

Research Article

Dynamic Learning from Adaptive Neural Control of Uncertain Robots with Guaranteed Full-State Tracking Precision

Min Wang, Yanwen Zhang, and Huiping Ye

School of Automation Science and Engineering, Guangzhou Key Laboratory of Brain Computer Interaction and Applications, South China University of Technology, Guangzhou 510641, China

Correspondence should be addressed to Min Wang; auwangmin@scut.edu.cn

Received 21 March 2017; Accepted 30 April 2017; Published 14 August 2017

Academic Editor: Yanan Li

Copyright © 2017 Min Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A dynamic learning method is developed for an uncertain n -link robot with unknown system dynamics, achieving predefined performance attributes on the link angular position and velocity tracking errors. For a known nonsingular initial robotic condition, performance functions and unconstrained transformation errors are employed to prevent the violation of the full-state tracking error constraints. By combining two independent Lyapunov functions and radial basis function (RBF) neural network (NN) approximator, a novel and simple adaptive neural control scheme is proposed for the dynamics of the unconstrained transformation errors, which guarantees uniformly ultimate boundedness of all the signals in the closed-loop system. In the steady-state control process, RBF NNs are verified to satisfy the partial persistent excitation (PE) condition. Subsequently, an appropriate state transformation is adopted to achieve the accurate convergence of neural weight estimates. The corresponding experienced knowledge on unknown robotic dynamics is stored in NNs with constant neural weight values. Using the stored knowledge, a static neural learning controller is developed to improve the full-state tracking performance. A comparative simulation study on a 2-link robot illustrates the effectiveness of the proposed scheme.

1. Introduction

In the past decades, the force/position tracking control problem of robots has attracted wide attention in both theory and applications [1, 2]. In the early stage of robotic tracking control, the system model is usually assumed to be accurately known, and the corresponding model-based control method has been proposed in [3, 4]. Along with the diversity of robot working environment and the complexity of the robot's structure, the force/position tracking control has been studied in different kinds of uncertainties. Nowadays, ignoring uncertainties to simplify control design may cause the large steady-state errors or/and poor transient response [5]. For the case of parametric uncertainties, adaptive control method has been presented in [6, 7] to make robots adapt the changing control environment. In order to enhance system robustness on the uncertain parameters in the presence of external disturbances, sliding mode control [8, 9] has been proposed to obtain the desired robotic tracking control performance. Owing to the universal approximation property [10–17],

a great number of intelligent control schemes, such as adaptive neural/fuzzy control, have been developed for controlling robotic systems with uncertain nonlinearities [18–22].

Although intelligent control of robotic systems has attracted considerable attention in the past few years, relatively few robot control methods could achieve human-like performance in a dynamic and uncertain environment. It is well known that intelligent control was initially inspired by the learning and control abilities of human beings, thus intelligent control should at least possess the aforementioned properties “learning by doing” and “doing with the learned knowledge” [23–26]. But most existing intelligent control schemes can only ensure the stability of closed-loop systems without being able to achieve the information acquisition and storage and utilization of unknown system dynamics. This means that the existing intelligent control schemes do not solve the accurate convergence of estimated parameters, which usually needs to guarantee the exponential stability of the derived closed-loop system. However, it is extremely difficult for uncertain nonlinear systems to verify

the exponential stability of the derived closed-loop system. Recently, a deterministic learning method was proposed in [27] using RBF NNs for a second-order Brunovsky system, where the derived closed-loop system was described by a class of linear time-varying systems. By verifying that RBF NNs satisfied persistent excitation (PE) condition, the convergence of partial neural weights and accurate neural approximation of unknown system dynamics were guaranteed in [27] because of the exponential convergence of the derived linear time-varying closed-loop systems. The deterministic learning method was further extended to n th-order Brunovsky systems with an unknown affine/nonaffine term [28, 29], where the derivative of unknown affine terms was assumed to be bounded. By combining backstepping with a system decomposition strategy, an elegant dynamic learning method was proposed to cope with the learning and control problem of third-order strict-feedback systems [30]. By combining dynamic surface control technology [31], the result in [30] was further extended to n th-order strict-feedback systems. The deterministic learning method was also applied in many physical systems such as marine surface vessels [32, 33] and robot manipulators [34].

Recently, the deterministic learning methods are mainly used to solve the learning and control problem for single input single output nonlinear systems without any constraint. In practice, there are many different kinds of constraints in most of physical systems, such as output or state constraints, tracking performance constraints. The violation of the constraints may cause severe performance degradation, safety problem, or system damage [35]. Therefore, it is of great importance for solving the control and learning problem of constrained systems. Based on Lyapunov theorem, a barrier Lyapunov function (BLF) method has been presented to solve output constraints for strict-feedback nonlinear systems [36], output feedback nonlinear systems [37], flexible systems [38], and robotic manipulator [35, 39]. The BLF-based methods were also extended to solve state constraints [40–42]. Although the aforementioned results on the output or state constraints can guarantee that system outputs or states converge to a predefined bounded set, the predefined performance requirements on the convergence rate, maximum overshoot, and steady-state error have not been studied fully. The predefined performance issue is an extremely challenging problem. Recently, adaptive neural prescribed performance controller was proposed in [43, 44] for feedback linearizable nonlinear systems by means of transformation functions. The proposed method was also developed to deal with the constrained output tracking control problem in many applications such as robotic systems [45], nonlinear servo mechanisms [46], marine surface vessels [33], nonlinear stochastic large-scale systems with actuator faults [47], and switched nonlinear systems [48]. To solve partial tracking error constraints, a fuzzy dynamic surface control design was developed in [49, 50] for a class of strict-feedback nonlinear systems by transforming the state tracking errors into new virtual variables. However, the existing control schemes, such as [35–51], can only guarantee the stability of closed-loop systems with different constraints, which are not capable of achieving the learning of unknown system dynamics. The

main reason is that the derived closed-loop error system is extremely complex, such that its exponential convergence is difficult to be verified using the existing stability analysis tools. To solve this problem, a neural learning control with the output tracking error constraint was presented in [52, 53] for a class of nonlinear systems. These methods proposed in [52, 53] cannot be adopted to deal with the dynamic learning problem for multi-input and multioutput nonlinear systems with full-state tracking error constraints.

Based on the above discussions, this paper proposes a novel dynamic learning method for a multi-input and multi-output n -link robot with full-state tracking error constraints and a mild assumption. To prevent the violation of the full-state tracking error constraints, performance functions are firstly introduced to characterize the transient and steady-state performance of full-state tracking errors. And then, using a nonlinear transformation method, the constrained tracking control problem is effectively transformed into the stabilization problem of equivalent unconstrained transformation error systems. By combining backstepping and Lyapunov stability, a novel adaptive neural control scheme is proposed to guarantee the uniformly ultimate boundedness of all closed-loop signals and the prescribed full-state tracking error performance. It should be pointed out that the proposed control scheme is different from the traditional backstepping design. In our control design, the correlative interconnection term can not be compensated in next step of backstepping design because of the full-state tracking error constraints. To overcome the difficulty, two independent Lyapunov functions are constructed in each step of backstepping design. Based on the independent Lyapunov functions, the unconstrained transformation error of the link angular velocity tracking error is firstly proved to be bounded, and the corresponding link angular velocity tracking error is further proved to satisfy the prescribed performance. Invoking the boundedness of the link angular velocity tracking error, we backward derive the link angular position tracking error to satisfy the prescribed performance. By means of the tracking convergence in the steady-state control process, the regression subvector consisting of the RBFs along the recurrent tracking orbit satisfies the partial PE condition. Subsequently, an appropriate state transformation is introduced to transform the closed-loop system into a linear time-varying (LTV) system with small perturbed terms. With the perturbation theory of LTV system, the knowledge of the closed-loop robotic dynamics can be accurately stored by RBF NNs with constant weight values. The stored knowledge can be reused to develop a static neural learning controller for achieving the better control performance with smaller transient-state tracking errors, smaller control gains, and less computational time.

The rest of this paper is organized as follows. Section 2 introduces the system formulation, the full-state tracking error transformation, and some useful preliminaries about RBF networks. In Section 3, a novel adaptive neural control design is proposed for rigid robotic manipulators with constrained full-state tracking performances. Neural learning control is developed in Section 4, which achieves the knowledge acquisition, storage, and utilization of unknown robotic dynamics. In Section 5, simulation studies on 2-link robotic

system are given to show the effectiveness of the proposed method. Section 6 includes the conclusions of the paper.

2. Problem Formulation and Preliminaries

2.1. Uncertain Robotic System. The dynamics of an n -link robotic system is described in the following Lagrange form [1]:

$$M(q)\ddot{q} + V_m(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) + d(t) = \tau, \quad (1)$$

where $q, \dot{q}, \ddot{q} \in R^n$ are the angular position, velocity, and acceleration, respectively, $\tau \in R^n$ refers to the input torque, $M(q) \in R^{n \times n}$ is an unknown symmetric positive definite inertia matrix, $V_m(q, \dot{q}) \in R^{n \times n}$ denotes unknown centripetal and Coriolis torques, $G(q) \in R^n$ is an unknown gravitational force vector, $F(\dot{q}) \in R^n$ is an unknown friction vector, and $d(t) \in R^n$ denotes unknown disturbances,

Property 1 (see [54]). The matrix $\dot{M}(q) - 2V_m(q, \dot{q})$ is skew-symmetric.

Assumption 1. The unknown external disturbance $d(t)$ is bounded, that is, there exists a constant $d^* \in R^+$ such that $\|d(t)\| < d^*$.

In this paper, we choose $y_d = x_{d1} \in R^n$ as a recurrent reference trajectory of the angular position q , which is generated by the following reference model:

$$\begin{aligned} \dot{x}_{d1} &= x_{d2}, \\ \dot{x}_{d2} &= F_d(x_d), \end{aligned} \quad (2)$$

where $x_d = [x_{d1}^T, x_{d2}^T]^T \in R^{2n}$ is the state vector of the reference model, which is assumed to be recurrent signals, and $F_d(x_d)$ is a known smooth function. The reference orbit along the given initial condition $x_d(0)$ is defined as $\varphi_d(x_d(0))$. In this paper, $\varphi_d(x_d(0))$ is assumed to be a recurrent motion.

2.2. Full-State Tracking Constraints. Define full-state tracking errors as $e_1 = q - y_d = [e_{11}, e_{12}, \dots, e_{1n}]^T \in R^n$ and $e_2 = \dot{q} - \dot{y}_d = [e_{21}, e_{22}, \dots, e_{2n}]^T \in R^n$, where $\alpha_1 \in R^n$ are continuously differentiable virtual control signals. For guaranteeing the prescribed transient and steady-state bounds of full-state tracking errors, e_1 and e_2 need to satisfy the following predefined condition:

$$-\bar{\delta}_{ij}\rho_i(t) < e_{ij} < \bar{\delta}_{ij}\rho_i(t), \quad (3)$$

where $i = 1, 2, j = 1, 2, \dots, n$, and $\bar{\delta}_{ij}, \bar{\delta}_{ij}$ are positive design constants and $\rho_i(t)$ is a smooth, strictly positive, bounded, and decreasing function, which satisfies $\lim_{t \rightarrow \infty} \rho_i(t) > 0$ and is called performance function. In this paper, $\rho_i(t)$ is chosen as the following exponential performance function:

$$\rho_i(t) = (\rho_{i0} - \rho_{i\infty})e^{-\kappa_i t} + \rho_{i\infty}, \quad \forall t \geq 0, \quad (4)$$

where $\rho_{i0}, \rho_{i\infty}$ and κ_i are positive design constants.

For any given initial condition $e_i(0)$, these design constants $\rho_{i0}, \bar{\delta}_{ij}$ and $\bar{\delta}_{ij}$ can be chosen appropriately such that $e_i(0)$ satisfies the predefined condition (3). From (3) and (4), the different selection of these design parameters, $\rho_{i0}, \rho_{i\infty}, \kappa_i, \bar{\delta}_{ij}$ and $\bar{\delta}_{ij}$, can obtain different performance requirements on the tracking error e_{ij} .

2.3. RBF Neural Networks. The RBF NNs can be described by $f_{nn}(Z) = W^T S(Z)$, where $Z \in \Omega_Z \subset R^q$ are NN input variables with $q > 1$ being NN input dimension and Ω_Z being a compact set, $W = [w_1, \dots, w_N]^T \in R^N$ is the weight vector, $N > 1$ is the NN node number, and $S(Z) = [s_1(\|Z - \mu_1\|), \dots, s_N(\|Z - \mu_N\|)]^T$ is the radial basis function vector. The RBF neural networks have the following desired properties.

(1) Universal Approximation

Lemma 2 (see [55]). Using sufficiently large node number N , the RBF NNs $W^T S(Z)$ can approximate any smooth function $f(Z) : \Omega_Z \rightarrow R^q$ over a compact set Ω_Z to any arbitrary accuracy as

$$f(Z) = W^{*T} S(Z) + \eta(Z), \quad (5)$$

where W^* is the ideal weight vector of W and $\eta(Z)$ is any small approximation error which satisfies $|\eta(Z)| \leq \eta^*$.

(2) Spatially Localized Approximation Property. It should be noted that the radial basic function satisfies $s_i(\|Z - \mu_i\|) \rightarrow 0$ when $\|Z - \mu_i\| \rightarrow \infty$. This property shows that the network output is only locally affected by each basis function. Therefore, any smooth function $f(Z)$ over a compact set Ω_Z can be approximated using a limited number of neurons, which are located in a local region along bounded input trajectory Z :

$$f(Z) = W_\zeta^{*T} S_\zeta(Z) + \eta_\zeta(Z), \quad (6)$$

where $S_\zeta(Z) \in R^{N_\zeta}$, $N_\zeta < N$ is the subvector of $S(Z)$, composed of RBFs that are close to the trajectory Z , $W_\zeta^* \in R^{N_\zeta}$ is the corresponding subvector of W^* , and approximation errors $\eta_\zeta(Z) = \eta(Z) - W_\zeta^{*T} S_\zeta(Z)$, $\bar{\zeta}$ are defined as the region far away from the trajectory Z , which means $\|S_{\bar{\zeta}}(Z)\| \rightarrow 0$. Therefore, $\eta_\zeta(Z)$ is close to $\eta(Z)$, which is any small value.

(3) Partial PE Condition. Persistent excitation plays a key role in accurate convergence of neural weight estimator. To clearly show that RBF NNs satisfy the PE property, a definition of PE condition is given as follows:

Definition 3 (see [55]). A continuous, uniformly bounded, vector-valued function $S : [0, \infty] \rightarrow R^m$ is said to satisfy the persistent excitation condition, if there exist positive constants α_1, α_2 , and δ such that

$$\alpha_1 \|c\|^2 \leq \int_{t_0}^{t_0 + \sigma} |S(\tau)^T c|^2 d\mu(\tau) \leq \alpha_2 \|c\|^2, \quad \forall t_0 \geq 0, \quad (7)$$

holds for every constant vector $c \in \mathbb{R}^N$, where μ is a positive, Σ -finite Borel measure on $[0, \infty]$.

Lemma 4 (see [27]). *Consider any recurrent orbit $Z \in \mathbb{R}^q$, remaining in a compact set Ω_Z with $\Omega_Z \subset \mathbb{R}^q$. Then, for the RBF network $W^T S(Z)$ with centers placed on a regular lattice, which is large enough to cover the compact set Ω_Z , the regressor subvector $S_\tau(Z)$ in (6) (rather than the entire regressor vector $S(Z)$) is persistently exciting.*

3. Adaptive Neural Control with Full-State Tracking Error Constraints

In this section, a novel stable adaptive neural tracking control scheme will be developed for the system (1) with full-state tracking error constraints (3) using nonlinear error transformations, independent Lyapunov functions, and backstepping. The proposed control scheme will guarantee that all the signals in the closed-loop system are ultimately bounded and the full-state tracking errors satisfy the prescribed performances (3).

In order to transform the constrained tracking control problem (3) into an equivalent unconstrained one, a performance transformation method is introduced as follows:

$$e_{ij} = \rho_i(t) T_{ij}(\varsigma_{ij}), \quad (8)$$

where $T_{ij}(\varsigma_{ij})$ is a smooth and strictly increasing function and satisfies $T_{ij}(\cdot) : (-\infty, \infty) \rightarrow (-\bar{\delta}_{ij}, \bar{\delta}_{ij})$. In this paper, the transformation function $T_{ij}(\varsigma_{ij})$ is constructed as follows:

$$T_{ij}(\varsigma_{ij}) = \frac{\bar{\delta}_{ij} e^{\varsigma_{ij}} - \underline{\delta}_{ij} e^{-\varsigma_{ij}}}{e^{\varsigma_{ij}} + e^{-\varsigma_{ij}}}. \quad (9)$$

Owing to the properties $T_{ij}(\varsigma_{ij})$ and $\rho_i(t) \geq \rho_{i\infty} > 0$, the inverse transformation of $T_{ij}(\varsigma_{ij})$ exists and is well defined as follows:

$$\varsigma_{ij} = T_{ij}^{-1} \left[\frac{e_{ij}}{\rho_i(t)} \right]. \quad (10)$$

From (3) and (10), if ς_{ij} is verified to be bounded, then we can obtain that e_{ij} satisfies the predefined error performance (3). Differentiating ς_{ij} yields

$$\dot{\varsigma}_{ij} = r_{ij} \left[\dot{e}_{ij} - \frac{\dot{\rho}_i(t)}{\rho_i(t)} e_{ij} \right], \quad (11)$$

where $r_{ij} = \partial T_{ij}^{-1}[e_{ij}/\rho_i(t)] / [\rho_i(t) \partial(e_{ij}/\rho_i(t))]$. Next, define the transformed unconstrained error vector $\xi_i = [\varsigma_{i1}, \varsigma_{i2}, \dots, \varsigma_{in}]^T \in \mathbb{R}^n$, $i = 1, 2$. It follows from (11) that

$$\dot{\xi}_i = \Upsilon_i (\dot{e}_i - \Psi_i e_i), \quad (12)$$

where $\Psi_i = \dot{\rho}_i(t)/\rho_i(t)$ and $\Upsilon_i = \text{diag}\{r_{i1}, r_{i2}, \dots, r_{in}\} \in \mathbb{R}^{n \times n}$. Notice that $e_1 = q - y_d$ and $e_2 = \dot{q} - \alpha_1$, using system (1), and thus the dynamics of the transformed error vector can be rewritten as

$$\dot{\xi}_1 = \Upsilon_1 (e_2 + \alpha_1 - x_{d2} - \Psi_1 e_1), \quad (13)$$

$$\dot{\xi}_2 = \Upsilon_2 M^{-1}(q) [F(q, \dot{q}, \vartheta) + \tau - d(t)], \quad (14)$$

where $F(q, \dot{q}, \vartheta) = -V_m(q, \dot{q})\dot{q} - G(q) - F(\dot{q}) - M(q)\vartheta$ is an unknown smooth function vector and $\vartheta = \ddot{\alpha}_1 + \Psi_2 e_2$ is a computable variable. Next, a control scheme will be developed for the system (13)-(14) based on backstepping.

Step 1. For the transformed error subsystem (13), design a virtual control

$$\alpha_1 = -k_1 \xi_1 + x_{d2} + \Psi_1 e_1. \quad (15)$$

Then, we have

$$\dot{\xi}_1 = \Upsilon_1 (e_2 - k_1 \xi_1), \quad (16)$$

where $k_1 = k_{10} + k_{11}$ with $k_{10} > 0$ and $k_{11} > 0$.

Constructing the following Lyapunov function candidate:

$$V_1 = \frac{\xi_1^T \xi_1}{2} \quad (17)$$

whose derivative along (16) yields

$$\dot{V}_1 = -k_1 \xi_1^T \Upsilon_1 \xi_1 + \xi_1^T \Upsilon_1 e_2. \quad (18)$$

Remark 5. From (18), the boundedness of the transformation error ξ_1 depends on the boundedness of the link angular velocity tracking error e_2 . In the traditional backstepping design, the term $\xi_1^T \Upsilon_1 e_2$ is called the correlative interconnection term, which can be usually compensated in the next step of backstepping. However, in our control design, it is impossible to deal with $\xi_1^T \Upsilon_1 e_2$ in Step 2. The main reason is that the full-state tracking error constraints are considered in this paper, which derives the transformation error ξ_2 in Step 2, instead of the traditional error e_2 . Therefore, it is difficult to construct the Lyapunov V_2 with ξ_2 to compensate for $\xi_1^T \Upsilon_1 e_2$; see Step 2 for details.

Step 2. By adding and subtracting $\Upsilon_2^{-1} V_m(q, \dot{q}) \xi_2$, the transformed error subsystem (14) can be rewritten as

$$\dot{\xi}_2 = \Upsilon_2 M^{-1} [\Phi(Z) + \tau - d(t) - \Upsilon_2^{-1} V_m(q, \dot{q}) \xi_2], \quad (19)$$

where $Z = [q^T, \dot{q}^T, \xi_2^T, \vartheta^T]^T$, and

$$\Phi(Z) = F(q, \dot{q}, \vartheta) + \Upsilon_2^{-1} V_m(q, \dot{q}) \xi_2. \quad (20)$$

It should be pointed out that the term $\Upsilon_2^{-1} V_m(q, \dot{q}) \xi_2$ introduced in (19) facilitates the stability analysis based on the property (1). Since $F(q, \dot{q})$, M , and $V_m(q, \dot{q})$ are unknown smooth function vectors, $\Phi(Z)$ is also unknown and smooth, which can not directly be used to construct the controller. To solve this problem, the unknown dynamics $\Phi(Z)$ are approximated by RBF NN (5) in this paper. Then, we have

$$\dot{\xi}_2 = \Upsilon_2 M^{-1} [W^{*T} S(Z) + \eta(Z) + \tau - \Upsilon_2^{-1} V_m(q, \dot{q}) \xi_2 - d(t)], \quad (21)$$

where the approximation error $\eta(Z) \in R^n$ satisfies $\|\eta(Z)\| \leq \eta^*$ and $W^* = [W_1^*, \dots, W_n^*] \in R^{N \times n}$ is the unknown optimal NN weight vector with NN node number $N > 1$, and $S(Z) = [s_1(Z), s_2(Z), \dots, s_N(Z)]^T \in R^N$ is a radial basis function vector. Define \widehat{W} as the estimated weight values of W^* , and let $\widetilde{W} = \widehat{W} - W^*$ be the corresponding estimated error. Then, design the adaptive neural control law as

$$\tau = -k_{20}Y_2^{-1}\xi_2 - k_{21}Y_2\xi_2 - \widehat{W}^T S(Z) \quad (22)$$

and construct the neural weight updated law as

$$\dot{\widehat{W}} = \dot{\widetilde{W}} = \Gamma S(Z) (Y_2 \xi_2)^T - \sigma \Gamma \widehat{W}, \quad (23)$$

where k_{20} and k_{21} are positive design constants, $\Gamma > 0$ is positive diagonal matrix, and $\sigma > 0$ is a small value, which is used to improve the robustness of the adaptive controller (22).

Substituting (22) into (21), we have

$$\begin{aligned} \dot{\xi}_2 &= Y_2 M^{-1} [-k_{20}Y_2^{-1}\xi_2 - k_{21}Y_2\xi_2 - \widehat{W}^T S(Z) \\ &\quad - Y_2^{-1}V_m(q, \dot{q})\xi_2 + \bar{\eta}], \end{aligned} \quad (24)$$

where $\bar{\eta} = \eta(Z) - d(t)$. Noting that the inertia matrix M is positive definite, thus we construct the following Lyapunov function candidate:

$$V_2 = \frac{1}{2}\xi_2^T M \xi_2 + \frac{1}{2}\text{tr}[\widetilde{W}^T \Gamma^{-1} \widetilde{W}]. \quad (25)$$

In what follows, one of our main results will be shown by the following theorem.

Theorem 6 (stability and tracking). *Consider the closed-loop system consisting of the robotic system (1), the bounded reference trajectory (2), the full-state tracking performance condition (3), the transformed error (10), the proposed adaptive neural control law (22) with the virtual control law (15), and the weight updated law (23). Assume the given bounded initial conditions satisfy the condition (3) (this condition can be satisfied by choosing proper designed parameters $\underline{\delta}_{ij}$, $\bar{\delta}_{ij}$, and ρ_{i0}). Then, we have that*

- (1) all signals of the closed-loop system remain uniformly ultimately bounded
- (2) the constrained full-state tracking errors e_{ij} satisfy the prescribed performance (3) and they converge to a small residual set of zero in a finite time T .

Proof. Differentiating V_2 in (25) with respect to time and using (18), (24) yields

$$\begin{aligned} \dot{V}_2 &= \xi_2^T Y_2 [-k_{20}Y_2^{-1}\xi_2 - k_{21}Y_2\xi_2 - \widehat{W}^T S(Z) + \bar{\eta}] \\ &\quad + \frac{1}{2}\xi_2^T (\dot{M} - 2V_m(q, \dot{q}))\xi_2 + \text{tr}[\widetilde{W}^T \Gamma^{-1} \dot{\widetilde{W}}]. \end{aligned} \quad (26)$$

Using the Property 1 and substituting (23) into (26), the derivation of V_2 is rewritten as

$$\begin{aligned} \dot{V}_2 &= \text{tr}[-\sigma \widetilde{W}^T \widetilde{W}] \\ &\quad + \xi_2^T Y_2 [-k_{20}Y_2^{-1}\xi_2 - k_{21}Y_2\xi_2 + \bar{\eta}]. \end{aligned} \quad (27)$$

Subsequently, using the appropriate inequality and combining Assumption 1, we have

$$\xi_2^T Y_2 \bar{\eta} \leq k_{21}\xi_2^T Y_2 \xi_2 + \frac{\bar{\eta}^{*2}}{4k_{21}}, \quad (28)$$

$$\text{tr}[-\sigma \widetilde{W}^T \widetilde{W}] \leq -\frac{\sigma}{2}\|\widetilde{W}\|_F^2 + \frac{\sigma}{2}\|W^*\|_F^2,$$

where $\bar{\eta}^{*2} = \eta^* + d^*$ is a positive constant. Substituting (28) into (27) gives

$$\begin{aligned} \dot{V}_2 &\leq -k_{20}\xi_2^T \xi_2 - \frac{\sigma}{2}\|\widetilde{W}\|_F^2 + \frac{\sigma}{2}\|W^*\|_F^2 + \frac{\bar{\eta}^{*2}}{4k_{21}} \\ &\leq -a_2 V + b_2, \end{aligned} \quad (29)$$

where $a_2 = \min\{k_{20}/\lambda_{\max}(M), \sigma\lambda_{\min}(\Gamma)\}$, $b_2 = (\sigma/2)\|W^*\|_F^2 + \bar{\eta}^{*2}/4k_{21}$. Let $\beta_2 = b_2/a_2$; we have

$$V_2 \leq (V_2(0) - \beta_2) \exp(-a_2 t) + \beta_2. \quad (30)$$

□

From the definition (25) of V_2 , the transformed error ξ_2 and the neural weight error \widetilde{W} are bounded. Noticing that $\xi_2 = [\xi_{21}, \xi_{22}, \dots, \xi_{2n}]^T \in R^n$, and the error transformed relationship (8)–(10), thus we obtain that the tracking error e_2 is bounded and satisfies the prescribed tracking error constraints (3). This means $\|e_2\| \leq \delta^*$, δ^* is any small value depending on design parameters $\underline{\delta}_{ij}$, $\bar{\delta}_{ij}$, ρ_{20} , and $\rho_{2\infty}$. Using the bounded property of e_2 , we can obtain that

$$\xi_1^T Y_1 e_2 \leq k_{11}\xi_1^T Y_1 \xi_1 + \frac{\max(r_{i1})\delta^{*2}}{4k_{11}}. \quad (31)$$

Noting $k_1 = k_{10} + k_{11}$, thus (18) can be rewritten as

$$\dot{V}_1 \leq -k_{10}\xi_1^T Y_1 \xi_1 + \frac{\max(r_{i1})\delta^{*2}}{4k_{11}} \leq -a_1 V_1 + b_1, \quad (32)$$

where $a_1 = \min\{2k_{10}/\lambda_{\max}(Y)\}$, $b_1 = \max(r_{i1})\delta^{*2}/4k_{11}$. Further, let $\beta_1 = b_1/a_1$; we have

$$V_1 \leq (V_1(0) - \beta_1) \exp(-a_1 t) + \beta_1. \quad (33)$$

Since $V_1 = \xi_1^T \xi_1/2$, it follows from (33) that ξ_1 is bounded. Similarly, we can obtain that the tracking error e_1 is bounded and satisfies the prescribed tracking error constraints (3). By combining the boundedness of x_{d1} , x_{d2} and the deigned bounded function Y , Ψ_1 , it can be verified that q and α_1 are bounded. Because of $e_2 = \dot{q} - \alpha_1$, \dot{q} is also bounded. Because the desired weight W^* is bounded, the weight estimate \widehat{W} is also bounded. From (22), we can verify that τ is also bounded. Hence, all closed-loop signals are uniformly ultimately bounded.

Moreover, from (17), (25), (30), and (33), we can obtain the convergence region of the transformed error ξ_1 , ξ_2 as follows:

$$\begin{aligned} \sum_{i=1}^n \varsigma_{1i}^2 &\leq 2\beta_1 + 2V_1(0) \exp(-a_1 t), \\ \sum_{i=1}^n \varsigma_{2i}^2 &\leq \frac{2\beta_2}{\lambda_{\min}(M)} + \frac{2V_2(0) \exp(-a_2 t)}{\lambda_{\min}(M)}. \end{aligned} \quad (34)$$

By choosing $\varphi_1 > \sqrt{2\beta_1}$ and $\varphi_2 > \sqrt{2\beta_2/\lambda_{\min}(M)}$, there exists a finite time T , such that for $\forall t > T$ the transformed error ς_{ij} satisfies

$$|\varsigma_{ij}| < \varphi_i, \quad i = 1, 2, \quad j = 1, 2, \dots, n. \quad (35)$$

Subsequently, the convergence region after a finite time T can be constructed as follows:

$$\Omega_\varsigma = \{\varsigma_{ij} \mid |\varsigma_{ij}| < \varphi_i, X_d \in \Omega_d\}, \quad (36)$$

where φ can be adjusted to be arbitrarily small if we choose appropriate controller parameters k_{i0} , k_{i1} , and σ .

From (36), the transformed error ς_{ij} converges to a small residual set Ω_ς in a finite time T . Owing to the convergence of ς_{ij} and the error transformed relationship (8)–(10), the full-state constrained tracking errors $e_{ij}(t)$ satisfy (3) in a finite time T . From (3) and (4), the tracking error e_i exponentially converges to the interval $[-\underline{\delta}_{ij}\rho_{i\infty}, \bar{\delta}_{ij}\rho_{i\infty}]$, which can be adjusted to be a small residual set of zero by choosing the appropriate design parameters $\rho_{i\infty}$, $\underline{\delta}_{ij}$, $\bar{\delta}_{ij}$.

Remark 7. In order to verify the boundedness of transformation errors ξ_1 and ξ_2 , two independent Lyapunov functions V_1 and V_2 are constructed in Steps 1 and 2. Using the appropriate inequality technology, we firstly prove the transformation error ξ_2 is bounded. Subsequently, the boundedness of the tracking error e_2 is indirectly verified based on the error transformed relationship (8)–(10). As indicated by the Remark 5, the boundedness of e_2 backward derives the boundedness of the transformation error ξ_1 . Similarly, e_1 can be verified to satisfy the predefined tracking performance (3).

Remark 8. In order to verify the boundedness of transformation errors ξ_1 and ξ_2 , two independent Lyapunov functions V_1 and V_2 are constructed in Steps 1 and 2. Using the appropriate inequality technology, we firstly prove the transformation error ξ_2 is bounded. Although the boundedness of ξ_2 depends on W^* in (30) which may be large, the tracking error e_2 can still satisfy the prescribed performance (3) based on the error transformed relationship (8)–(10). Subsequently, the boundedness of e_2 backward derives the boundedness of the transformation error ξ_1 . Similarly, e_1 can be verified to satisfy the predefined tracking performance (3).

4. Neural Learning Control

Based on the stable adaptive neural control scheme developed in Section 3, this section will use the spatially localized

approximation ability of RBF NNs to achieve the knowledge acquisition and storage of the unknown system dynamics $\Phi(Z)$. And then, the stored knowledge will be reused to develop a neural learning controller so that the improved control performance of the robotic system (1) can be achieved for the same or a similar control task.

4.1. Knowledge Acquisition, Expression, and Storage. In this section, the regression subvector $S_\zeta(Z)$ of RBF NN is firstly verified to satisfy the PE condition, which is key to achieve the exponential convergence of NN weight values \widehat{W} and the accurate NN approximation of the unknown system dynamics $\Phi(Z)$. From Lemma 4, the NN input vector $Z = [q^T, \dot{q}^T, \xi_2^T, \vartheta^T]^T$ needs to be verified as recurrent signals, so that the regression subvector $S_\zeta(Z)$ along the input orbit Z satisfies partial PE condition. Based on Theorem 6, it can be obtained that the system output q converges to a small neighborhood of y_d for $\forall t > T$. Since y_d is a recurrent signal, q is also recurrent. Since $\dot{q} = \alpha_1 + e_2$, $\alpha_1 = -k_1\xi_1 + x_{d2} + \Psi_1 e_1$, e_1 , and ξ_1 are very small values, \dot{q} is recurrent with the same period as x_{d2} . In the steady-state control, ξ_2 is small and recurrent. Noting that $\vartheta = \dot{\alpha}_1 + \Psi_2 e_2$, which is a function of the variables e_i , ξ_i , x_i , and x_{di} , so it can be recurrently verified as a recurrent signal. According to Lemma 4, the regression subvector $S_\zeta(Z)$ satisfies partial PE condition.

Using the spatially localized approximation ability of RBF NNs, the closed-loop system from (23) and (24) can be given by

$$\begin{aligned} \dot{\xi}_2 &= G(Y_2, q) \left[-K_2(t) \xi_2 - \widehat{W}_\zeta^T S_\zeta(Z) + \bar{\eta}_\zeta \right], \\ \dot{\widehat{W}}_\zeta &= \Gamma_\zeta S_\zeta(Z) (Y_2 \xi_2)^T - \sigma \Gamma_\zeta \widehat{W}_\zeta, \\ \dot{\widehat{W}}_{\bar{\zeta}} &= \Gamma_{\bar{\zeta}} S_{\bar{\zeta}}(Z) (Y_2 \xi_2)^T - \sigma \Gamma_{\bar{\zeta}} \widehat{W}_{\bar{\zeta}}, \end{aligned} \quad (37)$$

where $K_2(t) = k_{20}Y_2^{-1} + k_{21}Y_2 + Y_2^{-1}V_m(q, \dot{q})$, $G(Y_2, q) = Y_2 M^{-1}$, which is a positive definite and symmetric matrix, $S_\zeta(Z)$ is the subvector of $S(Z)$, which is composed of RBFs that are close to the reference orbit Z , $\widehat{W}_\zeta = [\widehat{W}_{\zeta 1}, \widehat{W}_{\zeta 2}, \dots, \widehat{W}_{\zeta n}] \in R^{\zeta \times n}$ is the corresponding estimated weight subvector with $\widehat{W}_{\zeta i} \in R^\zeta$ and $0 < \zeta < N$, $\bar{\zeta}$ denotes the region far away from the orbit Z , and $\bar{\eta}_\zeta = \bar{\eta} - \widehat{W}_{\bar{\zeta}}^T S_{\bar{\zeta}}(Z)$ are the approximate errors along the reference orbit.

Theorem 9. Consider the closed-loop system consisting of the robotic system (1) with Assumption 1, the bounded reference model (2), full-state tracking error constrained condition (3), the transformation error (10) and adaptive neural control law (22) with the virtual control law (15), and the NN updated law (23). Assume the given bounded initial conditions satisfy (3) and $\widehat{W}_i(0) = 0$, $1 \leq i \leq n$. Then, for any recurrent orbit $\phi_d(x_d(t))|t \geq 0$, we have that the NN weight estimates \widehat{W}_i exponentially converge to a small neighborhood of optimal values W_i^* after $t \geq T$, and the corresponding system dynamics

$\Phi_i(Z)$ along recurrent signals Z is approximated by the stored neural knowledge $\bar{W}_i^T S(Z)$ as:

$$\Phi_i(Z) = \bar{W}_i^T S(Z) + \eta_{i\bar{W}}(Z), \quad (38)$$

where $\eta_{i\bar{W}}(Z)$ approaches the desired error η_i^* , and the constant weight values \bar{W}_i are chosen as

$$\bar{W}_i = \text{mean}_{t \in [t_{ai}, t_{bi}]} \widehat{W}_i(t) = \frac{1}{t_{bi} - t_{ai}} \int_{t_{ai}}^{t_{bi}} \widehat{W}_i(s) ds \quad (39)$$

with $[t_{ai}, t_{bi}]$, $1 \leq i \leq n$, $t_{bi} > t_{ai} > T$ representing a time segment in a steady-state stage.

Proof. Up to now, we have verified that $S_\zeta(Z)$ satisfies the PE condition. Furthermore, in order to achieve the exponential convergence of neural weight estimates \widehat{W}_i , the closed-loop system (37) needs to be transformed into a class of linear time-varying (LTV) systems with small perturbations based on Lemma 4.6 given in [56]. From (37), the perturbation term $G(Y_2, q)\bar{\eta}_\zeta$ may be very large. The main reasons lie in the fact that the term $\bar{\eta}_\zeta = \bar{\eta} - \bar{W}_\zeta^T S_\zeta(Z) = \eta(Z) - d(t) - \bar{W}_\zeta^T S_\zeta(Z)$ may be large with the possible large $d(t)$ and the term $G(Y_2, q) = Y_2 M^{-1}(q)$ may be also large due to the possible large Y_2 and $M^{-1}(q)$. Noting $G(Y_2, q)$ is a positive definite and symmetric matrix, a state transformation $E_2 = G^{-1}(Y_2, q)\xi_2/M_s$ is introduced to obtain the following class of LTV systems:

$$\begin{bmatrix} \dot{E}_2 \\ \dot{\bar{W}}_{\zeta 1} \\ \vdots \\ \dot{\bar{W}}_{\zeta n} \end{bmatrix} = \begin{bmatrix} A(t) & B^T(t) \\ -C(t) & 0 \end{bmatrix} \begin{bmatrix} E_2 \\ \bar{W}_{\zeta 1} \\ \vdots \\ \bar{W}_{\zeta n} \end{bmatrix} + \begin{bmatrix} \bar{\eta}_\zeta \\ \frac{M_s}{-\sigma \Gamma_\zeta \bar{W}_{\zeta 1}} \\ \vdots \\ -\sigma \Gamma_\zeta \bar{W}_{\zeta n} \end{bmatrix}, \quad (40)$$

where G is abbreviation of $G(Y_2, q)$, and

$$\begin{aligned} A(t) &= -K_2(t)G + \dot{G}^{-1}G \in R^{n \times n}, \\ B^T(t) &= -\frac{1}{M_s} \text{diag}\{S_\zeta^T, \dots, S_\zeta^T\} \in R^{n \times \zeta}, \\ C(t) &= -M_s \bar{\Gamma}_\zeta \bar{S}_\zeta(Z) Y_2 G \in R^{\zeta \times n}, \\ \bar{\Gamma}_\zeta &= \text{diag}\{\Gamma_\zeta, \dots, \Gamma_\zeta\} \in R^{\zeta \times \zeta}, \\ \bar{S}_\zeta &= \text{diag}\{S_\zeta, \dots, S_\zeta\} \in R^{\zeta \times n}. \end{aligned} \quad (41)$$

□

It is worth pointing out that $\bar{\eta}_\zeta/M_s$ and $-\sigma \Gamma_\zeta \bar{W}_{\zeta i}$ are small perturbations by choosing large enough M_s and small enough σ . Therefore, the system (40) can be regarded as a class of LTV systems with very small perturbations. It has been shown in [28] that the nominal part of the system (40)

can be guaranteed to be exponentially stable if the system (40) satisfies the following three conditions:

- (i) There exists a positive constant ψ such that, for all $t \geq 0$, the bound $\max\{\|B(t)\|, \|dB(t)/dt\|\} \leq \psi$ is satisfied;
- (ii) There exist symmetric and positive matrices $P(t)$ and $Q(t)$ such that $A^T(t)P(t) + P(t)A(t) + \dot{P}(t) = -Q(t)$;
- (iii) $S_\zeta(Z)$ satisfies the PE condition.

From Section 3, all closed-loop signals remain uniformly ultimately bounded. Therefore, the satisfaction of condition (i) can be easily checked. Moreover, we have verified that $S_\zeta(Z)$ satisfy the PE condition (see the above analysis of Theorem 9 for the details). Next, choose a matrix $P(t) = Y_2 G$. Since Y_2 and G are positive definite and symmetric, the matrix $P(t)$ is also symmetric and positive. Then, we have

$$\begin{aligned} A^T(t)P(t) + P(t)A(t) + \dot{P}(t) \\ = 2[-K_2(t)G + \dot{G}^{-1}G]Y_2 G + \dot{Y}_2 G + Y_2 \dot{G}. \end{aligned} \quad (42)$$

The inequality $A^T P + PA + \dot{P} < 0$ holds when choosing the appropriate control parameter $K_2(t) > (2\dot{G}^{-1}GY_2G + \dot{Y}_2G + Y_2\dot{G}) * (GY_2G)^{-1}/2$. Therefore, we have verified that the nominal part of the system (40) is uniformly exponentially stable. Further, based on the perturbation theory given in Lemma 4.6 [56], the weight estimate error $\bar{W}_{i\zeta}$ converges exponentially to a small neighborhood of zero for $\forall t \geq T$. Noting that $\bar{W}_{i\zeta} = \bar{W}_{i\zeta} - W_{i\zeta}^*$, so $\bar{W}_{i\zeta}$ converges exponentially to a small neighborhood of the desired weight value $W_{i\zeta}^*$ in a finite time T , and the corresponding desired weight value can be stored by constant neural weight values \bar{W}_i in (39).

Based on the spatially localized approximation property of RBF NN (6) and the constant weight values \bar{W} , the unknown system dynamics $\phi(Z)$ could be expressed as

$$\phi(Z) = \bar{W}_\zeta^T S_\zeta(Z) + \bar{\eta}_\zeta = \bar{W}^T S(Z) + \bar{\eta}, \quad (43)$$

where $\bar{\eta}_\zeta$ and $\bar{\eta}$ are close to η . From (43), neural networks $\bar{W}^T S(Z)$, containing the experience knowledge \bar{W} , can be used to accurately approximate the unknown system dynamics $\Phi(Z)$.

Furthermore, the learned knowledge can be described as follows: for the experienced recurrent orbit $\varphi_d(x_d(t))$, there exist positive constants d and ζ^* , which describe a local region Ω_{x_d} along $\varphi_d(x_d(t))$ such that

$$\begin{aligned} \text{dist}(\varphi(Z(t))|_{t \geq T}, \varphi_d(x_d(t))) &< d \implies \\ \left| \bar{W}^T S(Z) - \phi(Z) \right| &< \zeta^*, \end{aligned} \quad (44)$$

where ζ^* is close to η^* . For a new task, once the NN inputs $Z(t)$ enter the region Ω_{x_d} , the trained RBF networks $\bar{W}^T S(Z)$ can accurately approximate the uncertain nonlinearity $\phi(Z)$.

4.2. Static Controller Design with Knowledge Utilization. By invoking the stored weight values \bar{W} (39), a static controller

will be developed in this section to guarantee the prescribed performance of full-state tracking errors of the robotic system (1) for the same or a similar control task. Using the stored knowledge \bar{W} in (39), a static control law without neural weight estimated parameter adjustment online, instead of the adaptive NN control law (22), is designed as follows:

$$\tau = -k_{20}Y_2^{-1}\xi_2 - k_{21}Y_2\xi_2 - \bar{W}^T S(Z), \quad (45)$$

where k_{20} and k_{21} are positive design parameters and ξ_2 and Y_2 are defined in (12). Moreover, the virtual control law α is chosen the as same as Section 3; see (15) for the detail. Then, by combining (16), (19), and (45), we can obtain the following closed-loop system:

$$\begin{aligned} \dot{\xi}_1 &= Y_1(e_2 - k_1\xi_1), \\ \dot{\xi}_2 &= Y_2M^{-1} \left[-k_{20}Y_2^{-1}\xi_2 - k_{21}Y_2\xi_2 + \Phi(Z) \right. \\ &\quad \left. - \bar{W}^T S(Z) - Y_2^{-1}V_m(q, \dot{q})\xi_2 + \eta' \right], \end{aligned} \quad (46)$$

where $\eta' = -\dot{d}(t)$. Subsequently, construct the following Lyapunov function candidate:

$$\begin{aligned} V_1 &= \frac{\xi_1^T \xi_1}{2}, \\ V_2 &= \frac{1}{2} \xi_2^T M \xi_2. \end{aligned} \quad (47)$$

Noting the condition (44) and applying the similar backstepping step in Section 3, we have the following results.

Theorem 10. Consider the closed-loop system consisting of the robotic system (1), the bounded reference trajectory (2), the full-state tracking performance condition (3), the transformed error (10), the static neural learning control law (45) with the stored constant weight \bar{W} given in (39), and the virtual control law (15). Then, for the same or a similar recurrent reference orbit $\varphi_d(x_d(t))$ given in Theorem 6 and the initial conditions satisfying the prescribed performance (3), it can be guaranteed that all the closed-loop signals are uniformly ultimately bounded, and the constrained full-state tracking errors e_{ij} satisfy the prescribed performance (3) and converge to a small residual set of zero.

5. Simulation Results

To demonstrate the effectiveness of the proposed dynamic learning scheme, we consider a 2-link robot manipulator which is shown in Figure 1. From the n -link rigid robotic system (1), $q = [q_1, q_2]^T$ denotes the angular position of each joint and $\tau = [\tau_1, \tau_2]^T$ is the actuator input applied at the manipulator joints, respectively. Based on the system (1), the dynamic parameters of a 2-link robot manipulator are given by

$$M(q) = \begin{bmatrix} M_{11} & M_{12} \\ M_{12} & M_{22} \end{bmatrix},$$

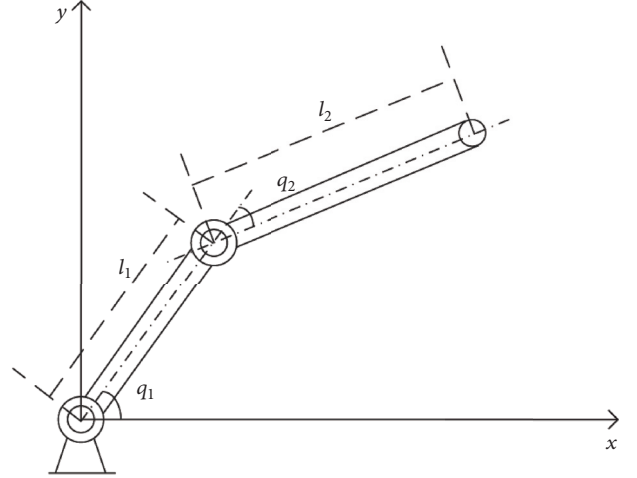


FIGURE 1: Diagram of a 2-link robotic manipulator.

$$\begin{aligned} F(\dot{q}) &= \begin{bmatrix} 0.8\dot{q}_1 \\ 0.6\dot{q}_2 \end{bmatrix}, \\ V_m(q, \dot{q})\dot{q} &= \begin{bmatrix} -2l_1l_2m_2 \sin(q_2)(\dot{q}_1\dot{q}_2 + 0.5\dot{q}_2^2) \\ 0.52l_1l_2m_2 \sin(q_2)\dot{q}_1^2 \end{bmatrix}, \\ G(q) &= \begin{bmatrix} (m_1 + m_2)l_1g \cos(q_1) + m_2l_2g \cos(q_2) \\ m_2l_2g \cos(q_1 + q_2) \end{bmatrix}, \end{aligned} \quad (48)$$

where

$$\begin{aligned} M_{11} &= l_2^2m_2 + l_1^2(m_1 + m_2) + 2l_1l_2m_2 \cos(q_2), \\ M_{12} &= l_2^2m_2 + 0.52l_1l_2m_2 \cos(q_2), \\ M_{22} &= l_2^2m_2, \end{aligned} \quad (49)$$

and l_i and m_i denote the length and the mass of link- i , $i = 1, 2$, and g denotes the gravity acceleration. In this paper, these system parameters are chosen as $m_1 = m_2 = 1$ kg, $l_1 = 0.8$ m, $l_2 = 2.3$ m, and $g = 9.8$ m/s², and the external disturbances $d = [0.1 \sin(t), 0.1 \cos(t)]^T$, which are bounded and satisfy Assumption 1.

The system output y is required to track the following desired reference trajectory $Y_d = [y_{d1}, y_{d2}]^T$:

$$\begin{aligned} y_{d1} &= \frac{80\pi}{360} \sin\left(\frac{2\pi t}{7}\right), \\ y_{d2} &= \frac{60\pi}{360} \cos\left(\frac{2\pi t}{7}\right). \end{aligned} \quad (50)$$

For full-state constrained tracking errors $e_1 = y - y_d = [e_{11}, e_{12}]^T$ and $e_2 = \dot{q} - \alpha_1 = [e_{21}, e_{22}]^T$, our target is to achieve the following prescribed transient and steady-state tracking error bounds:

$$-\underline{\delta}_{ij}\rho_i(t) < e_{ij}(t) < \bar{\delta}_{ij}\rho_i(t), \quad (51)$$

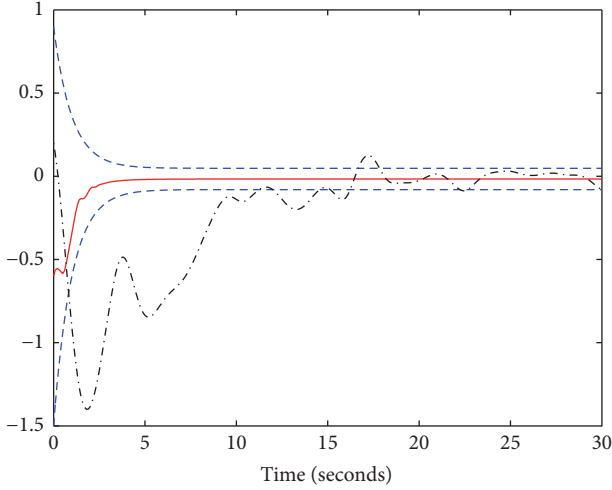


FIGURE 2: Angular position tracking error e_{11} : ANC with error transformation method (—), ANC in [30] (---), and the error bounds (---).

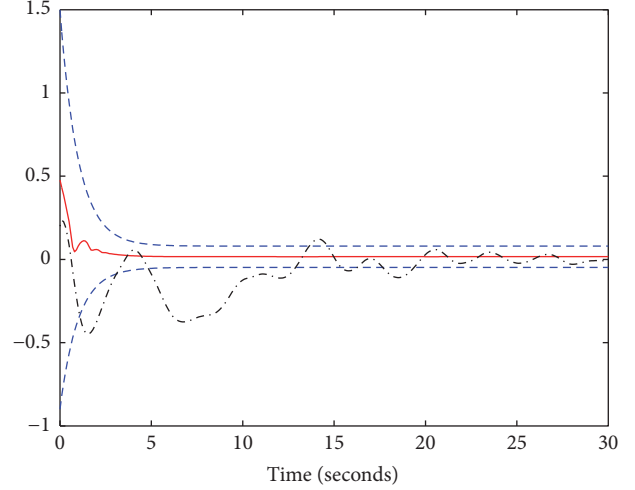


FIGURE 3: Angular position tracking error e_{12} : ANC with error transformation method (—), ANC method in [30] (---), and the error bounds (---).

where $i = 1, 2, j = 1, 2, \underline{\delta}_{11} = \underline{\delta}_{21} = 0.6, \bar{\delta}_{11} = \bar{\delta}_{21} = 1, \underline{\delta}_{12} = \underline{\delta}_{22} = 1$, and $\bar{\delta}_{12} = \bar{\delta}_{22} = 0.6$; the performance function $\rho_i(t)$ and the transformation function $T_{ij}^{-1}[e_{ij}/\rho_i(t)]$ are designed as

$$\rho_1(t) = (1.5 - 0.05)e^{-t} + 0.05, \quad \forall t \geq 0, \quad (52)$$

$$\rho_2(t) = (4 - 0.1)e^{-t} + 0.1, \quad \forall t \geq 0, \quad (53)$$

$$T_{ij}^{-1}\left[\frac{e_{ij}}{\rho_i(t)}\right] = \frac{1}{2} \ln \frac{\underline{\delta}_{ij} + e_{ij}/\rho_i(t)}{\bar{\delta}_{ij} - e_{ij}/\rho_i(t)}. \quad (54)$$

5.1. ANC Results with Full-State Tracking Error Constraints. According to Theorems 6 and 9, the main objective of this section is to use the proposed stable adaptive neural controller (22) with virtual control law (15) and neural weight adaptation law (23) such that the full-state tracking errors e_1 and e_2 satisfy the prescribed performance (51); the neural weight estimator \widehat{W} exponentially converges to the constant weight value \bar{W} ; the unknown system dynamics $\Phi(Z)$ in (20) is accurately approximated by the constant RBF NNs $\bar{W}^T S(Z)$.

In the simulation studies, the RBF network $W^T S(Z)$ consists of 3375 neurons whose centers are evenly spaced on $[-0.9, 0.9] \times [-0.9, 0.9] \times [-0.9, 0.9] \times [-0.5, 0.5] \times [-0.5, 0.5] \times [-0.5, 0.5]$ with the width $\eta_i = 0.4$. The other design parameters are chosen as follows: $k_{10} = 5, k_{20} = 25, k_{21} = 10, \Gamma = 15$, and $\sigma = 0.001$. The initial states are $\widehat{W}_1(0) = \widehat{W}_2(0) = 0, q(0) = [-0.6, 1]^T$, and $\dot{q}(0) = [1.5, -1]^T$. Simulation results are shown in Figures 2–10. Figures 2–5 show the constrained joint angular position and velocity tracking error performances, respectively. From Figures 2–5, it can be clearly seen that good transient performances have been achieved by adjusting the performance function (53) and design parameters $\underline{\delta}_{ij}, \bar{\delta}_{ij}$. The control input response is

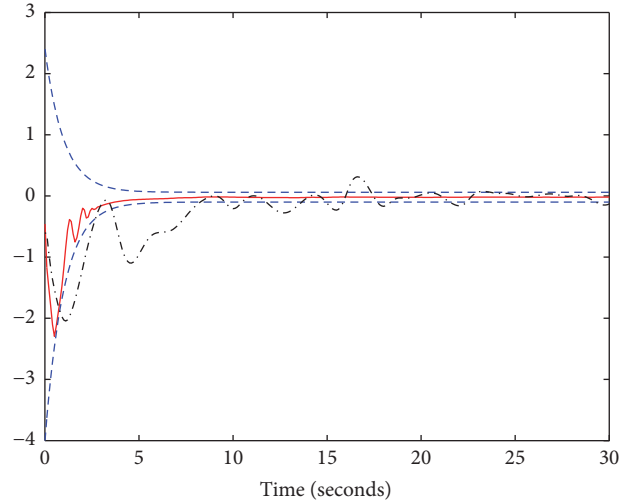


FIGURE 4: Angular velocity tracking error e_{21} : ANC with error transformation method (—), ANC in [30] (---), and the error bounds (---).

given in Figure 6. The partial weight convergence of $\widehat{W} = [\widehat{W}_1, \widehat{W}_2]$ is presented in Figures 7 and 8. It can be seen from Figures 7 and 8 that only partial weights converge to relatively large values, which means only the neurons along recurrent input signals Z can be motivated. Based on Theorem 9, the constant neural weight \bar{W}_i is chosen in the simulation as

$$\bar{W}_i = \min_{t \in [80, 100]} \widehat{W}_i(t), \quad i = 1, 2. \quad (55)$$

Figures 9 and 10 display unknown system dynamics $\Phi_1(Z)$ and $\Phi_2(Z)$, along the periodic reference signals Z , and can be accurately approximated by the constant RBF NN $\bar{W}_1^T S(Z)$ and $\bar{W}_2^T S(Z)$.

To further show the improved transient and steady-state tracking performance for the proposed control method with

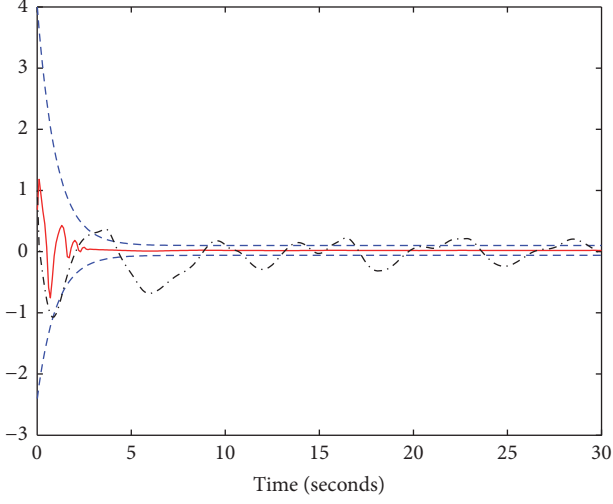


FIGURE 5: Angular velocity tracking error e_{22} : ANC with error transformation method (—), ANC method in [30] (---), and the error bounds (· · ·).

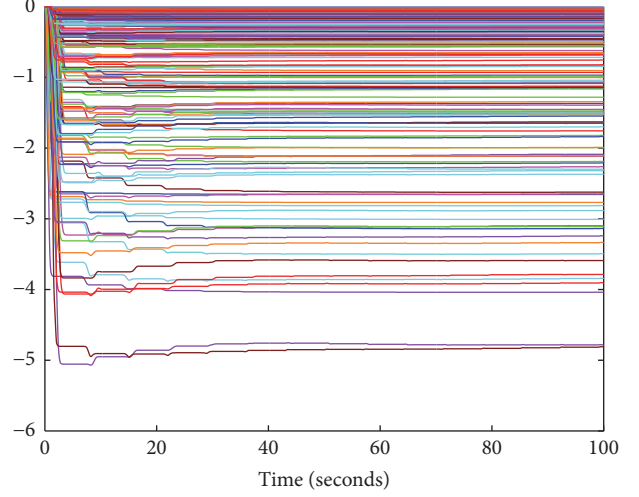


FIGURE 7: Partial parameter convergence of \widehat{W}_1 .

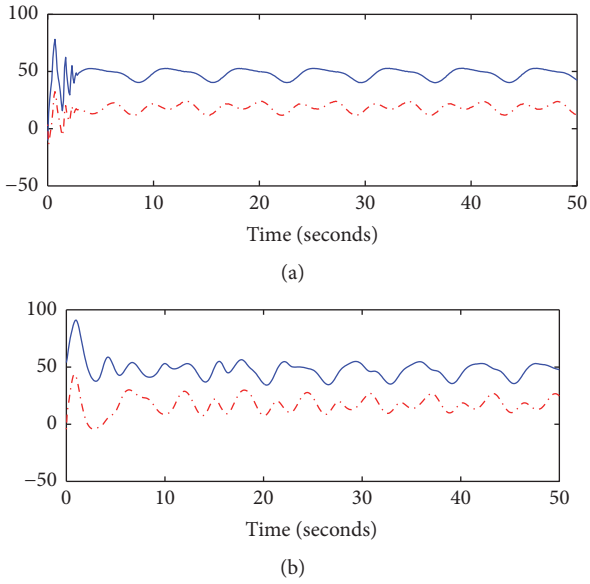


FIGURE 6: Control input responses: τ_1 (—), τ_2 (---): (a) ANC with error transformation and (b) ANC in [30].

full-state tracking performance constraints, the simulation comparison is given between the proposed method and the existing method without prescribed performances [30]. For comparison purpose, the existing method [30] is also used to control the same 2-link robot manipulator with the same initial condition $q(0) = [-0.6, 1]^T$, $\dot{q}(0) = [1.5, -1]^T$ and the same reference trajectory (50). For clarity, the existing method without prescribed performance proposed in [30] is recalled as follows: the control law is $\tau = -e_1 - c_2 e_2 - \widehat{W}^T S(Z)$ and neural weight updated laws $\dot{\widehat{W}} = \Gamma[S(Z)e_2^T - \sigma \widehat{W}]$ and $\alpha_1 = -c_1 e_1 + x_{d2}$. By choosing the appropriate control parameters $c_1 = 1$, $c_2 = 18$, $\sigma = 0.001$, and $\Gamma = 10$, to be fair, both control input signals of the two methods are

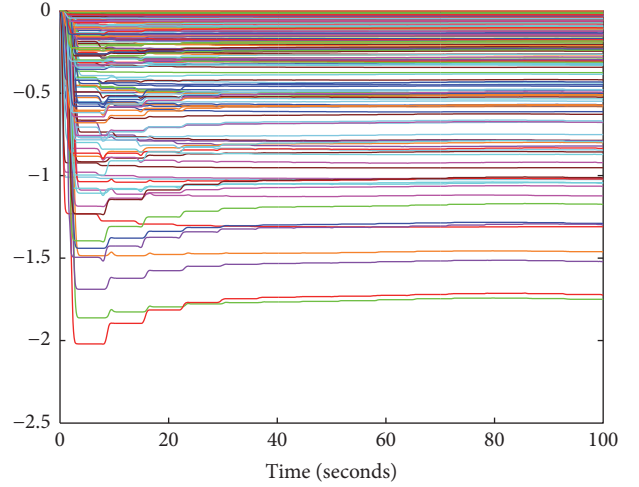


FIGURE 8: Partial parameter convergence of \widehat{W}_2 .

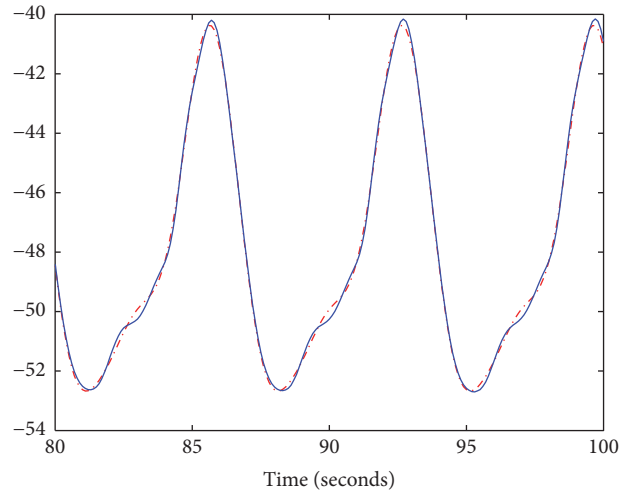


FIGURE 9: Function approximation: $\Phi_1(Z)$ (—) and $\widehat{W}_1^T S(Z)$ (---).

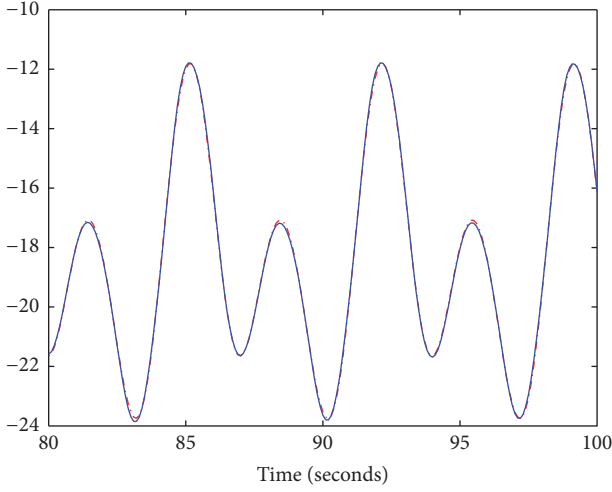


FIGURE 10: Function approximation: $\Phi_2(Z)$ (—) and $\bar{W}_2^T S(Z)$ (----).

required to have the similar amplitude, which is shown in Figure 6. The simulation results for comparison are given in Figures 2–5. It is clearly showed from Figures 2–5 that the proposed adaptive neural control scheme with full-state tracking performance constraints achieves a better transient and steady-state tracking control performance with smaller overshoot, faster convergence rate, and smaller steady-state error.

Remark 11. It is well known that the tracking performance relies on the choice of the control parameters and structure of RBF neural networks. However, how to choose design parameters for achieving the good tracking performance is still an open problem. In this simulation, the RBF networks $W^T S(Z)$ are constructed appropriately such that all neurons can cover the entire NN input trajectory $Z(t)$ space. Moreover, these controller parameters are chosen with large enough k_{10} , k_{20} , k_{21} , and Γ and small enough σ . It should be pointed out that these design parameters are chosen in this simulation by a trial-and-error method.

5.2. Neural Learning Control Results. By using the stored constant weight values \bar{W}_i in (55), the objective of this section is to invoke a static neural learning controller (45) to achieve the improved control performance with full-state tracking error constraints for the same or similar control tasks. For comparison purpose, the plant controlled, the reference trajectory, and the constrained tracking error performance are chosen the same as Section 5.1, while the initial conditions and the neural network structures are unchanged. In the simulation, with the control gains selected as $k_{10} = 3$, $k_{20} = 15$, and $k_{21} = 8$, simulation results for static neural learning control (45) are shown in Figures 11–15. From Figures 11–14, it can be seen that the smaller overshoot and the faster convergence are obtained using the learned knowledge \bar{W}_i in (55), while full-state tracking errors satisfy the prescribed performance. It is worth pointing out that a smaller control signal is used in neural learning control to achieve the

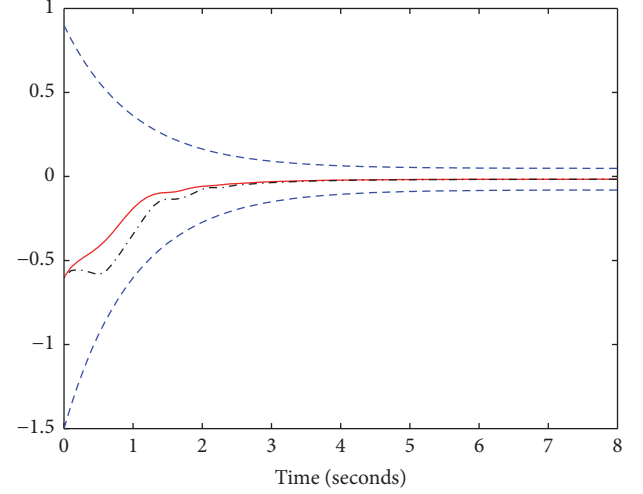


FIGURE 11: Angular position tracking error e_{11} : neural learning control (—), ANC (----), and the error bounds (---).

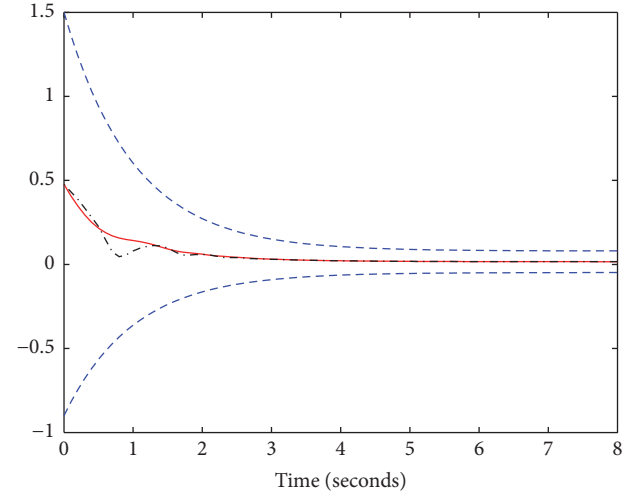


FIGURE 12: Angular position tracking error e_{12} : neural learning control (—), ANC (----), and the error bounds (---).

aforementioned improved tracking control performance; see Figure 15 for the details. Moreover, because the proposed static learning control scheme avoids the online adjustment of the neural weight values, the running time saves nearly 1/2 for the same simulation time interval $t = [0, 100]$ s and the same computer configuration. The static neural learning controller (45) especially avoids the trial-and-error process on control design parameters and NN parameters which is tuned in adaptive neural control process. Therefore, the proposed learning control scheme avoids effectively a great deal of time consumed by the adaptive neural control process.

6. Conclusions

This paper focused on the problem of full-state tracking error constraints for an n -link rigid robot with unknown

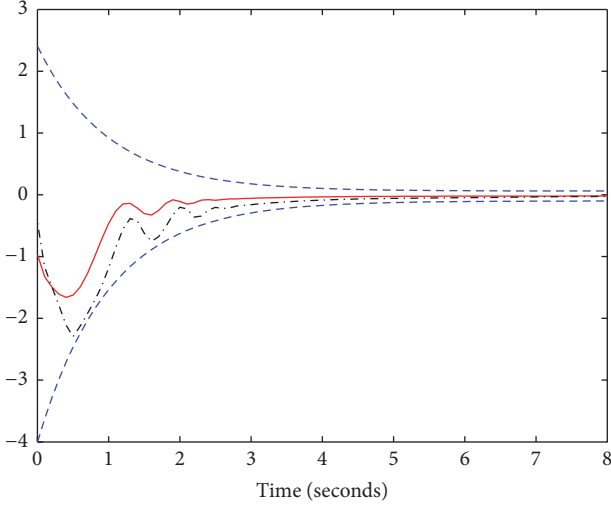


FIGURE 13: Angular velocity tracking error e_{21} : neural learning control (—), ANC (---), and the error bounds (- - -).

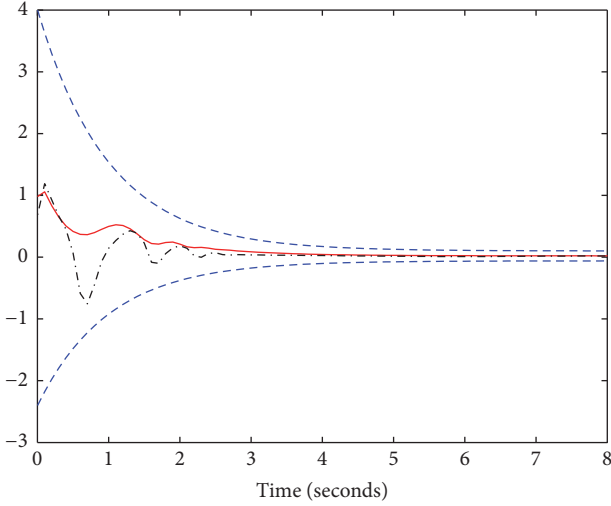


FIGURE 14: Angular velocity tracking error e_{22} : neural learning control (—), ANC (---), and the error bounds (- - -).

system dynamics and external disturbances. The performance transformation method was employed to transform the constrained full-state tracking errors into the unconstrained ones. By combining backstepping design and two independent Lyapunov functions, a novel adaptive neural control scheme was presented to guarantee all the signals in the closed-loop system are uniformly ultimately bounded, while this control scheme achieves predefined transient and steady-state tracking control performances concerning the link angular position and velocity tracking errors. Particularly, in the steady-state control process, the proposed neural control scheme can acquire, express, and store the knowledge of unknown system dynamics. The stored knowledge was reused to complete the same or similar tasks, so that the improved control performance was achieved with the less computational burden and the better transient-state tracking

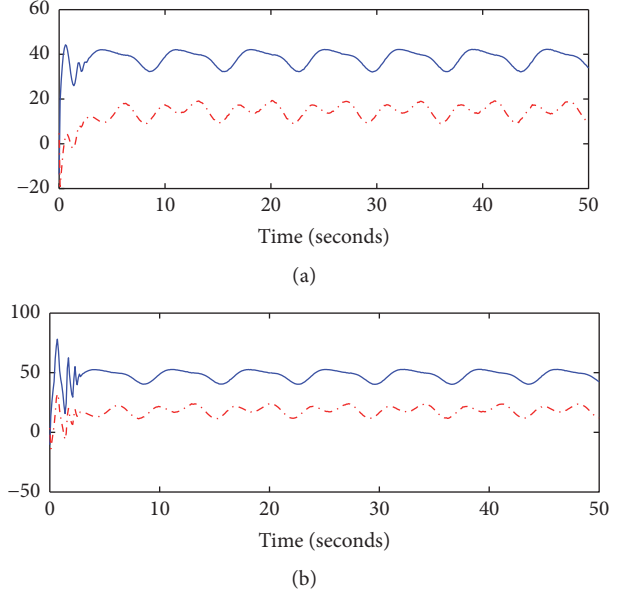


FIGURE 15: Control input responses: τ_1 (—), τ_2 (---): (a) neural learning control and (b) ANC.

performance. It should be pointed out that the considered n -link rigid robot is a class of simple multi-input and multioutput nonlinear systems. Therefore, how to extend the proposed method to complex nonlinear systems, such as nonaffine nonlinear systems, switched nonlinear systems, and stochastic large-scale systems, presents a challenging opportunity for future work.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (nos. 61374119, 61611130214, and 61473121), the Royal Society Newton Mobility Grant IE150858, the Guangdong Natural Science Foundation under Grant 2014A030312005, the Science and Technology New Star of Zhujiang, and the Fundamental Research Funds for the Central Universities.

References

- [1] F. L. Lewis, C. T. Abdallah, and D. M. Dawson, *Control of Robot Manipulators*, Macmillan, New York, NY, USA, 1993.
- [2] S. S. Ge, T. H. Lee, and C. J. Harris, *Adaptive Neural Network Control of Robotic Manipulators*, World Scientific, London, UK, 1998.
- [3] M. W. Spong and R. Ortega, "On adaptive inverse dynamics control of rigid robots," *Institute of Electrical and Electronics Engineers. Transactions on Automatic Control*, vol. 35, no. 1, pp. 92–95, 1990.

- [4] F. Cheng and G. Lee, "Robust control of manipulators using the computed torque plus H ∞ compensation method," *IEEE Proceedings - Control Theory and Applications*, vol. 143, no. 1, pp. 64–72, 1996.
- [5] Z. Li, J. Li, and Y. Kang, "Adaptive robust coordinated control of multiple mobile manipulators interacting with rigid environments," *Automatica. A Journal of IFAC, the International Federation of Automatic Control*, vol. 46, no. 12, pp. 2028–2034, 2010.
- [6] N. Sadegh and R. Horowitz, "Communications: An Exponentially Stable Adaptive Control Law For Robot Manipulators," *IEEE Transactions on Robotics and Automation*, vol. 6, no. 4, pp. 491–496, 1990.
- [7] M. Namvar and F. Aghili, "Adaptive force-motion control of coordinated robots interacting with geometrically unknown environments," *IEEE Transactions on Robotics*, vol. 21, no. 4, pp. 678–694, 2005.
- [8] D. Chwa, "Sliding-mode tracking control of nonholonomic wheeled mobile robots in polar coordinates," *IEEE Transactions on Control Systems Technology*, vol. 12, no. 4, pp. 637–644, 2004.
- [9] J. Yang and J. Kim, "Sliding mode control for trajectory tracking of nonholonomic wheeled mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 3, pp. 578–587, 1999.
- [10] R. M. Sanner and J. E. Slotine, "Gaussian networks for direct adaptive control," *IEEE Transactions on Neural Networks*, vol. 3, no. 6, pp. 837–863, 1992.
- [11] S. S. Ge and C. Wang, "Direct adaptive NN control of a class of nonlinear systems," *IEEE Transactions on Neural Networks*, vol. 13, no. 1, pp. 214–221, 2002.
- [12] Y.-J. Liu, L. Tang, S. Tong, and C. L. Chen, "Adaptive NN controller design for a class of nonlinear MIMO discrete-time systems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 5, pp. 1007–1018, 2015.
- [13] M. Wang, X. Liu, and P. Shi, "Adaptive neural control of pure-feedback nonlinear time-delay systems via dynamic surface technique," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 41, no. 6, pp. 1681–1692, 2011.
- [14] T.-P. Zhang, H. Wen, and Q. Zhu, "Adaptive fuzzy control of nonlinear systems in pure feedback form based on input-to-state stability," *IEEE Transactions on Fuzzy Systems*, vol. 18, no. 1, pp. 80–93, 2010.
- [15] H. Q. Wang, P. Shi, H. Li, and Q. Zhou, "Adaptive neural tracking control for a class of nonlinear systems with unmodeled dynamics," *IEEE Transactions on Cybernetics*, 2016.
- [16] H. Wang, P. X. Liu, and P. Shi, "Observer-based fuzzy adaptive output-feedback control of stochastic nonlinear multiple time-delay systems," *IEEE Transactions on Cybernetics*, 2017.
- [17] S. Tong, Y. Li, and P. Shi, "Observer-based adaptive fuzzy backstepping output feedback control of uncertain MIMO pure-feedback nonlinear systems," *IEEE Transactions on Fuzzy Systems*, vol. 20, no. 4, pp. 771–785, 2012.
- [18] C. M. Kwan, A. Yesildirek, and F. L. Lewis, "Robust force/motion control of constrained robots using neural network," *Journal of Robotic Systems*, vol. 16, no. 12, pp. 697–714, 1999.
- [19] Y. Karayiannidis, G. Rovithakis, and Z. Doulgeri, "Force/position tracking for a robotic manipulator in compliant contact with a surface using neuro-adaptive control," *Automatica. A Journal of IFAC, the International Federation of Automatic Control*, vol. 43, no. 7, pp. 1281–1288, 2007.
- [20] C. Yang, X. Wang, Z. Li, Y. Li, and C. Su, "Teleoperation control based on combination of wave variable and neural networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 8, pp. 2125–2136, 2017.
- [21] C. Yang, J. Luo, Y. Pan, Z. Liu, and C. Su, "Personalized variable gain control with tremor attenuation for robot teleoperation," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2017.
- [22] C. Yang, K. Huang, H. Cheng, Y. Li, and C. Su, "Haptic identification by ELM-controlled uncertain manipulator," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 8, pp. 2398–2409, 2017.
- [23] K. S. Fu, "Learning Control Systems and Intelligent Control Systems: An Intersection of Artificial Intelligence and Automatic Control," *IEEE Transactions on Automatic Control*, vol. 16, no. 1, pp. 70–72, 1971.
- [24] B. Xu, C. Yang, and Z. Shi, "Reinforcement learning output feedback NN control using deterministic learning technique," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 3, pp. 635–641, 2014.
- [25] C. Yang, X. Wang, L. Cheng, and H. Ma, "Neural-learning-based telerobot control with guaranteed performance," *IEEE Transactions on Cybernetics*, 2017.
- [26] B. Luo, H.-N. Wu, and T. Huang, "Off-policy reinforcement learning for H ∞ control design," *IEEE Transactions on Cybernetics*, vol. 45, no. 1, pp. 65–76, 2015.
- [27] C. Wang and D. J. Hill, "Learning from neural control," *IEEE Transactions on Neural Networks*, vol. 17, no. 1, pp. 130–146, 2006.
- [28] T. Liu, C. Wang, and D. J. Hill, "Learning from neural control of nonlinear systems in normal form," *Systems & Control Letters*, vol. 58, no. 9, pp. 633–638, 2009.
- [29] S.-L. Dai, C. Wang, and M. Wang, "Dynamic learning from adaptive neural network control of a class of nonaffine nonlinear systems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 1, pp. 111–123, 2014.
- [30] C. Wang, M. Wang, T. Liu, and D. J. Hill, "Learning from ISS-modular adaptive NN control of nonlinear strict-feedback systems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 10, pp. 1539–1550, 2012.
- [31] M. Wang and C. Wang, "Learning from adaptive neural dynamic surface control of strict-feedback systems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 6, pp. 1247–1259, 2015.
- [32] S.-L. Dai, C. Wang, and F. Luo, "Identification and learning control of ocean surface ship using neural networks," *IEEE Transactions on Industrial Informatics*, vol. 8, no. 4, pp. 801–810, 2012.
- [33] S.-L. Dai, M. Wang, and C. Wang, "Neural learning control of marine surface vessels with guaranteed transient tracking performance," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 3, pp. 1717–1727, 2016.
- [34] M. Wang and A. Yang, "Dynamic learning from adaptive neural control of robot manipulators with prescribed performance," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 8, pp. 2244–2255, 2017.
- [35] W. He, A. O. David, Z. Yin, and C. Sun, "Neural network control of a robotic manipulator with input deadzone and output constraint," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 6, pp. 759–770, 2016.
- [36] K. P. Tee, S. S. Ge, and E. H. Tay, "Barrier Lyapunov functions for the control of output-constrained nonlinear systems," *Automatica. A Journal of IFAC, the International Federation of Automatic Control*, vol. 45, no. 4, pp. 918–927, 2009.

- [37] 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 Transactions on Neural Networks*, vol. 21, no. 8, pp. 1339–1345, 2010.
- [38] W. He, C. Sun, and S. S. Ge, "Top tension control of a flexible marine riser by using integral-barrier lyapunov function," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 2, pp. 497–505, 2015.
- [39] 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 Transactions on Industrial Informatics*, vol. 13, no. 3, pp. 1162–1171, 2017.
- [40] K. P. Tee and S. S. Ge, "Control of nonlinear systems with partial state constraints using a barrier Lyapunov function," *International Journal of Control*, vol. 84, no. 12, pp. 2008–2023, 2011.
- [41] W. He, Y. Chen, and Z. Yin, "Adaptive neural network control of an uncertain robot with full-state constraints," *IEEE Transactions on Cybernetics*, vol. 46, no. 3, pp. 620–629, 2016.
- [42] Y.-J. Liu and S. Tong, "Barrier Lyapunov functions-based adaptive control for a class of nonlinear pure-feedback systems with full state constraints," *Automatica. A Journal of IFAC, the International Federation of Automatic Control*, vol. 64, pp. 70–75, 2016.
- [43] C. P. Bechlioulis and G. A. Rovithakis, "Robust adaptive control of feedback linearizable MIMO nonlinear systems with prescribed performance," *Institute of Electrical and Electronics Engineers. Transactions on Automatic Control*, vol. 53, no. 9, pp. 2090–2099, 2008.
- [44] Y. Li, S. Tong, and T. Li, "Adaptive fuzzy output-feedback control for output constrained nonlinear systems in the presence of input saturation," *Fuzzy Sets and Systems. An International Journal in Information Science and Engineering*, vol. 248, pp. 138–155, 2014.
- [45] A. K. Kostarigka, Z. Doulgeri, and G. A. Rovithakis, "Prescribed performance tracking for flexible joint robots with unknown dynamics and variable elasticity," *Automatica. A Journal of IFAC, the International Federation of Automatic Control*, vol. 49, no. 5, pp. 1137–1147, 2013.
- [46] J. Na, Q. Chen, X. Ren, and Y. Guo, "Adaptive prescribed performance motion control of servo mechanisms with friction compensation," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 1, pp. 486–494, 2014.
- [47] Y. Li, Z. Ma, and S. Tong, "Adaptive fuzzy output-constrained fault-tolerant control of nonlinear stochastic large-scale systems with actuator faults," *IEEE Transactions on Cybernetics*, 2017.
- [48] Y. Li, S. Tong, L. Liu, and G. Feng, "Adaptive output-feedback control design with prescribed performance for switched nonlinear systems," *Automatica. A Journal of IFAC, the International Federation of Automatic Control*, vol. 80, pp. 225–231, 2017.
- [49] S. I. Han and J. M. Lee, "Partial tracking error constrained fuzzy dynamic surface control for a strict feedback nonlinear dynamic system," *IEEE Transactions on Fuzzy Systems*, vol. 22, no. 5, pp. 1049–1061, 2014.
- [50] S. Tong, S. Sui, and Y. Li, "Fuzzy adaptive output feedback control of MIMO nonlinear systems with partial tracking errors constrained," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 4, pp. 729–742, 2015.
- [51] C. P. Bechlioulis, Z. Doulgeri, and G. A. A. A. Rovithakis, "Neuro-adaptive force/position control with prescribed performance and guaranteed contact maintenance," *IEEE Transactions on Neural Networks*, vol. 21, no. 12, pp. 1857–1868, 2010.
- [52] M. Wang, C. Wang, and X. Liu, "Dynamic learning from adaptive neural control with predefined performance for a class of nonlinear systems," *Information Sciences. An International Journal*, vol. 279, pp. 874–888, 2014.
- [53] M. Wang, C. Wang, P. Shi, and X. Liu, "Dynamic learning from neural control for strict-feedback systems with guaranteed predefined performance," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 12, pp. 2564–2576, 2016.
- [54] R. Selmic and F. Lewis, "Deadzone compensation in motion control systems using neural networks," *Institute of Electrical and Electronics Engineers. Transactions on Automatic Control*, vol. 45, no. 4, pp. 602–613, 2000.
- [55] A. J. Kurdila, F. J. Narcowich, and J. D. Ward, "Persistency of excitation in identification using radial basis function approximations," *SIAM Journal on Control and Optimization*, vol. 33, no. 2, pp. 625–642, 1995.
- [56] H. K. Khalil, *Nonlinear Systems*, Prentice-Hall, Upper Saddle River, NJ, USA, 3rd edition, 2002.

