) \oslash l \tag{21}$$

where $C_2 = \text{diag}(c_{12}, c_{22})$ is similar to C_1, then

$$\dot{V}_2 = -z_1^T C_1 z_1 - z_2^T C_2 z_2 + z_2^T z_3 \odot lA_p. \tag{22}$$

Step 3: The derivative of V_3 is given by

$$\begin{aligned} \dot{V}_3 &= \dot{V}_2 + z_3^T \dot{z}_3 \\ &= -z_1^T C_1 z_1 - z_2^T C_2 z_2 \\ &\quad + z_3^T(A_p l \odot z_2 + \upsilon - b_0 x_3 - \dot{\alpha}_2). \end{aligned} \tag{23}$$

If the final control variable υ is designed as

$$\upsilon = -C_3 z_3 - A_p l \odot z_2 + b_0 x_3 + \dot{\alpha}_2 \tag{24}$$

then

$$\dot{V}_3 \leq -z_1^T C_1 z_1 - z_2^T C_2 z_2 - z_3^T C_3 z_3 < 0. \tag{25}$$

Remark 3: From (16) and (25), the control variable υ (24) integrated with the virtual controls α_1 (18) and α_2 (21) can not only guarantee all the system errors z_{ij} $(i = 1, 2,\ j = 1, 2, 3)$ asymptotic to zero but also restrict the dynamic errors e_1 and e_2 in the PPC $\underline{k}_i \rho_i(t) < e_i(t) < \bar{k}_i \rho_i(t)$.

C. Adaptive Neural Network Controller

According to *Remark 1*, some parametric uncertainties exist in the matrices H, C, G, and f_υ, which lead to negative effect in the virtual control α_2 (21) and the final control variable υ (24). Thus, an adaptive NN is adopted to handle these unknown dynamics.

Definition 3 [28]: A class of RBFNN is usually used to estimate an unknown continuous function $f_i(X)$ as follows:

$$f_i(X) = W_i^T S_i(X) + \varepsilon_i(X), \quad i = 1, 2 \tag{26}$$

where X is the input vector, W_i is the weight vector with the k_i nodes of the RBFNN, $S_i(X) = [s_1, s_2, \ldots, s_{k_i}]^T$ is the Gaussian basis function vector, and $\varepsilon_i(X)$ is the estimation error of the RBFNN, which is bounded by $|\varepsilon_i(X)| < \varepsilon_{\max}$ for $i = 1, 2$, and $\varepsilon_{\max}$ is an unknown boundary.

The radial basis element s_j of $S_i(X)$ is given by

$$s_j(X) = \exp\left(\frac{-(X - \mu_j)^T (X - \mu_j)}{\sigma_j^2}\right), \ j = 1, \ldots, k_i \tag{27}$$

where μ_j is the center of the receptive field and σ_j is the width of the Gaussian function [42].

If the input vector X of the RBFNN is defined as $X = [x_1, x_2, \alpha_1, \dot{\alpha}_1]^T$, then the RBFNN estimation $W^T S(X)$ can be trained by sufficient sample data to approximate the following uncertain dynamic model:

$$\begin{aligned} W^T S(X) + \varepsilon(X) &= H(x_1)\dot{\alpha}_1 + C(x_1, x_2)\alpha_1 \\ &\quad + G(x_1) + f_\upsilon(x_1) \odot l(x_1) \\ &= F(X, \delta n_i, \delta\mu) \end{aligned} \tag{28}$$

where F is the model uncertainty by model parameters n_i $(i = 1, \ldots, 5)$ and μ is mentioned in *Remark 1*.

Simultaneously, to enhance the robustness of the RBFNN controller, an adaptive estimation law for the weight vector W_i is designed as follows:

$$\dot{\hat{W}}_i = -\Gamma_i (S_i(X) z_{2i} + \eta_i \hat{W}_i), \quad i = 1, 2 \tag{29}$$

where $\hat{W}_i$ is the adaptive estimated value of the ideal weight W_i, and $\Gamma_i > 0$ and $\eta_i > 0$ are the diagonal matrix and constant gains of the adaptive estimation law.

Remark 4: The weight estimation $\hat{W}_i$ is online self-tuning by (29) based on a preliminary training value $W_i(0)$ of the designed RBFNN, which can eliminate the negative effect of parametric uncertainties existed in H, C, G, and f_υ.

If the adaptive weight estimation law (29) is considered in backstepping iteration, then the virtual control α_2 (21) is revised as follows:

$$\alpha_2 = A_p^{-1}\left(-R z_1 + \sum_{i=1}^{2} \hat{W}_i^T S_i(X) - C_2 z_2\right) \oslash l. \tag{30}$$

Theorem 2: Considering the manipulator model (8), if the RBFNN controller is designed as (18), (24), (28), and (30), together with the adaptive weight estimation law (29), then the system state errors (12) and (15) are all ultimate boundary [43] and the error convergence domain is an hypersphere H_r, i.e.,

$$H_r \in \left\{ \begin{aligned} z_1^T z_1 &+ z_2^T H z_2 + z_3^T z_3 + \sum_{i=1}^{2} \tilde{W}_i^T \Gamma_i^{-1} \tilde{W}_i \\ &= 2V_3(0) e^{-\lambda^* t_f} + 2\delta/\lambda^* \end{aligned} \right\} \tag{31}$$

where λ^* and δ are the positive constants, and $V_3(0)$ is the initial system state error, $\forall t > t_f$ (t_f is a finite time).

Proof: Consider the candidate Lyapunov functions of (8) as follows:

$$\begin{cases} V_1 = z_1^T z_1 / 2 \\ V_2 = V_1 + z_2^T H(x_1) z_2 / 2 + \sum_{i=1}^{2} \tilde{W}_i^T \Gamma_i^{-1} \tilde{W}_i / 2 \\ V_3 = V_2 + z_3^T z_3 / 2 \end{cases} \tag{32}$$

where $\tilde{W}_i = W_i - \hat{W}_i$ is the self-tuning weight error of W_i.

Then, the virtual control $\alpha_i (i = 1, 2)$ and the final control variable υ can also be derived by the derivatives of $V_i (i = 1, 2, 3)$.

Different from the backstepping iteration in Section III-B, substituting (28) into (20), the derivative $\dot{V}_2$ is given by

$$\begin{aligned} \dot{V}_2 &= \dot{V}_1 + z_2^T H(x_1)\dot{z}_2 + z_2^T \dot{H}(x_1) z_2 / 2 - \sum_{i=1}^{2} \tilde{W}_i^T \Gamma_i^{-1} \dot{\hat{W}}_i \\ &= -z_1^T C_1 z_1 + z_2^T \Bigg(R z_1 + z_3 \odot l A_p + \alpha_2 \odot l A_p \\ &\quad - \sum_{i=1}^{2} \hat{W}_i^T S_i - \sum_{i=1}^{2} \tilde{W}_i^T S_i - \varepsilon \Bigg) - \sum_{i=1}^{2} \tilde{W}_i^T \Gamma_i^{-1} \dot{\hat{W}}_i. \end{aligned} \tag{33}$$

By Young's inequality, we can obtain

$$\begin{aligned} z_2^T \varepsilon &\leq \left(z_2^T z_2 + \|\varepsilon\|^2\right)/2 \\ \tilde{W}_i^T W_i &\leq \left(\tilde{W}_i^T \tilde{W}_i + W_i^T W_i\right)/2. \end{aligned} \tag{34}$$

Substituting the revised virtual control α_2 (30), the adaptive weight estimation law (29), and Young's inequalities (34) into (33), the derivative $\dot{V}_2$ is given by

$$\begin{aligned} \dot{V}_2 \leq & -z_1^T C_1 z_1 - z_2^T \left(C_2 - \frac{1}{2} I_{2\times 2}\right) z_2 - \sum_{i=1}^{2} \frac{\gamma_i}{2} \|\tilde{W}_i\|^2 \\ & + \frac{1}{2}\|\varepsilon\|^2 + \sum_{i=1}^{2} \frac{\gamma_i}{2}\|W_i\|^2 + z_2^T z_3 \odot l A_p. \end{aligned} \tag{35}$$

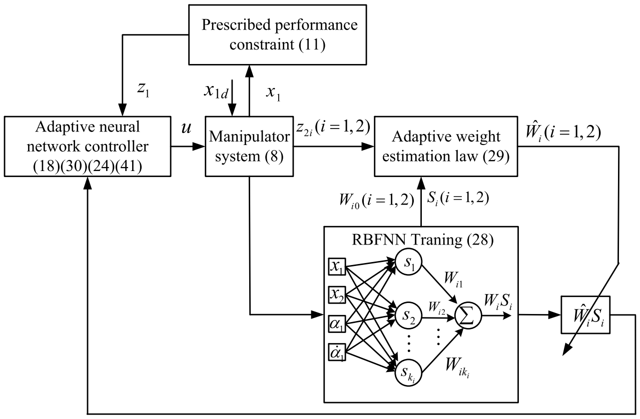

Fig. 2. Block diagram of the adaptive NN control scheme.

Similar to *Step 3*, if the control variable υ (24) is substituted into the derivative of V_3 in (32), then

$$\dot{V}_3 \le -z_1^T C_1 z_1 - z_2^T\left(C_2 - \frac{1}{2}I_{2\times2}\right)z_2 - z_3^T C_3 z_3 - \sum_{i=1}^{2}\frac{\gamma_i}{2}\|\tilde{W}_i\|^2 + \frac{1}{2}\|\varepsilon\|^2 + \sum_{i=1}^{2}\frac{\gamma_i}{2}\|W_i\|^2. \quad (36)$$

If a positive constant λ^* is defined as

$$\lambda^* = \min\left\{\begin{matrix} 2\lambda_{\min}(C_1), & \dfrac{2\lambda_{\max}(C_2 - \frac{1}{2}I_{2\times2})}{\lambda_{\min}(H)} \\ 2\lambda_{\min}(C_3), & \min\limits_{i=1,2}\left(\dfrac{\sigma_i}{\lambda_{\max}(\Gamma_i^{-1})}\right)\end{matrix}\right\} \quad (37)$$

then (36) becomes

$$\dot{V}_3 \le -\lambda^* V_3 + \delta \quad (38)$$

where the positive constant $\delta = \|\varepsilon\|^2/2 + \sum_{i=1}^{2}\sigma_i\|W_i\|^2/2$.

When premultiply and postmultiply the inequality (38) by $e^{\lambda^* t}$, the following equality holds:

$$\frac{d(V_3 e^{\lambda^* t})}{dt} = \delta e^{\lambda^* t}. \quad (39)$$

Integrating (39), the following inequality holds:

$$V_3(t) \le \left(V_3(0) - \frac{\delta}{\lambda^*}\right)e^{-\lambda^* t} + \frac{\delta}{\lambda^*} \le V_3(0)e^{-\lambda^* t} + \frac{\delta}{\lambda^*}. \quad (40)$$

Now substituting (32) into (40), and let $t > t_f$, the error convergence domain H_r in (31) is obtained. Furthermore, the size of H_r mainly depends on the element δ/λ^*. Thus, the increased control gains $C_i(i=1,2,3)$ and the parametric estimation gains $\Gamma_i(i=1,2)$ can arbitrarily shrink the size of H_r as $t \to \infty$. ■

From (1), (2), and the indirect control variable υ in (8), the actual control current u of the servo valve is given by

$$u = \frac{\upsilon}{K_{sv}(a_1 s + a_0)}. \quad (41)$$

Fig. 2 shows the block diagram of the proposed adaptive NN control scheme. The RBFNN (28) is trained by the system states x_1 and x_2, the virtual control α_1, and its derivative $\dot{\alpha}_1$ in the traditional backstepping iteration from (18), (21), and (24), which can obtain the preliminary weight value W_{i0} and the RBF $S_i(X)$ for $i=1,2$. Then, the actual weights $\hat{W}_i(i=1,2)$ are online self-tuning by the adaptive weight estimation law (29), which compensates the inaccurate dynamic model in (8). According to the PPC (11), the ANNC u (41) is constructed by (18), (24), and (30) to guarantee the dynamic performance of the manipulator system (8).

TABLE I
SOME MECHANICAL AND HYDRAULIC PARAMETERS

Parameter	Value	Parameter	Value
m_1	6.012 kg	m_2	1.479 kg
m_f	1.068 kg	I_1	$0.118\ \mathrm{kg}\cdot\mathrm{m}^2$
I_2	$0.017\mathrm{kg}\cdot\mathrm{m}^2$	L_{c1}	0.177 m
L_{c2}	0.114 m	L_1	0.438 m
L_2	0.345 m	κ_i	0.403 m
ς_i	0.055 m	ϵ_1	6.1°
ϵ_2	13.8°	b_0	124
a_1	7	a_0	2.2×10^8
μ	500 s/m	A_p	$4.91 \times 10^{-4}m^2$
K_{sv}	0.125 mm/mA	g	$9.81\ \mathrm{m}^2/\mathrm{s}$

IV. SIMULATION

To verify the proposed ANNC in simulation, some nominal mechanical and hydraulic parameters of this robotic manipulator are shown in Table I. The hydraulic parameters a_1, a_0, and b_0 are simplified from the linear load pressure model p_L controlled by the valve spool position x_υ [44].

The motion ranges of two joint angles are $29.84° \le q_1 \le 115.76°$ and $47.85° \le q_2 \le 135.92°$. The prescribed performance boundaries of two tracking errors $\underline{k}_1 = \underline{k}_2 = -100°$ and $\bar{k}_1 = \bar{k}_2 = 100°$. Three parameters of the weighted performance function are $\rho(0) = [0.95, 0.95]^T$, $\rho(\infty) = [0.02, 0.02]^T$, and $\lambda = 0.5$. The control gains are designed as $C_1 = \mathrm{diag}\{1, 1\}$, $C_2 = \mathrm{diag}\{10, 10\}$, and $C_3 = \mathrm{diag}\{2000, 2000\}$. The approximate order of magnitude for these control gains is determined by the virtual controls (18) and (21) and the final control (24). For the RBFNN estimation, 68 and 57 nodes are used for each $S_i(X)$ with centers selected in the area of $[-1, 1]$ with 8-D grids. The variances of centers are $\sigma_j^2 = 1,\ (j = 1, \ldots, k_i)$. Two diagonal matrix gains of the adaptive estimation law are $\Gamma_1 = 20 \times I_{68}$ and $\Gamma_2 = 20 \times I_{57}$, where I_n denotes the $n \times n$ identity matrix, $\eta_1 = \eta_2 = 0.02$. These matrix gains are well tuned, considering both the fast convergent weights and the desirable tracking performance.

To illustrate the problem, the proposed ANNC based on (18), (24), (30), and (41) is compared with the following two controllers as follows.

1) Proportional–integral-derivative (PID) controller $u = k_p\ (y_d - x_1) + k_i \int (y_d - x_1)dt + k_d\ (\dot{y}_d - x_2)$, where the control gains k_p = 140, k_i = 17, and $k_d = 7$ have been well tuned to guarantee fast responses of two robotic joint angles.
2) The traditional backstepping controller (TBC) based on (18), (21), (24), and (41).

A. Comparison With PID Control

Two sinusoidal demands of the joint angles are $q_{1d} = 33\sin(1\pi t) + 72.8°$ and $q_{2d} = 34\sin(2\pi t) + 91.9°$. The comparison results of the two controllers are shown in Figs. 3–5.

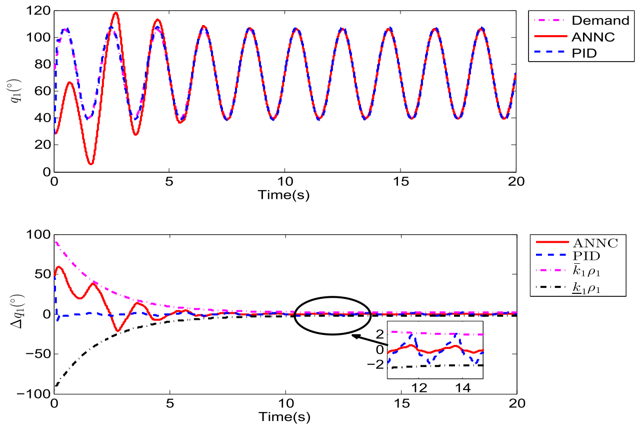

Fig. 3. Comparison results with PID controller. q_1 is the shoulder joint angle and $\Delta q_1 = q_{1d} - q_1$ is the tracking angle error.

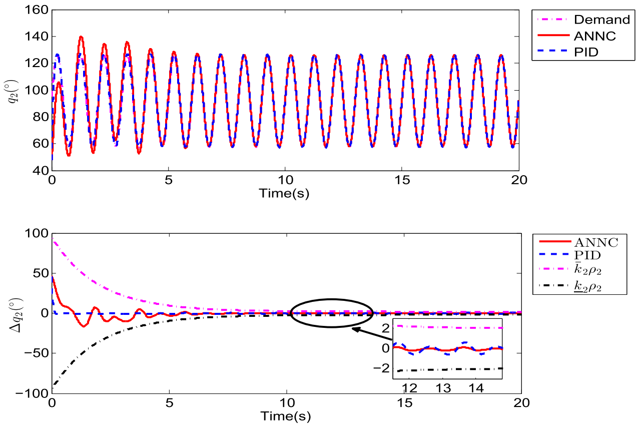

Fig. 4. Comparison results with the PID controller. q_2 is the elbow joint angle and $\Delta q_2 = q_{2d} - q_2$ is the tracking angle error.

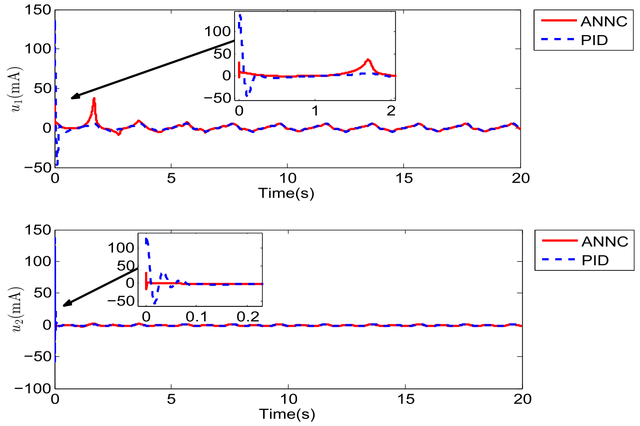

Fig. 5. Comparison results with the PID controller. u_1 and u_2 are the control currents of the two joint EHAs.

As the time $t < 5$ s, the dynamic tracking performances $\Delta q_i (i = 1, 2)$ of PID are better than ANNC. However, the two control currents u_i $(i = 1, 2)$ of the PID controller are more consumed than that of ANNC, which surpass 100 mA near the initial zero time. Since the weighted performance function $\rho(t)$ is exponentially attenuated, the dynamic tracking error of ANNC is restricted in $\underline{k}_i\rho_i(t) < e_i(t) < \bar{k}_i\rho_i(t)$ mentioned in (11). As two joint angles approach their steady states, the steady-state errors of ANNC $|e_i| < 2°$ [i.e., $\underline{k}_i\rho_i(\infty) < e_i < \bar{k}_i\rho_i(\infty)$], which has better performances than PID, as shown in Fig. 5. Certainly, the steady tracking error of PID can be further reduced by increasing PID gains. However, the system stability margin may be degraded.

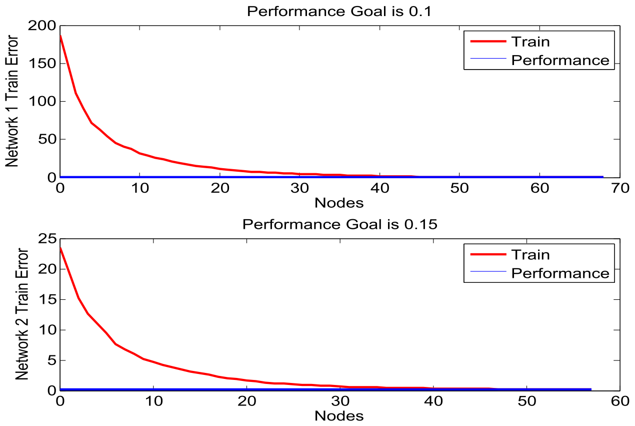

Fig. 6. Train results of two NNs by (28).

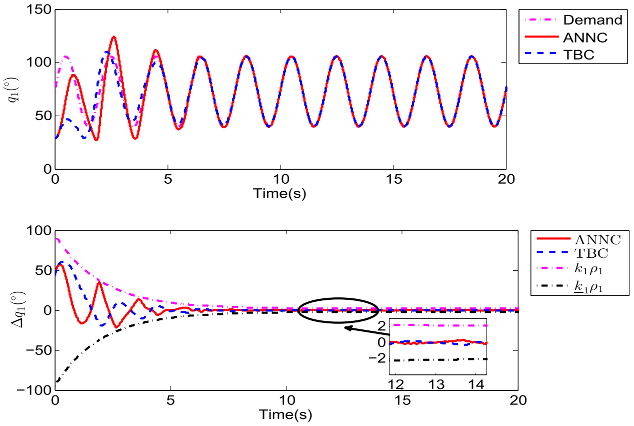

Fig. 7. Comparison results with traditional backstepping controller. q_1 is the shoulder joint angle. $\Delta q_1 = q_{1d} - q_1$ is the tracking angle error.

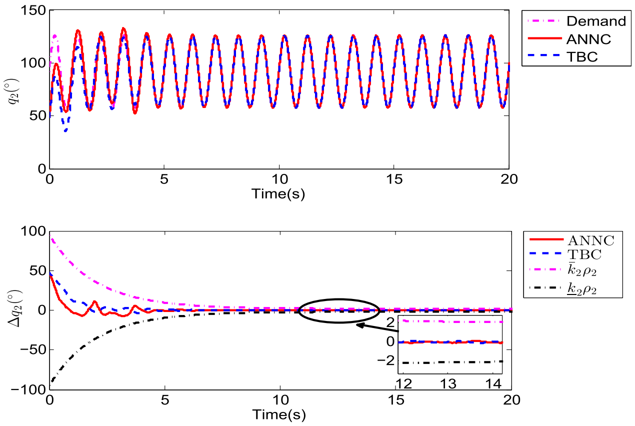

Fig. 8. Comparison results with traditional backstepping controller. q_2 is the elbow joint angle. $\Delta q_2 = q_{2d} - q_2$ is the tracking angle error.

B. Comparison With TBC

The two joint angle demands are the same as Section IV-A. To realize the ANNC algorithm, the model (28) is trained by two RBFNNs $W_i^T S_i(X)$ $(i = 1, 2)$. If two performance

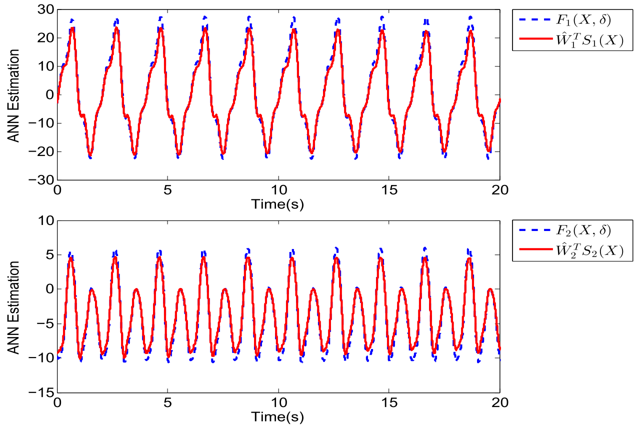

Fig. 9. ANN estimation for model uncertainty, where $F_i(X,\delta)$ is the model uncertainty and $\hat{W}_i^T S_i$ is the corresponding estimation of ANN.

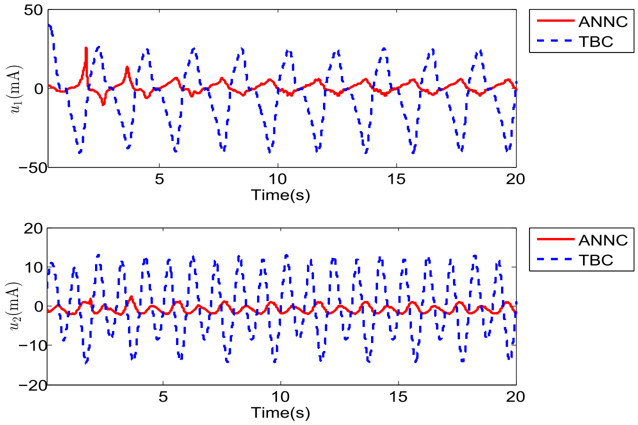

Fig. 10. Comparison results with a traditional backstepping controller. u_1 and u_2 are the control currents of two joint EHAs.

goals are selected as 0.1 and 0.15, the model-trained errors are asymptotic convergence by the selected 68 and 57 network nodes, respectively, as shown in Fig. 6. Then, the well-trained weights of the two RBFNNs are obtained and used as the initial value of the adaptive weight estimation law (29). Figs. 7–10 show the comparison results of ANNC with TBC. Since the PPC is considered by both the two controllers, the dynamic and steady tracking errors of ANNC are similar to TBC, which indicates the favorable optimal performance of the model (28) by the RBFNN. After 5 s, the tracking errors of two joint angles are restricted in 1°. However, the control current magnitude of TBC is still larger than ANNC, which surpasses 20 mA, as shown in Fig. 10. Thus, the PPC (11) consumes obvious control capability by TBC without adaptive weight estimation law. The model estimation results by adaptive NN (28) and (29) are shown in Fig. 9. The model estimation error about $H\dot{\alpha}_1 + C\alpha_1 + G + f_\upsilon \odot l$ includes two elements, i.e., the RBFNN estimation error $\varepsilon_i(X)$ and the self-tuning weight error $\tilde{W}_i$. From Fig. 9, the model estimation error can be constrained in a satisfactory neighborhood by ANNC.

C. Comparative Results With Parametric Uncertainty

To verify the robustness of ANNC, two sinusoidal demands of the joint angles are chosen as $q_{1d} = 33\sin(1.3\pi t) + 72.8°$ and $q_{2d} = 34\sin(1.5\pi t) + 91.9°$. Some parametric uncertainties are assumed as $m_1' = 1.5\, m_1$, $m_2' = 1.5\, m_2$, $m_f' = 1.5\, m_f$, $L_1' = 1.3\ L_1$, $L_2' = 0.7\ L_2$, $I_1' = 0.7\ I_1$, $I_2' = 1.3\ I_2$, and $\mu' = 1.5\mu$. The control current saturation is $u_{\max} = \pm 20$ mA. Then, the comparison results are shown in Figs. 11–13. The dynamic and steady tracking errors can also be restricted in the PPC (11) by TBC and ANNC. However, the control saturation emerges in the shoulder joint motion by TBC, as shown

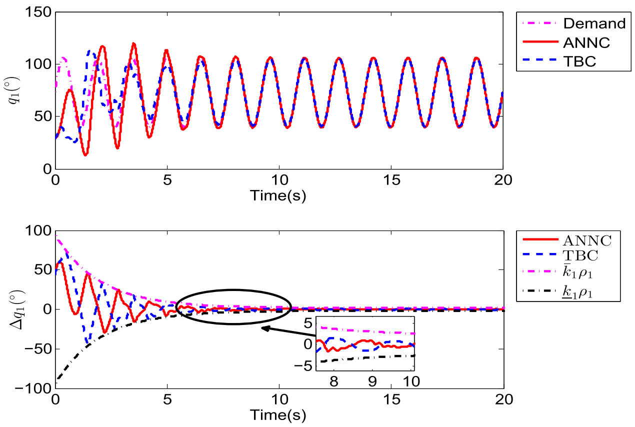

Fig. 11. Comparison results with the traditional backstepping controller, q_1 is the shoulder joint angle, $\Delta q_1 = q_{1d} - q_1$ is the tracking angle error.

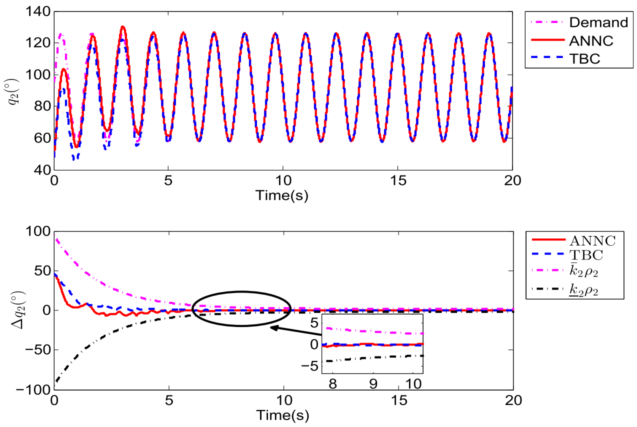

Fig. 12. Comparison results with traditional backstepping controller. q_2 is the elbow joint angle and $\Delta q_2 = q_{2d} - q_2$ is the tracking angle error.

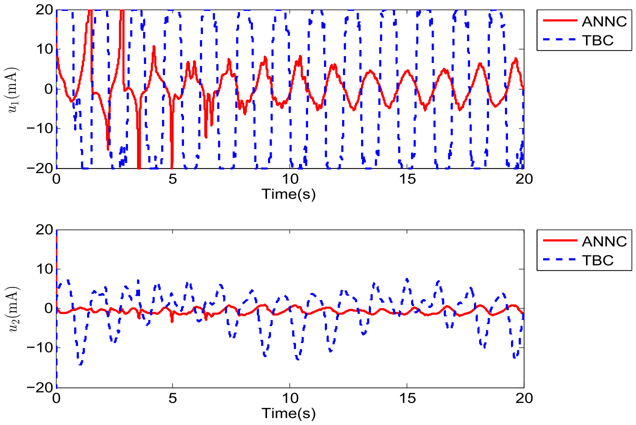

Fig. 13. Comparison results with the traditional backstepping controller. u_1 and u_2 are the control currents of two joint EHAs.

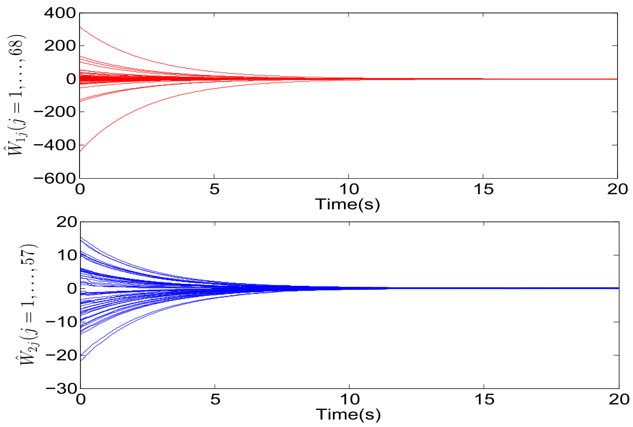

Fig. 14. Weight estimations by (29). $\hat{W}_{1j}$ $(j = 1, \ldots, 68)$ and $\hat{W}_{2j}$ $(j = 1, \ldots, 57)$ are the respective estimation of two numbers of nodes.

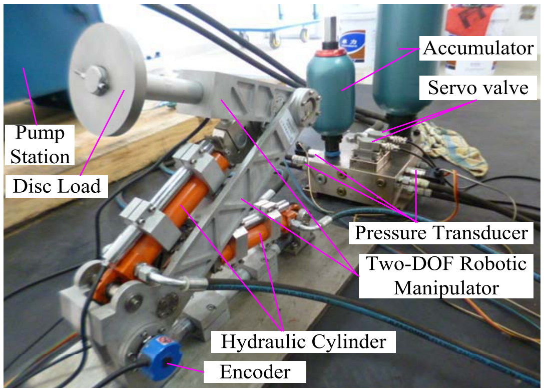

Fig. 15. Experimental bench of the robotic manipulator driven by the EHA.

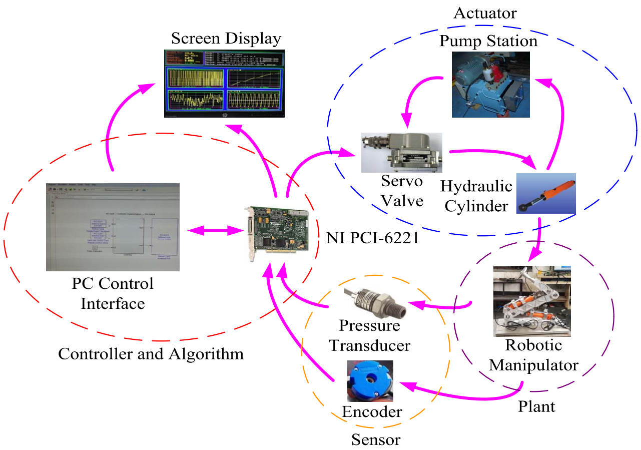

Fig. 16. Control implementation diagram of the robotic manipulator.

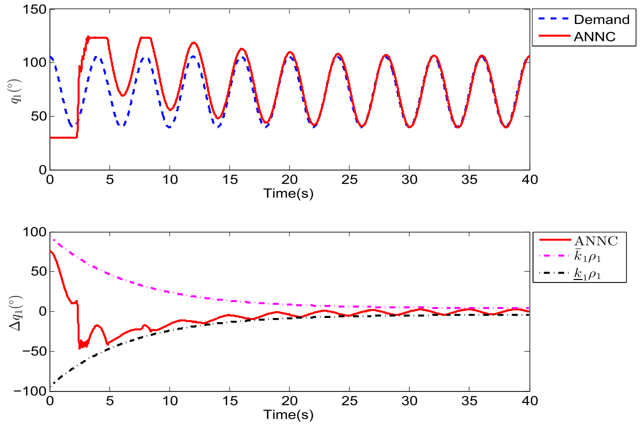

Fig. 17. Experimental results of ANNC. q_1 is the shoulder joint angle and $\Delta q_1 = q_{1d} - q_1$ is the tracking angle error.

in Fig. 13. Two control currents of TBC u_1 and u_2 are larger than that of ANNC. Due to the parametric uncertainties injected in the manipulator system (8), TBC consumed obvious control cost to compensate the model error and to guarantee two tracking angle errors in the PPC. However, it is necessary for ANNC to adopt the adaptive NN estimation model (30) rather than the uncertain model (21) by TBC. Even though there exists modeling uncertainties, the dynamic and steady tracking errors of two joint angles can be still restricted in the PPC $[\underline{k}_i\rho_i(t) < e_i(t) < \bar{k}_i\rho_i(t)]$. Furthermore, the control current output by the two servo valves of EHAs is satisfactory, as the adaptive weight estimation law is adopted to self-tune every node weight, as shown in Fig. 14.

V. Experiment

The experimental bench of a Two-DOF robotic manipulator is set up as shown in Fig. 15 to verify the effectiveness of the proposed ANNC. Two EHAs are composed of two servo valves (FF-102/03021T240), two cylinders (UG1511R25/16-80), a pump station (HY-36CC-01/11kw), and two accumulators (NXQ1-L1.0/31.5H). Four cylinder pressures are measured by the pressure transducer (BD-Sensors-DMP-331). Two joint angles are measured by the incremental encoder (HENGSILL ALN722R7LSDN13188).

The control implementation of this experiment is shown in Fig. 16. The encoder and the pressure transducer data are sampled by National Instruments (NI) card as the feedback information. The control algorithm is realized by MATLAB/Simulink tool in the host PC computer and the control demand is returned to NI card, which drives the servo valve to regulate the cylinder pressure supplied by the pump station. The interval of the whole algorithm execution is 5 ms. Due to the variable load pressure, the joint motion control of the robotic manipulator is driven by this EHA. The screen display shows the measurement and computed signals in real time.

A. Experimental Results of ANNC

Two joint angle demands are chosen as $q_{1d} = 33 \sin(0.5\pi t) + 72.8°$ and $q_{2d} = 34 \sin(\pi t) + 91.9°$. The experimental results of ANNC are shown in Figs. 17–20. The dynamic and steady tracking errors of two joint angles are restricted in the PPC $[\underline{k}_i\rho_i(t) < e_i(t) < \bar{k}_i\rho_i(t)]$. The steady tracking errors $\Delta q_i (i = 1, 2)$ are less than 4°, as shown in Figs. 17 and 18. From Fig. 20, since the load pressure of shoulder EHA p_{L1} is larger than that of elbow actuator p_{L2}, the dynamic response of shoulder angle is slower than that of the elbow angle. Thus, two hydraulic accumulators are used to store energy and improve the flow velocity in the hydraulic control loop. However, the duration of energy storage integrated with pressure preparation is approximately

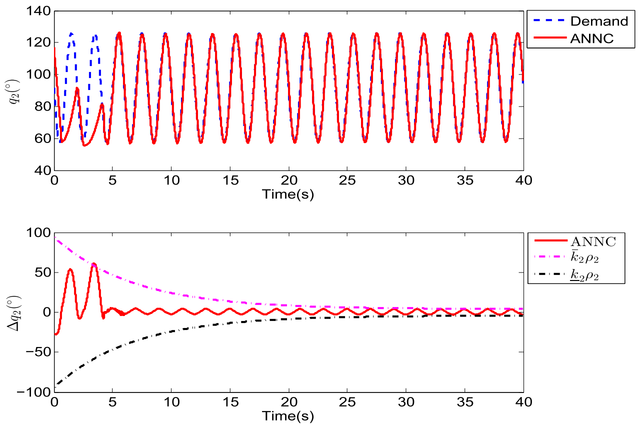

Fig. 18. Experimental results of ANNC. q_2 is the elbow joint angle and $\Delta q_2 = q_{2d} - q_2$ is the tracking angle error.

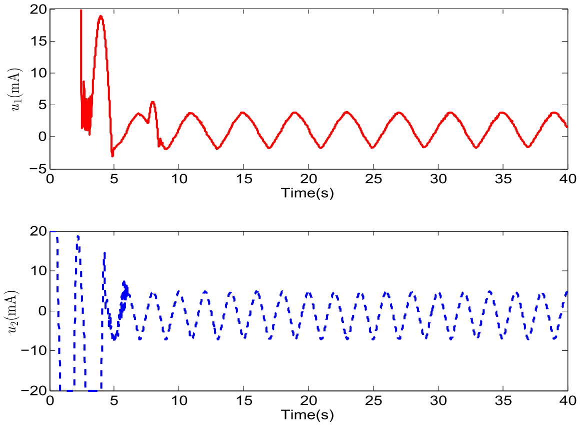

Fig. 19. Experimental results of ANNC. u_1 and u_2 are the control currents of two joint EHAs.

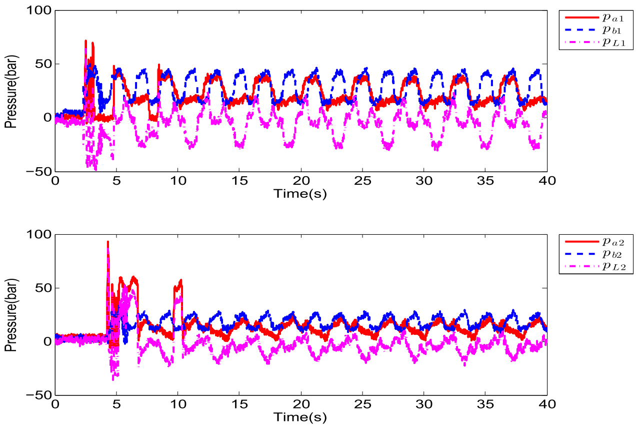

Fig. 20. Experimental results of ANNC. p_{ai} and p_{bi} are the two cylinder chamber pressures. $p_{Li} = p_{ai} - p_{bi}$ is the load pressure of the corresponding EHA for $i = 1, 2$.

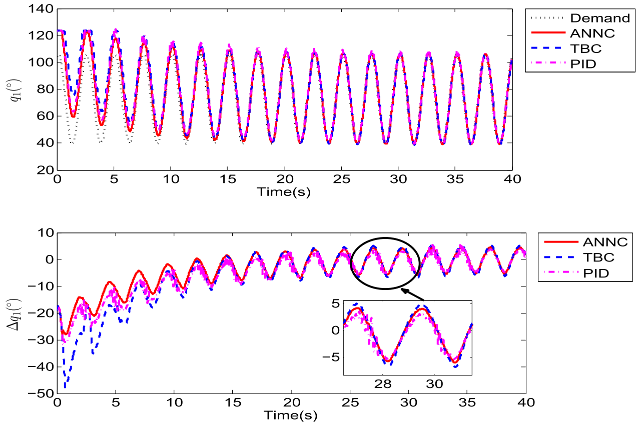

Fig. 21. Experimental results of three controllers. q_1 is the shoulder joint angle and $\Delta q_1 = q_{1d} - q_1$ is the tracking angle error.

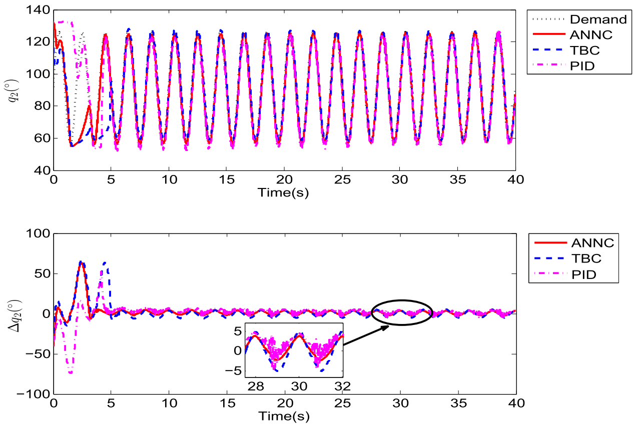

Fig. 22. Experimental results of three controllers. q_2 is the elbow joint angle and $\Delta q_2 = q_{2d} - q_2$ is the tracking angle error.

2–5 s, as shown in Fig. 20, which results in the control current saturation of ±20 mA in the initial response time of ANNC, as shown in Fig. 19. After 10 s, two joint angles approach the steady state and the control current magnitudes of two servo valves are periodically regulated to guarantee two joint angles $q_i (i = 1, 2)$ that track the corresponding demands q_{id} $(i = 1, 2)$. Meanwhile, the cylinder chamber pressures p_{ai} and p_{bi} $(i = 1, 2)$ of two EHAs are less than 50 bar, which are constrained by the supply pressure p_s of the pump station.

B. Comparison Results

Then, two joint demands are chosen as $q_{1d} = 33 \sin (0.8\pi t) + 72.8°$ and $q_{2d} = 34 \sin(\pi t) + 91.9°$. The experimental results of three controllers are shown in Figs. 21–24. Although the steady tracking errors of PID are less than the other two controllers, some angle chatters emerge in two joint responses, as shown in Figs. 21 and 22. Of course, the control gains of PID can be reduced to eliminate these angle chatters, which may degrade the dynamic and steady performance. Two steady errors of ANNC are less than 4°, which has more favorable performances than TBC. From Fig. 23, the control current of elbow actuator approaches the control saturation, since the elbow motion frequency is larger than the shoulder. Furthermore, some control chatters also emerges in PID rather than the other two controllers. The model estimation results $\hat{W}_i^T S_i(X)$ $(i = 1, 2)$ by the ANN (28) and (29) are shown in Fig. 24, which indicates the robustness of ANNC when two joint demands vary with different motion frequencies.

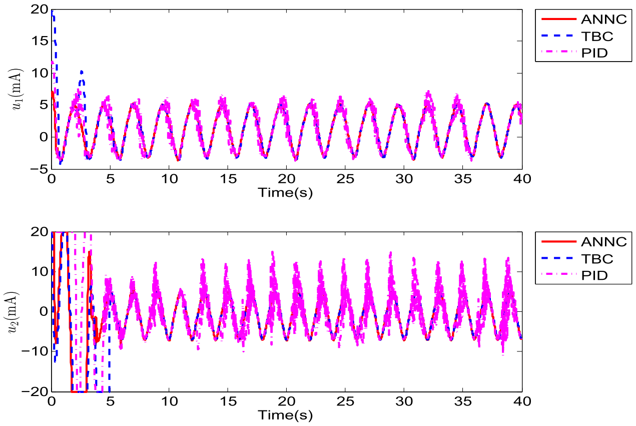

Fig. 23. Experimental results of three controllers. u_1 and u_2 are the control currents of two joint EHAs.

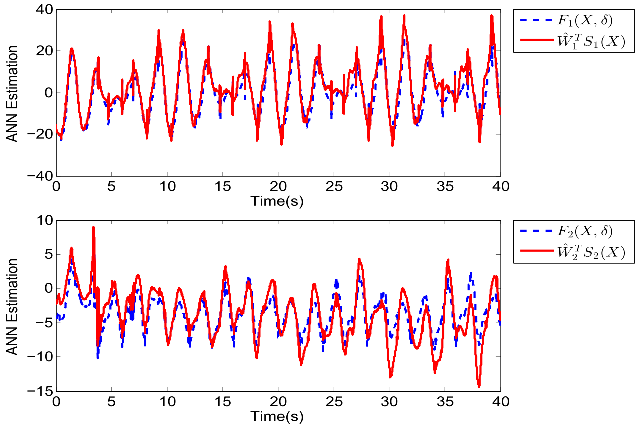

Fig. 24. ANN estimation for model uncertainty, where $F_i(X, \delta)$ is the model uncertainty and $\hat{W}_i^T S_i$ is the corresponding estimation of ANN.

VI. Conclusion

In this paper, an adaptive NN control with backstepping was proposed for the Two-DOF manipulator driven by the EHA. Considering the output constrained problem, a weighted performance function was designed to restrict the tracking angle errors of two joints in a PPC. To avoid the unknown dynamics in the model-based control design, an RBFNN was constructed to train the unknown model dynamics. Although the network-estimated model depended on different training samples, the node weights of RBFNN can be self-tuned by an adaptive estimation law according to the system state errors. The comparison results with PID and traditional backstepping controller indicated that the proposed ANNC had three advantages as follows.

1) The unknown dynamic model parameters need not to be preknown. By the RBFNN learning, the complicated nonlinear model can be well identified.
2) The control sensitivity will be relaxed and the robustness is improved when some model dynamics and parametric uncertainty exit in the robotic manipulator.
3) The RBFNN-based controller will not be easy to emerge chatters when the demand frequency and magnitude increased and guarantee the desirable tracking performance by PPC, which indicates the adaptation capability of the RBFNN to address uncertain parameters and disturbance.

To be honest, due to the control saturation of the hydraulic actuator and the load pressure delay caused by hydraulic pipe transmission, the control gains cannot be big enough and the experimental dynamic response of the proposed controller is certainly lower than the corresponding simulation results. To further improve the performance of this manipulator, the controller should consider the time-delay model of the EHA, and the hydraulic elements configuration may be optimized in the future.

References

[1] Z. Li, S. S. Ge, M. Adams, and W. S. Wijesoma, “Adaptive robust output-feedback motion/force control of electrically driven nonholonomic mobile manipulators,” *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 6, pp. 1308–1315, Nov. 2008.

[2] W. Dong, “On trajectory and force tracking control of constrained mobile manipulators with parameter uncertainty,” *Automatica*, vol. 38, no. 9, pp. 1475–1484, Sep. 2002.

[3] Z. Li, C. Yang, C. Y. Su, and W. Ye, “Adaptive fuzzy-based motion generation and control of mobile under-actuated manipulators,” *Eng. Appl. Artif. Intell.*, vol. 30, pp. 86–95, Apr. 2014.

[4] Y. Yan, J. Xu, and M. Wiercigroch, “Basins of attraction of the bistable region of time-delayed cutting dynamics,” *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 96, no. 3, p. 032205, Sep. 2017.

[5] Y.-J. Liu, L. Tang, S. Tong, and C. L. P. Chen, “Adaptive NN controller design for a class of nonlinear MIMO discrete-time systems,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 5, pp. 1007–1018, May 2015.

[6] B. Xu, C. Yang, and Z. Shi, “Reinforcement learning output feedback NN control using deterministic learning technique,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 3, pp. 635–641, Mar. 2014.

[7] S. Lin and A. A. Goldenberg, “Neural-network control of mobile manipulators,” *IEEE Trans. Neural Netw.*, vol. 12, no. 5, pp. 1121–1133, Sep. 2001.

[8] L. Wang, T. Chai, and C. Yang, “Neural-network-based contouring control for robotic manipulators in operational space,” *IEEE Trans. Control Syst. Technol.*, vol. 20, no. 4, pp. 1073–1080, Jul. 2012.

[9] A. Karakasoglu, S. I. Sudharsanan, and M. K. Sundareshan, “Identification and decentralized adaptive control using dynamical neural networks with application to robotic manipulators,” *IEEE Trans. Neural Netw.*, vol. 4, no. 6, pp. 919–930, Nov. 1993.

[10] M. Zhihong, X. H. Yu, K. Eshraghian, and M. Palaniswami, “A robust adaptive sliding mode tracking control using an RBF neural network for robotic manipulators,” in *Proc. IEEE Int. Conf. Neural Netw.*, Nov./Dec. 1995, pp. 2403–2408.

[11] C. Yang, X. Wang, Z. Li, Y. Li, and C.-Y. Su, “Teleoperation control based on combination of wave variable and neural networks,” *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 8, pp. 2125–2136, Aug. 2017.

[12] Z. Cao, Q. Xiao, R. Huang, and M. Zhou, “Robust neuro-optimal control of underactuated snake robots with experience replay,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 1, pp. 208–217, Jan. 2018.

[13] C. S. Chen, “Dynamic structure neural-fuzzy networks for robust adaptive control of robot manipulators,” *IEEE Trans. Ind. Electron.*, vol. 55, no. 9, pp. 3402–3414, Sep. 2008.

[14] R. J. Wai and R. Muthusamy, “Fuzzy-neural-network inherited sliding-mode control for robot manipulator including actuator dynamics,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 2, pp. 274–287, Feb. 2013.

[15] H. D. Patino, R. Carelli, and B. R. Kuchen, “Neural networks for advanced control of robot manipulators,” *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 343–354, Mar. 2002.

[16] M. Yue, L. Wang, and T. Ma, “Neural network based terminal sliding mode control for wmrs affected by an augmented ground friction with slippage effect,” *IEEE/CAA J. Autom. Sinica*, vol. 4, no. 3, pp. 498–506, Jul. 2017.

[17] W. He, A. O. David, Z. Yin, and C. Sun, “Neural network control of a robotic manipulator with input deadzone and output constraint,” *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 6, pp. 759–770, Jun. 2016.

[18] W. He, Z. Yan, C. Sun, and Y. Chen, "Adaptive neural network control of a flapping wing micro aerial vehicle with disturbance observer," *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3452–3465, Oct. 2017.

[19] Z. Chen, Z. Li, and C. L. P. Chen, "Adaptive neural control of uncertain MIMO nonlinear systems with state and input constraints," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 6, pp. 1318–1330, Jun. 2017.

[20] C. Sun, W. He, and J. Hong, "Neural network control of a flexible robotic manipulator using the lumped spring-mass model," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 8, pp. 1863–1874, Aug. 2017.

[21] H. Yang and J. Liu, "An adaptive RBF neural network control method for a class of nonlinear systems," *IEEE/CAA J. Autom. Sinica*, vol. 5, no. 2, pp. 457–462, Mar. 2018.

[22] S. Dutta, P. K. Patchaikani, and L. Behera, "Near-optimal controller for nonlinear continuous-time systems with unknown dynamics using policy iteration," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 7, pp. 1537–1549, Jul. 2016.

[23] K. P. Tee, S. S. Ge, and E. H. Tay, "Barrier Lyapunov functions for the control of output-constrained nonlinear systems," *Automatica*, vol. 45, no. 4, pp. 918–927, Apr. 2009.

[24] K. P. Tee, B. Ren, and S. S. Ge, "Control of nonlinear systems with time-varying output constraints," *Automatica*, vol. 47, no. 11, pp. 2511–2516, Nov. 2011.

[25] B. Ren, S. S. Ge, K. P. Tee, and T. H. Lee, "Adaptive neural control for output feedback nonlinear systems using a barrier Lyapunov function," *IEEE Trans. Neural Netw.*, vol. 21, no. 8, pp. 1339–1345, Aug. 2010.

[26] W. He, H. Huang, and S. S. Ge, "Adaptive neural network control of a robotic manipulator with time-varying output constraints," *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3136–3147, Oct. 2017.

[27] W. He, Y. Dong, and C. Sun, "Adaptive neural impedance control of a robotic manipulator with input saturation," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 3, pp. 334–344, Mar. 2016.

[28] W. He, Y. Chen, and Z. Yin, "Adaptive neural network control of an uncertain robot with full-state constraints," *IEEE Trans. Cybern.*, vol. 46, no. 3, pp. 620–629, Mar. 2016.

[29] C. Yang, Y. Jiang, Z. Li, W. He, and C.-Y. Su, "Neural control of bimanual robots with guaranteed global stability and motion precision," *IEEE Trans. Ind. Informat.*, vol. 13, no. 3, pp. 1162–1171, Jun. 2016.

[30] Y. Zhang, S. Su, A. Savkin, B. Celler, and H. Nguyen, "Multiloop integral controllability analysis for nonlinear multiple-input single-output processes," *Ind. Eng. Chem. Res.*, vol. 56, no. 28, pp. 8054–8065, Jul. 2017.

[31] Q. Guo, Y. Zhang, B. Celler, and S. Su, "Backstepping control of electro-hydraulic system based on extended-state-observer with plant dynamics largely unknown," *IEEE Trans. Ind. Electron.*, vol. 63, no. 11, pp. 6909–6920, Nov. 2016.

[32] C. P. Bechlioulis and G. A. Rovithakis, "Adaptive control with guaranteed transient and steady state tracking error bounds for strict feedback systems," *Automatica*, vol. 45, no. 2, pp. 532–538, 2009.

[33] Z. Zhikai, D. Guangren, and H. Mingzhe, "Longitudinal attitude control of a hypersonic vehicle with angle of attack constraints," in *Proc. 10th Asian Control Conf.*, May/Jun. 2015, pp. 1–6.

[34] M. Tokita, T. Mituoka, T. Fukuda, T. Shibata, and F. Arai, "Position and force hybrid control of robotic manipulator by neural network (adaptive control of 2 DOF manipulators)," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, Nov. 1991, pp. 113–121.

[35] Y. Jiang, C. Yang, J. Na, G. Li, Y. Li, and J. Zhong, "A brief review of neural networks based learning and control and their applications for robots," *Complexity*, 2017.

[36] W. Kim, D. Shin, D. Won, and C. C. Chung, "Disturbance-observer-based position tracking controller in the presence of biased sinusoidal disturbance for electrohydraulic actuators," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 6, pp. 2290–2298, Nov. 2013.

[37] H. Merritt, *Hydraulic Control Systems*. New York, NY, USA: Wiley, 1967.

[38] Q. Guo, T. Yu, and D. Jiang, "Robust H_∞ positional control of 2-DOF robotic arm driven by electro-hydraulic servo system," *ISA Trans.*, vol. 59, pp. 55–64, Nov. 2015.

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

[40] Z. Zhang, G. Duan, and M. Hou, "Robust adaptive dynamic surface control of uncertain non-linear systems with output constraints," *IET Control Theory Appl.*, vol. 11, no. 1, pp. 110–121, Jan. 2017.

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

[42] R. M. Sanner and J.-J. E. Slotine, "Gaussian networks for direct adaptive control," *IEEE Trans. Neural Netw.*, vol. 3, no. 6, pp. 837–863, Nov. 1992.

[43] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.

[44] Z. Wu, *Hydraulic Control System*, Beijing, China: Higher Education Press, 2008.

Qing Guo (M'16) received the B.E. degree in automation and the M.S. and Ph.D. degrees in navigation, guidance, and control from the Harbin Institute of Technology, Harbin, China, in 2003, 2005, and 2008, respectively.

From 2013 to 2014, he was an Academic Visitor with the Center for Power Transmission and Motion Control, Department of Mechanical Engineering, University of Bath, Bath, U.K. He is currently an Associate Professor with the School of Aeronautics and Astronautics, University of Electronic Science and Technology of China, Chengdu, China. His current research interests include robust and adaptive control, electrohydraulics, and exoskeleton and rehabilitation robot.

Dr. Guo is also the member of the Youth Working Committee, Fluid Transmission and Control Branch, Chinese Mechanical Engineering Society.

Yi Zhang (M'16) received the B.E. degree from the Automation College, Chongqing University of China, Chongqing, China, in 2007, the M.S. degree in engineering from the University of Technology Sydney, Ultimo, NSW, Australia, in 2008, the M.S. degree in engineering from The University of Sydney, Sydney, NSW, Australia, in 2009, and the Ph.D. degree from the University of Technology Sydney, in 2014.

He is currently a Lecturer in aeronautics and astronautics from the University of Electronic Science and Technology of China, Chengdu, China. His current research interests include exoskeleton and rehabilitation robot, biomedical system modeling and control, and fault tolerant control.

Dr. Zhang received the Australian Postgraduate Award Scholarship and the Top-up Scholarship in 2011.

Branko G. Celler (M'87–SM'12–F'14) received the B.Sc., B.E.E. (Hons), and Ph.D. degrees from the University of New South Wales (UNSW), Sydney, NSW, Australia, in 1969, 1971, and 1978.

He returned to the UNSW in 1981 and was the Foundation Director of the Biomedical Systems Laboratory, the Human Performance Laboratory, and the Center for Health Informatics. During his career, he has published more than 220 journal papers and refereed conference proceedings. His current research interests include biomedical instrumentation and systems, system modeling and control, noninvasive modeling of cardiovascular performance, home telecare and remote monitoring, and medical informatics.

Dr. Celler was a recipient of the Post-Doctoral Fellowship at Johns Hopkins University, Baltimore, MD, USA.

Steven W. Su (M'99–SM'17) received the B.S. and M.S. degrees from the Harbin Institute of Technology, Harbin, China, in 1990 and 1993, respectively, and the Ph.D. degree from the Research School of Information Sciences and Engineering, Australian National University, Canberra, ACT, Australia, in 2002.

He was a Post-Doctoral Research Fellow with the Faculty of Engineering, University of New South Wales, Sydney, NSW, Australia, from 2002 to 2006. He is currently an Associate Professor with the Faculty of Engineering and IT, University of Technology Sydney, Ultimo, NSW, Australia. His current research interests include biomedical system modeling and control, nonlinear robust control, fault tolerant control, and wearable monitoring systems.