IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 21, 2017, accepted November 10, 2017, date of publication December 4, 2017, date of current version February 14, 2018.

Digital Object Identifier 10.1109/ACCESS.2017.2779940

Robust Adaptive Tracking Control for Manipulators Based on a TSK Fuzzy Cerebellar Model Articulation Controller

JIANSHENG GUAN[1,2], CHIH-MIN LIN[3,4,5], (Fellow, IEEE), GUO-LI JI[1], LING-WU QIAN[1], AND YI-MIN ZHENG[1]

[1]Department of Automation, Xiamen University, Xiamen 361000, China
[2]High-Voltage Key Laboratory of Fujian Province, Xiamen University of Technology, Xiamen 361024, China
[3]School of Information Science and Engineering, Xiamen University, Xiamen 361000, China
[4]Department of Electrical Engineering, Yuan Ze University, Zhongli, Taiwan
[5]Innovation Center for Big Data and Digital Convergence, Yuan Ze University, Zhongli, Taiwan

Corresponding author: Chih-Min Lin (cml@saturn.yzu.edu.tw)

This work was supported in part by the Natural Science Foundation of Fujian Province, China, under Grant 2017J01782, in part by the Natural Sciences and Engineering Research Council of Canada, in part by the National Natural Science Foundation of China under Grant 61573296 and Grant 61473329, and in part by the National Science Council of Taiwan under Grant NSC98-2221-E-155-059-MY3.

ABSTRACT The robot manipulator system is a complicated system with multiple-input and multiple-output, high nonlinearity, strong coupling, and uncertainties, such as parameter disturbances, external interference, and unmodeled dynamics. A robust adaptive Takagi-Sugeuo-Kang fuzzy cerebellar model articulation controller (RATFC) is proposed and applied to a robot manipulator to achieve high-precision position and speed control. A Takagi-Sugeuo-Kang fuzzy cerebellar model articulation controller is adopted, and the parameters are regulated by the derived adaptable rules according to a Lyapunov function. The robust compensation controller mitigates approximation-based errors. Finally, simulation results show that the proposed RATFC can achieve improved tracking performance compared with other neural network controllers.

INDEX TERMS TSK fuzzy system, cerebellar model articulation controller, robot manipulator system.

I. INTRODUCTION

Control systems for robot manipulators is a hot topic in control problems. Due to its complicated physical structure and highly coupled nonlinear kinetics, the dynamic characteristics of the system are hard to represent with an exact mathematical model. Thus, robot manipulators have trouble tracking trajectories accurately. The mission becomes further intensified when the system is subjected to a variety of model uncertainties and immeasurable external interference. There are several control algorithms for manipulator path tracking. Traditional approaches include proportional-integral-derivative control [1], robust control [2], computed torque approach [3], variable structure control [4] and adaptive control [5]. Nevertheless, uncertainties such as modeling errors, high frequencies, manipulator joint friction and signal detection errors degrade the control system's performance. Thus, many of the traditional control system feedback algorithms are not able to meet the control demands. Intelligent design methods (e.g., neural network [6] and fuzzy theory [7]) have nonlinear shift features and parallel computing capabilities, which offer a valid method for intelligent control. Thus, the contemporary trend of control methods is not only to combine two approaches but also to develop new control skills.

Neural nets have been used in several trajectory-tracking control systems [8]–[14]. This type of algorithm possesses strong learning capabilities and a parallel distribution structure to approximate nonlinear relationships with superior robustness and error tolerance. The quality of neural network adaptive control law is improved with the addition of a Lyapunov steadiness norm. The application of neural networks to path-tracking robotic manipulators can be primarily separated into two categories: I. The neural net is used to approximate nonlinear or uncertain manipulator system parameters (neural net approximation), II. The neural net is accessed from within the manipulator system controller. In the entire control course, due to the existence of the approximate fault, the neural net is always regularly connected with other control approaches to make up for the effect of dynamic nonlinearity and system uncertainties so that the system's steadiness, convergence and robustness can be enhanced. Lin and Ting [8] suggested a recurrent wavelet neural net-based adaptive control system for a two-link robot manipulator system that used a recurrent wavelet neural network to simulate an optimal controller online while offering smooth and buffeting-free steadiness compensation via a bounding compensator. The mechanism can achieve the desired tracking performance and robustness without the appearance of buffeting during the control process. He *et al.* [9] showed an adaptable neural net tracking control of a robot manipulator with an input dead-zone and an output restraint. They applied an obstacle Lyapunov function to remove the output restraints and then employed an adaptive neural net to approximate the dead-zone function and the unknown model of the robot manipulator.

The precise mathematical model of the manipulator is not needed in the adaptive fuzzy manipulator control; thus, the experience of the control expert can be used [15], [16]. Adaptive fuzzy control of the robotic application complies with some rules in different forms. There are two main forms: I. Direct adaptive fuzzy control: its rules use control knowledge in accordance with the deviation between the real system performance and the optimal performance, which directly modifies the control parameters. II. Indirect adaptive fuzzy control: its rules adopt the knowledge of the controlled object to differentiate between the various interference or compensation nomenclature and to bind them correspondingly, which improves the model via online recognition of the control object and then creates an on-line fuzzy controller by considering the obtained model. Wai *et al.* [17] introduced a fuzzy neural network control for robotic manipulators that takes over the robust features of sliding-mode control and achieves a high-precision path-tracing and steady control performance. Lian [18] developed a self-organizing fuzzy radial basis-function neural net controller for robot manipulators, which dealt with the problem resulting from an unsuitable choice of parameters in an SOFC and removed the coupled dynamic influences on the degrees-of-freedom of the robotic control system.

The goal of this paper is to create a real-time parametric TFCMAC neural net that can compensate for the negative influence of friction and interference in the path-tracing control of a robotic manipulator. Based on unmodeled nonlinearities and uncertain interferences, cerebellar model articulation controllers (CMAC) have the capability of recognizing and controlling complicated nonlinear dynamic mechanisms because of its computational simplicity, rapid learning and great generalization abilities [19]. Integrating fuzzy logic control and CMAC into fuzzy CMAC (FCMAC) benefits the function approximation accuracy based on the CMAC weighting coefficients. The algorithm can provide a variable method for modeling, nonlinear identification and control of nonlinear dynamic systems, which are intrinsically not certain and precise [20]. This paper is organized as follows.

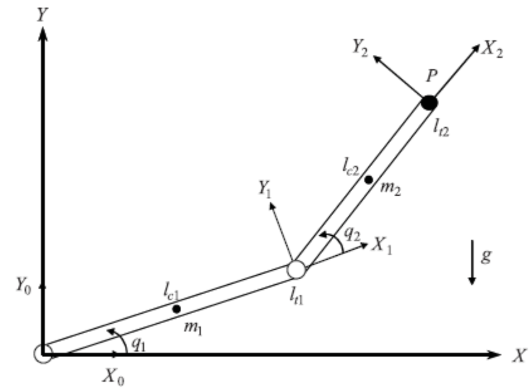


FIGURE 1. Two-link robot manipulator's architecture.

After the introduction, Section II shows the dynamical model and computed-torque control of an n-rigid link robot manipulator. In Section III, a TSKFCMAC is developed and used to approximate the system model to create a robust controller. The closed-loop control system and the adaptive laws in the TFCMAC system are derived in the sense of Lyapunov theory and the steepest decent algorithm in Section IV. The simulation results of a two-link robotic manipulator indicate the robust control performance of the proposed scheme in Section V. Conclusions are drawn in Section VI.

II. DYNAMICS OF MANIPULATOR AND COMPUTED-TORQUE CONTROL

A. THE DYNAMICAL MODEL

The nonlinear dynamic model of an n-link robot manipulator is described by the following Lagrange form [21]

$$M(q)\ddot{q} + V(q,\dot{q})\dot{q} + G(q) + F(\dot{q}) + \tau_d = \tau \tag{1}$$

where $M(q) \in R^{n\times n}$ is the symmetrical positive definite inertia matrix, $V(q,\dot{q}) \in R^n$ is the centripetal and Coriolis vector, $\dot{M}(q) - 2V(q,\dot{q})$ is the oblique symmetric matrix, $G(q) \in R^n$ is the gravity, $F(\dot{q}) \in R^n$ is the static and dynamic friction, $\tau_d(t) \in R^n$ is a disturbance caused by load changes or modeling errors, $\tau(t) \in R^n$ is the control input, and $q \in R^n$ is the joint variable vector.

B. COMPUTED-TORQUE CONTROL

The joint variables ensure the trajectory tracking. The desired manipulator trajectory is denoted as $q_d(t)$, and the tracking error is defined as

$$e(t) = q_d(t) - q(t) \tag{2}$$

The error-function is defined as

$$\alpha = \dot{e} + \Lambda e \tag{3}$$

where the selection of $\Lambda = \Lambda^T = \text{diag}[\lambda_1\lambda_2\cdots\lambda_n]^T > 0$; if $s^{n-1} + \lambda_{n-1}s^{n-2} + \cdots + \lambda_1$, is a Hurwitz polynomial. This result implies that while $\alpha \to 0$, then $e \to 0$.

From (2) and (3), the following is obtained

$$\dot{q} = -\alpha + \dot{q}_d + \Lambda e \tag{4}$$

and

$$
\begin{aligned}
M\dot{\alpha} &= M(\ddot{q}_d - \ddot{q} + \Lambda\dot{e}) = M(\ddot{q}_d + \Lambda\dot{e}) - M\ddot{q} \\
&= M(\ddot{q}_d + \Lambda\dot{e}) + V\dot{q} + G + F + \tau_d - \tau \\
&= M(\ddot{q}_d + \Lambda\dot{e}) - V\alpha + V(\dot{q}_d + \Lambda e) + G + F + \tau_d - \tau \\
&= -V\alpha - \tau + f^* + \tau_d \qquad (5)
\end{aligned}
$$

where $f^*(x) = M\ddot{q}_r + V\dot{q}_r + G + F$ and $\dot{q}_r = \dot{q}_d + \Lambda e$, $f^*(x)$ contains all the information of model, i.e., all model information can be expressed in $f^*(x)$.

Referring to the design method of [22], the control law for system (1) can be written

$$\tau = \hat{f}(x) + K_v\alpha \qquad (6)$$

where $\hat{f}(x)$ is the approximation of $f^*(x)$, $K_v = K_v^T > 0$ is a chosen gain matrix.

Incorporating (6) into (5) yields

$$
\begin{aligned}
M\dot{\alpha} &= -V\alpha - \hat{f} - K_v\alpha + f^* + \tau_d \\
&= -(K_v + V)\alpha + \tilde{f} + \tau_d \\
&= -(K_v + V)\alpha + \xi_0 \qquad (7)
\end{aligned}
$$

where $\tilde{f} = f^* - \hat{f}$, $\xi_0 = \tilde{f} + \tau_d$.

Then, a Lyapunov function is defined as

$$L_1 = \frac{1}{2}\alpha^T M\alpha \qquad (8)$$

According to the oblique symmetry characteristic of the robot manipulator $\alpha^T(\dot{M} - 2V)\alpha = 0$ [3], the derivative of L_1 will be

$$
\begin{aligned}
\dot{L}_1 &= \alpha^T M\dot{\alpha} + \frac{1}{2}\alpha^T\dot{M}\alpha \\
&= -\alpha^T K_v\alpha + \frac{1}{2}\alpha^T(\dot{M} - 2V)\alpha + \alpha^T\xi_0 \\
&= \alpha^T\xi_0 - \alpha^T K_v\alpha \qquad (9)
\end{aligned}
$$

This shows that if the K_v value is fixed, the stability of the control system depends on ξ_0, i.e., the approximation accuracy from $\hat{f}$ to f^* and the size of the disturbance τ_d.

Therefore, a Takagi–Suegeno–Kang Fuzzy Cerebellar Model Articulation Controller (TFC) is proposed to approximate the uncertain term f^* in this paper.

III. TAKAGI–SUEGENO–KANG FUZZY CEREBELLAR MODEL ARTICULATION CONTROLLER

Because the exact information about the dynamic models (i.e., $M(q)$, $V(q, \dot{q})$, $G(q)$ and $F(\dot{q})$) usually cannot be explicitly acquired; therefore, as presented in Fig. 2, a robust adaptive TFCMAC control system is developed to mitigate the problem. The control system is characterized by

$$u = \hat{u}_{TFC} + u_R \qquad (10)$$

where $\hat{u}_{TFC}$ is the output of TFC, and u_R is a robust compensator.

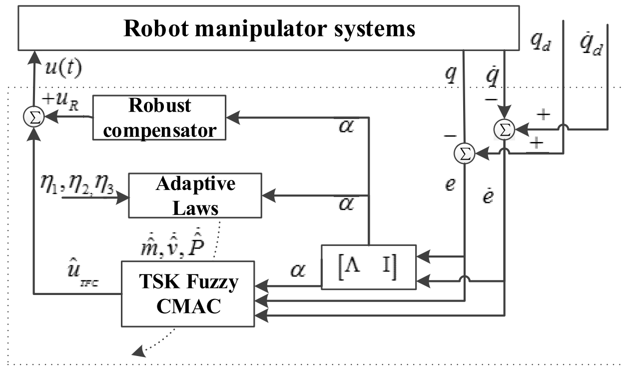


FIGURE 2. Block diagram of RATFC control system.

If a TSK fuzzy CMAC is considered, then the fuzzy inference rules are put forward as follows

$$
\begin{aligned}
R^l:&\ \text{If } X_1 \text{ is } f_{1jk} \text{ and } X_2 \text{ is } f_{2jk}, \ldots, \text{ and } X_{n_i} \text{ is } f_{n_ijk} \\
&\text{then } w_{jk} = p_{1jk}X_1 + p_{2jk}X_2 + \cdots + p_{n_ijk}X_{n_i} \\
&\text{for } j = 1, 2, \ldots, n_j,\ k = 1, 2, \ldots, n_k,\ n_l = 1, 2, \ldots, n_l \qquad (11)
\end{aligned}
$$

where X_i is the ith input, n_i is the dimension of input; n_j is the number of layers of each input dimension; n_k is the number of blocks of each layer; $n_l = n_jn_k$ is the number of fuzzy rules; f_{ijk} is the fuzzy set for the ith input, jth layer and kth block; and w_{jk} is the weight of TSK-type output in the consequent part. Compared with a fuzzy neural network, this fuzzy CMAC is constructed with blocks and layers in the input space, as presented in Fig. 3(a). A schematic diagram of a 2-D ($n_i = 2$) fuzzy CMAC with four layers ($n_j = 4$) and two blocks ($n_k = 2$) in each layer is demonstrated in Fig. 3(b).

The TFC consists of input, association memory, receptive-field, weight memory, and output. The signal propagation in each space is identified in the following description.

1) Input space: For a given $X = \left[X_1, \cdots, X_i, \cdots, X_{n_i}\right]^T \in \Re^{n_i}$, where X_i stands for the ith input to the node of layer 1, each input state variable X_i can be quantized into discrete regions (called an element) according to the given control space.

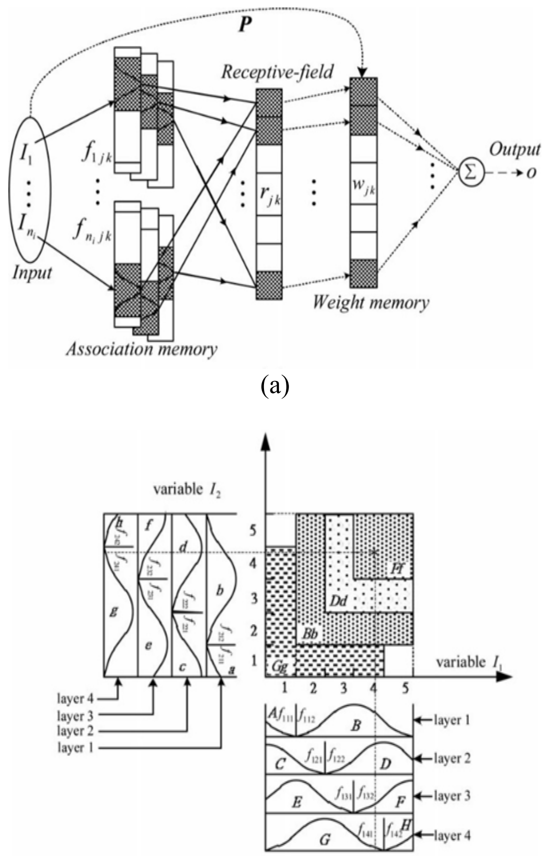
2) Association memory space (Membership function): The accumulation of several elements forms a block. Each block in this space performs the function of a receptive-field basis. The Gaussian function is employed as a receptive-field basis function that is described as

$$
\begin{aligned}
f_{ijk}(F_{ijk}) &= \exp(-F_{ijk}^2) \quad \text{for } i = 1, 2, \ldots, n_i, \\
&j = 1, 2, \ldots, n_j, \text{ and } k = 1, 2, \ldots, n_k \qquad (12)
\end{aligned}
$$

where $F_{ijk} = \frac{X_i - m_{ijk}}{v_{ijk}}$, m_{ijk} is a mean parameter, and v_{ijk} is a variance parameter of the Gaussian function.

3) Receptive-field space (Hypercube): The multidimensional receptive field function is defined as

$$
\begin{aligned}
r_{jk} &= \prod_{i=1}^{n_i} f_{ijk}(F_{ijk}) = \prod_{i=1}^{n_i} \exp\left(-\left(\frac{X_i - m_{ijk}}{v_{ijk}}\right)^2\right) \\
&\text{for } j = 1, 2, \ldots, n_j, \quad \text{and } k = 1, 2, \ldots, n_k \qquad (13)
\end{aligned}
$$



(a)

(b)

FIGURE 3. Structure of a TFCMAC. (a) Architecture of a TFC. (b) Organization of a 2-D fuzzy CMAC.

where r_{jk} is associated with the jth layer and kth block, i.e., the "product" operation of the fired value in the antecedent part of the fuzzy rules in (11).

The expression of the multidimensional receptive-field functions can be expressed as a vector

$$\mathbf{r} = \left[r_{11}, \ldots, r_{1n_k}, r_{21}, \ldots, r_{2n_k}, \ldots, r_{n_j1}, \ldots, r_{n_jn_k}\right]^T \in \Re^{n_jn_k} = \left[r_1, \ldots, r_l, \ldots, r_{n_l}\right]^T \in \Re^{n_l} \tag{14}$$

4) Weight memory (TSK fuzzy output weight): Each location of a receptive-field of a specific adaptable value in the weight memory space can be characterized as

$$\mathbf{w} = \left[w_{11}, \ldots, w_{1n_k}, w_{21}, \ldots, w_{2n_k}, \ldots, w_{n_j1}, \ldots, w_{n_jn_k}\right]^T \in \Re^{n_jn_k} = \left[w_1, \ldots, w_l, \ldots, w_{n_l}\right]^T \in \Re^{n_l} \tag{15}$$

where $w_{jk} = p_{1jk}X_1 + p_{2jk}X_2 + \cdots + p_{n_ijk}X_{n_i}$ represents the connecting weight value of the output associated with the jth layer and kth block. Then, W can be expressed as

$$W = \begin{bmatrix} w_{11} \\ \vdots \\ w_{1n_k} \\ w_{21} \\ \vdots \\ w_{2n_k} \\ \vdots \\ w_{n_j1} \\ \vdots \\ w_{n_jn_k} \end{bmatrix} = \begin{bmatrix} p_{111} & p_{211} & \cdots & p_{n_i11} \\ \vdots & \vdots & \ddots & \vdots \\ p_{11n_k} & p_{21n_k} & \cdots & p_{n_i1n_k} \\ p_{121} & p_{221} & \cdots & p_{n_i21} \\ \vdots & \vdots & \ddots & \vdots \\ p_{12n_k} & p_{22n_k} & \cdots & p_{n_i2n_k} \\ \vdots & \vdots & \ddots & \vdots \\ p_{1n_j1} & p_{2n_j1} & \cdots & p_{n_in_j1} \\ \vdots & \vdots & \ddots & \vdots \\ p_{1n_jn_k} & p_{2n_jn_k} & \cdots & p_{n_in_jn_k} \end{bmatrix} \times \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ \vdots \\ X_{n_i} \end{bmatrix} \equiv \mathbf{P}X \tag{16}$$

where $P \in \Re^{n_jn_k \times n_i}$ is the output parameter matrix and $X = [X_1, X_2, \ldots, X_{n_i}]^T \in \Re^{n_i}$.

5) Output: The output of TFC is the algebraic sum of the activated weight receptive-field and is represented by

$$u_{TFC} = o = \sum_{j=1}^{n_j}\sum_{k=1}^{n_k} w_{jk}r_{jk} = W^T r = (PX)^T r \tag{17}$$

For this fuzzy CMAC, if only one element (neuron) can be carried in each block, and each input space could only have one layer, the fuzzy CMAC can be simplified as a fuzzy neural network [23], [24]. Thus, the fuzzy CMAC is the overall expression of the fuzzy neural network but is more generalized, learns faster and recalls more than the latter. Additionally, the fuzzy rule system in (11) (in this case) could be reduced to a TSK fuzzy system [25]–[28]. Moreover, the system can return to a traditional CMAC after removing the fuzzy rule system from the fuzzy CMAC and after reducing the output weights from TSK-type to singleton-type.

IV. THE ROBUST ADAPTIVE TSK FUZZY CMAC CONTROLLER

Assume that an optimal u^*_{TFC} exists to approach $f^*(x)$ such that

$$f^*(x) = u^*_{TFC}(X, P^*, m^*, v^*) + \varepsilon \equiv \left(P^*X\right)^T r^* + \varepsilon \tag{18}$$

where ε is a minimum approximation error; P^*, m^*, v^* and r^* are the optimal parameter matrix and vector form of P, m, v and r, respectively. Nevertheless, since the optimal $f^*(x)$ cannot be acquired, the online estimation of $\hat{f}(x)$ is used to estimate the optimal TFC. From (17), the control law of (6) can be expressed in the following form

$$\tau = \hat{f}(x) + K_v\alpha - u_R \equiv (\hat{P}X)^T\hat{r} + K_v\alpha - u_R \tag{19}$$

where $\hat{P}$, $\hat{m}$, $\hat{v}$ and $\hat{\boldsymbol{r}}$ are the estimates of P^*, m^*, v^* and r^*, respectively. u_R is a robust term used to eliminate the approximation error ε.

Considering (18) and (19), the estimation error $\tilde{f}$ can be rewritten as

$$\begin{aligned}\tilde{f} &= f^*(x) - \hat{f}(x)\\ &= \left(P^*X\right)^T r^* + \varepsilon - \left(\hat{P}X\right)^T \hat{r} + u_R\\ &= X^T\tilde{P}^T r^* + X^T\hat{P}^T\tilde{r} + \varepsilon + u_R \end{aligned} \tag{20}$$

where $\tilde{P} = P^* - \hat{P}$ and $\tilde{r} = r^* - \hat{r}$. Moreover, this linearization technique is adopted to transform the receptive-field function into a quasi-linear form. According to the design method in [26], the Taylor series expansion of $\tilde{r}$ on $\tilde{m}$ and $\tilde{v}$ can be rewritten as

$$\tilde{r} = r^* - \hat{r} = r_m^T\tilde{m} + r_v^T\tilde{v} + O_t \tag{21}$$

where $\tilde{m} = m^* - \hat{m}$, $\tilde{v} = v^* - \hat{v}$, and $O_t \in R^{n_l}$ is a vector of higher-order terms.

Based on the nonlinear system demonstrated in (1), if the definition of the control law is designed as in (19), then TFC's adaptation laws are developed using (22) to (24), while (25) demonstrates the design of the robust controller.

$$\dot{\hat{P}} = \eta_1\Gamma\hat{r}X^T\alpha \tag{22}$$

$$\dot{\hat{m}} = \eta_2 r_m\hat{P}X\alpha \tag{23}$$

$$\dot{\hat{v}} = \eta_3 r_v\hat{P}X\alpha \tag{24}$$

$$u_R = -\mu\,\mathrm{sgn}(\alpha) \tag{25}$$

where $\mu = |\varepsilon + \tau_d|$ denotes the uncertainty bound; $\Gamma = \Gamma^T > 0$ is the adaptive gain matrix; and η_1, η_2, η_3 are the learning rates of $\dot{\hat{P}}$, $\dot{\hat{m}}$, $\dot{\hat{v}}$, respectively.

The Lyapunov function candidate is given by

$$L_2 = L_1 + \frac{1}{2\eta_1}tr\left(\tilde{P}^T\Gamma^{-1}\tilde{P}\right) + \frac{1}{2\eta_2}\tilde{m}^T\tilde{m} + \frac{1}{2\eta_3}\tilde{v}^T\tilde{v} \tag{26}$$

Taking the derivative of the Lyapunov function (26) and using (9) yields

$$\begin{aligned}\dot{L}_2 &= \dot{L}_1 + \frac{1}{\eta_1}tr\left(\tilde{P}^T\Gamma^{-1}\dot{\tilde{P}}\right) + \frac{1}{\eta_2}\tilde{m}^T\dot{\tilde{m}} + \frac{1}{\eta_3}\tilde{v}^T\dot{\tilde{v}}\\
&= \alpha^T M\dot{\alpha} + \frac{1}{2}\alpha^T\dot{M}\alpha + \frac{1}{\eta_1}tr\left(\tilde{P}^T\Gamma^{-1}\dot{\tilde{P}}\right)\\
&\quad + \frac{1}{\eta_2}\tilde{m}^T\dot{\tilde{m}} + \frac{1}{\eta_3}\tilde{v}^T\dot{\tilde{v}}\\
&= \alpha^T[-(K_v+V)\alpha + \tilde{f} + \tau_d] + \frac{1}{2}\alpha^T\dot{M}\alpha\\
&\quad + \frac{1}{\eta_1}tr\left(\tilde{P}^T\Gamma^{-1}\dot{\tilde{P}}\right) + \frac{1}{\eta_2}\tilde{m}^T\dot{\tilde{m}} + \frac{1}{\eta_3}\tilde{v}^T\dot{\tilde{v}}\\
&= -\alpha^T K_v\alpha + \frac{1}{2}\alpha^T(\dot{M} - 2V)\alpha + \alpha^T(\tilde{f} + \tau_d)\\
&\quad + \frac{1}{\eta_1}tr\left(\tilde{P}^T\Gamma^{-1}\dot{\tilde{P}}\right) + \frac{1}{\eta_2}\tilde{m}^T\dot{\tilde{m}} + \frac{1}{\eta_3}\tilde{v}^T\dot{\tilde{v}}\\
&= -\alpha^T K_v\alpha + \alpha^T(X^T\tilde{P}^T r^* + X^T\hat{P}^T\tilde{r} + \varepsilon + u_R + \tau_d)\\
&\quad + \frac{1}{\eta_1}tr\left(\tilde{P}^T\Gamma^{-1}\dot{\tilde{P}}\right) + \frac{1}{\eta_2}\tilde{m}^T\dot{\tilde{m}} + \frac{1}{\eta_3}\tilde{v}^T\dot{\tilde{v}}\\
&= -\alpha^T K_v\alpha + \alpha^T(X^T\tilde{P}^T\hat{r} + X^T\hat{P}^T\left(r_m^T\tilde{m} + r_v^T\tilde{v}\right) + \xi(t)\\
&\quad + u_R + \tau_d) + \frac{1}{\eta_1}tr\left(\tilde{P}^T\Gamma^{-1}\dot{\tilde{P}}\right) + \frac{1}{\eta_2}\tilde{m}^T\dot{\tilde{m}} + \frac{1}{\eta_3}\tilde{v}^T\dot{\tilde{v}}\\
&= -\alpha^T K_v\alpha + tr\left[\tilde{P}^T\left(\hat{r}X^T\alpha - \frac{1}{\eta_1}\Gamma^{-1}\dot{\hat{P}}\right)\right]\\
&\quad + \tilde{m}^T\left[r_m\hat{P}X\alpha - \frac{1}{\eta_2}\dot{\hat{m}}\right] + \tilde{v}^T\left[r_v\hat{P}X\alpha - \frac{1}{\eta_3}\dot{\hat{v}}\right]\\
&\quad + \alpha^T(\varepsilon + u_R + \tau_d)\\
&= -\alpha^T K_v\alpha + \alpha^T(\varepsilon + u_R + \tau_d)
\end{aligned} \tag{27}$$

In the expression above, the relationships $\alpha^T X^T\hat{P}^T r_m^T\tilde{m} = \tilde{m}^T r_m\hat{P}X\alpha$ and $\alpha^T X^T\hat{P}^T r_v^T\tilde{v} = \tilde{v}^T r_v\hat{P}X\alpha$ are used as scaling terms. Thus, the expression can also be defined as $\alpha X^T\tilde{P}^T\hat{r} = tr\left(\tilde{P}^T\alpha\hat{r}X^T\right)$. Then, via (25), (27) can be rewritten as

$$\begin{aligned}\dot{L}_2 &= -\alpha^T A\alpha + \alpha^T(\varepsilon + u_R + \tau_d)\\
&= -\alpha^T A\alpha + \alpha^T(\varepsilon + u_R + \tau_d)\\
&= -\alpha^T A\alpha + \alpha^T(\varepsilon + \tau_d) + \alpha^T u_R\\
&= -\alpha^T A\alpha + \alpha^T(\varepsilon + \tau_d) - \|\alpha\|\,\mu\\
&\le 0\end{aligned} \tag{28}$$

V. SIMULATION RESULTS

The two-link robot manipulator shown in Fig. 1 is used to examine the effectiveness of the proposed control scheme. The adopted robot system's dynamic model can be expressed in the form of Eq. (1) as in [19]. The specific system parameters for robot manipulators are as follows in (29) shown at the bottom of this page.

where q_1 and q_2 are the angle of joints 1 and 2, m_1 and m_2 are the mass of links 1 and 2, l_1 and l_2 are the length of links 1 and 2, and g is the gravity acceleration. Additionally, the following nonlinear viscous and dynamic friction terms of $F(\dot{q})$ and unknown disturbances τd have been covered in the manipulator dynamics.

$$\begin{aligned}&l_1 = 1.0\text{ m}\; l_2 = 1.0\text{ m}\; m_1 = 0.8\text{ kg}\\ &m_2 = 2.3\text{ kg}\quad g = 9.8\text{ m/s}^2\end{aligned} \tag{30}$$

$$F(\dot{q}) = 0.02\mathrm{sgn}(\dot{q})\quad \tau_d = [\,0.2\sin(t)\quad 0.2\sin(t)\,] \tag{31}$$

$$\begin{aligned}M(q) &= \begin{bmatrix} l_2^2m_2 + l_1^2(m_1+m_2) + 2l_1l_2m_2\cos(q_2) & l_2^2m_2 + l_1l_2m_2\cos(q_2)\\ l_2^2m_2 + l_1l_2m_2\cos(q_2) & l_2m_2\end{bmatrix}\\
V(q,\dot{q}) &= \begin{bmatrix} -l_1l_2m_2\dot{q}_2\sin(q_2) & -l_1l_2m_2(\dot{q}_1+\dot{q}_2)\sin(q_2)\\ m_2l_1l_2\sin(q_2) & 0\end{bmatrix}\\
G(q) &= \begin{bmatrix}(m_1+m_2)l_1g\cos(q_1) + l_2m_2\cos(q_1+q_2)\\ m_2l_2g\cos(q_1+q_2)\end{bmatrix}\end{aligned} \tag{29}$$

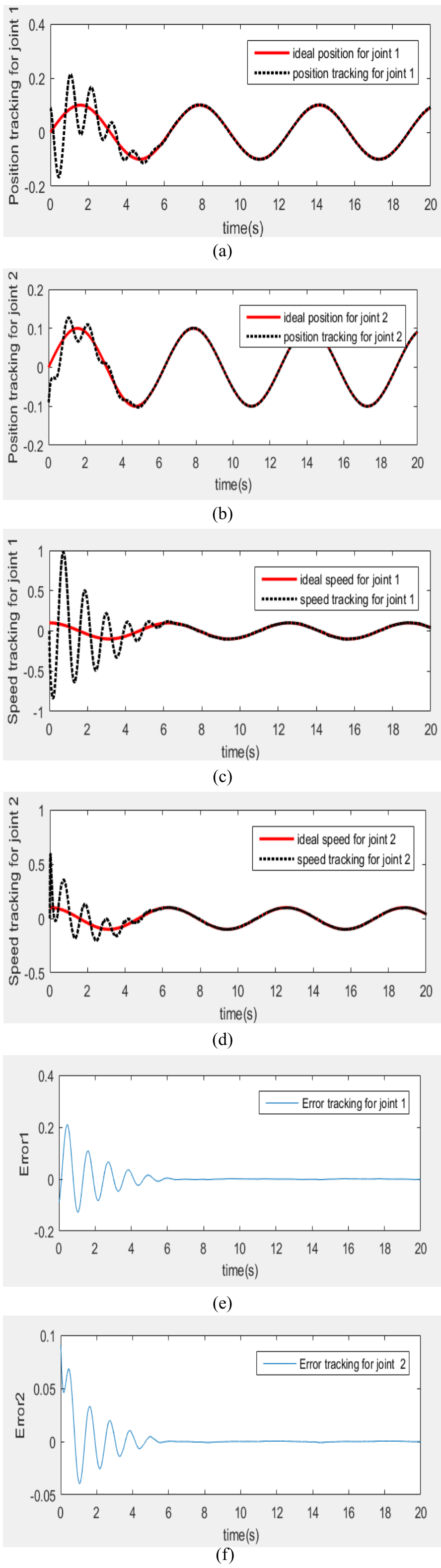


FIGURE 4. Simulated results of a computed torque position control system on a radial basis function neural network. (a) Position tracking for link 1; (b) Position tracking for link 2; (c) Speed tracking for link 1; (d) Speed tracking for link 2; (e) Tracking error for link 1; (f) Tracking error for link 2.

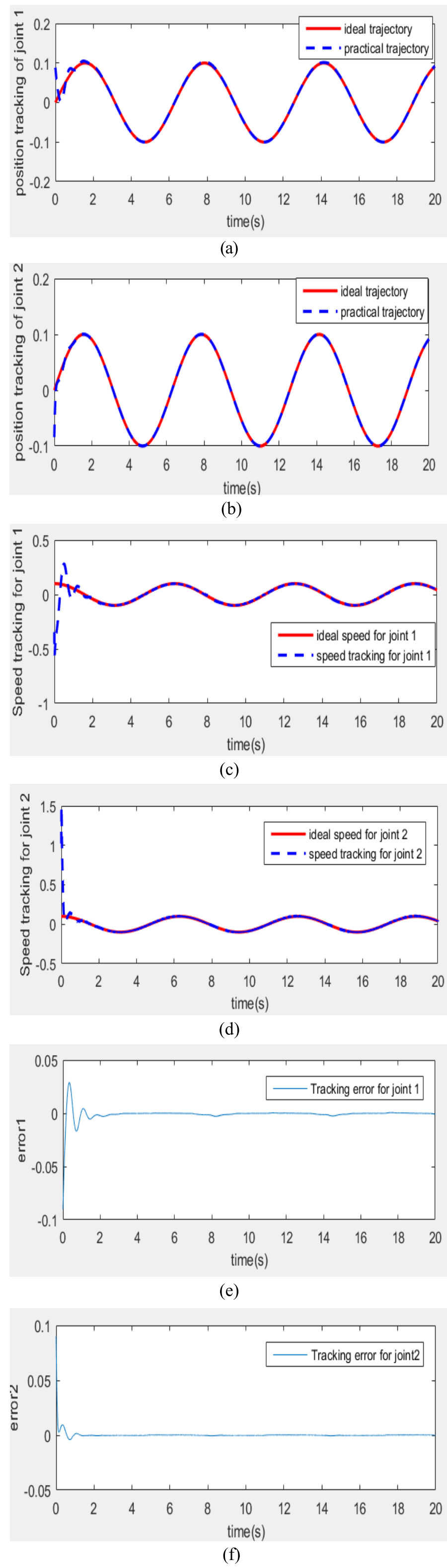


FIGURE 5. Simulated results of computed torque position control system on a cerebellar model articulation controller. (a) Position tracking for link 1; (b) Position tracking for link 2; (c) Speed tracking for link 1; (d) Speed tracking for link 2; (e) Tracking error for link 1; (f) Tracking error for link 2.

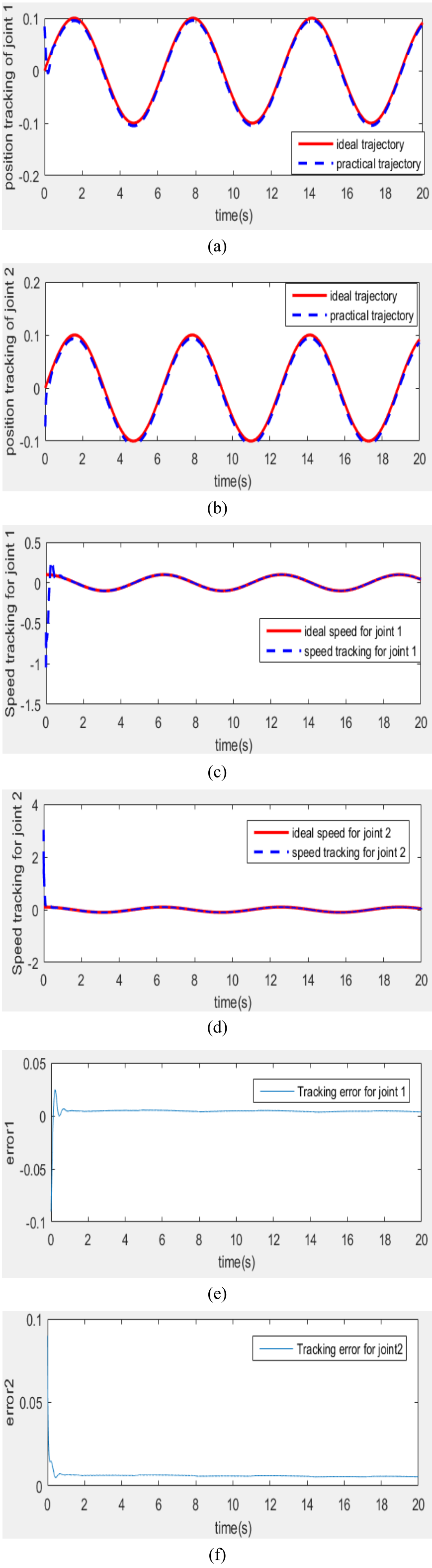


FIGURE 6. Simulated results of computed torque position control systems on a recurrent cerebellar model articulation controller. (a) Position tracking for link 1; (b) Position tracking for link 2; (c) Speed tracking for link 1; (d) Speed tracking for link 2; (e) Tracking error for link 1; (f) Tracking error for link 2.

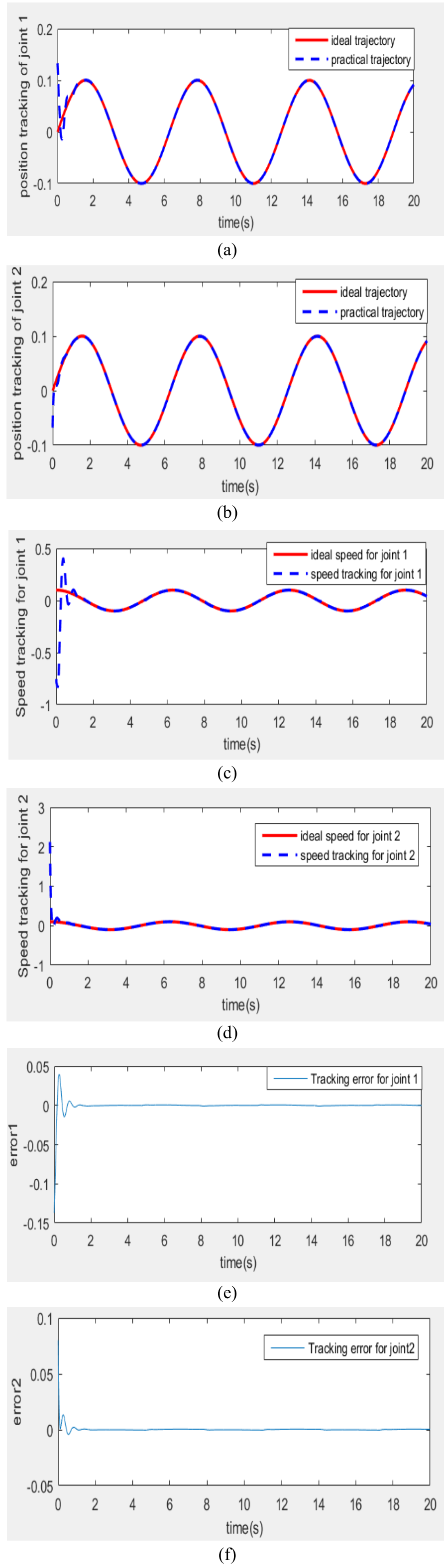
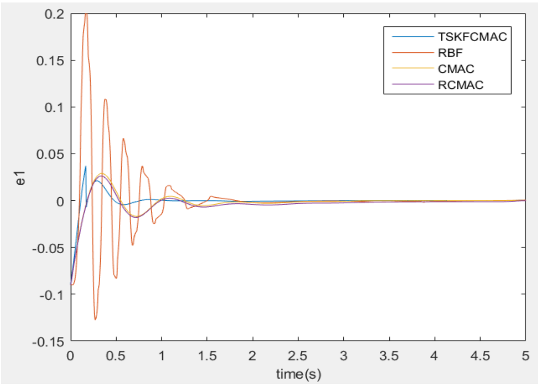
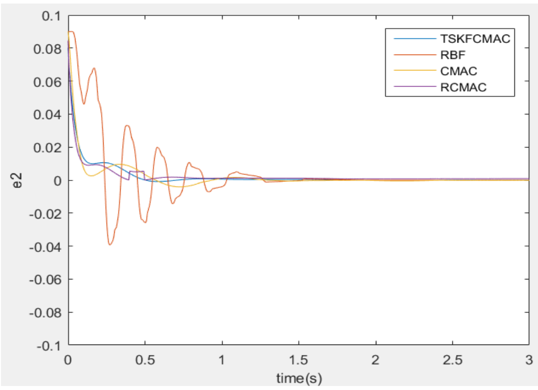


FIGURE 7. Simulated results of computed torque position control systems on TSK-based fuzzy cerebellar model articulation controller. (a) Position tracking for link 1; (b) Position tracking for link 2; (c) Speed tracking for link 1; (d) Speed tracking for link 2; (e) Tracking error for link 1; (f) Tracking error for link 2.



(a)



(b)

FIGURE 8. **Tracking error for joint 1 and 2 on four schemes. (a) Tracking error for joint 1. (b) Tracking error for joint 2.**

The initial state of the system is $[q_{1d}, \dot{q}_{1d}, q_{2d}, \dot{q}_{2d}]^T = [0.09\ 0\ -0.09\ 0]^T$, and the desired trajectory is expressed as

$$q_{1d}(t) = 0.1 \sin t \tag{32}$$

$$q_{2d}(t) = 0.1 \sin t \tag{33}$$

and $K_v = diag\{20, 20\}$, $\mu = 0.3$, $\Lambda = diag\{40, 40\}$.

The performance of the proposed adaptive TFCMAC control system is evaluated after it is applied on a two-link manipulator, as shown in Fig. 1. To confirm the superiority and the robustness of the TFCMAC control scheme, three other neural network control methods (radial basis function (RBF) neural network [18], CMAC [29] and recurrent CMAC [30]) are used to simulate and compare the position tracking and the speed tracking of the joints. Based on Fig. 4, Fig. 5 and Fig. 6, under the same coordinate presented Fig. 8, the error tracking of four intelligent control schemes were compared. Obviously, the robust adaptive TFCMAC is superior to the other three control strategies in the error convergence rate of the two joints.

Table 1 shows the root-mean-square error values obtained for the four investigated neural network controllers, which confirm once again that the robust adaptive TFCMAC is superior to the other three control schemes in the robot manipulator.

TABLE 1. **RMSE values of tracking performance.**

	RMSE/joint1	RMSE/joint2
RBF	0.0178	0.0079
CMAC	0.0094	0.0065
RCMAC	0.0063	0.0060
TFCMAC	0.0047	0.0032

VI. CONCLUSION

A robust adaptive TFCMAC control strategy has been presented and has effectively been used to solve inherent performance problems associated with manipulator-tracking control. The proposed control system consists of an adaptive TFCMAC and a robust compensator. The developed TFCMAC is applied as the main tracking controller, while the robust compensator is designed to mitigate the effects of approximation errors. Therefore, the desired tracking performance can be obtained. According to the simulation results, the proposed intelligent adaptive control scheme can ensure good tracking performance. Moreover, the quality of the performance such as the RMSE, the computing time and the convergence time, compared with RBF neural networks, CMAC and recurrent CMAC controllers exceeds expectations. One interesting future research topic is the extension of the proposed methods to networked control systems, as shown in [31]–[33], and large-scale fuzzy systems, as shown in [34] and [35].

ACKNOWLEDGMENT

The authors would like to thank the Editor-in-Chief, Associate Editor, and anonymous reviewers for their helpful comments which have greatly improved the paper, and to thank Dr. Yuan Sun and Dr. Zhixiong Zhong for constructive suggestions.

REFERENCES

[1] J. Alvarez-Ramirez, R. Kelly, and I. Cervantes, "Semiglobal stability of saturated linear PID control for robot manipulators," *Automatica*, vol. 39, no. 6, pp. 989–995, 2003.

[2] Z. Man, A. P. Paplinski, and H. R. Wu, "A robust MIMO terminal sliding mode control scheme for rigid robotic manipulators," *IEEE Trans. Autom. Control*, vol. 39, no. 12, pp. 2464–2468, Dec. 1994.

[3] Z. Q. Sun, *Intelligent Control of Robot*. Sanxi, Xi'an: Sanxi Edu. Press, 1995.

[4] F. Piltan, M. Mansoorzadeh, S. Zare, F. Shahryarzadeh, and M. Akbari, "Artificial tune of fuel ratio: Design a novel siso fuzzy backstepping adaptive variable structure control," *Int. J. Elect. Comput. Eng.*, vol. 3, no. 2, pp. 171–178, 2013.

[5] 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.

[6] A. M. Zou and K. D. Kumar, "Neural network-based distributed attitude coordination control for spacecraft formation flying with input saturation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 7, pp. 1155–1162, Jul. 2012.

[7] M. Seera, C. P. Lim, D. Ishak, and H. Singh, "Fault detection and diagnosis of induction motors using motor current signature analysis and a hybrid FMM–CART model," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 1, pp. 97–108, Jan. 2012.

[8] C.-M. Lin, A.-B. Ting, C.-F. Hsu, and C.-M. Chung, "Adaptive control for MIMO uncertain nonlinear systems using recurrent wavelet neural network," *Int. J. Neural Syst.*, vol. 21, no. 6, pp. 37–50, 2012.

[9] 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.

[10] S. Purwar, I. N. Kar, and A. N. Jha, "Adaptive output feedback tracking control of robot manipulators using position measurements only," *Expert Syst. Appl.*, vol. 34, no. 4, pp. 2789–2798, 2008.

[11] J. Liu and Y. Miao, "Research on trajectory control strategy of robot arm based on neural network compensation," *Control Decision*, vol. 7, no. 2, pp. 732–736, 2005.

[12] C. Y. Dong, L. Wang, and Q. Wang, "Decentralized adaptive control for robot manipulators based on neural network," *J. Syst. Simul.*, vol. 18, no. 5, pp. 1267–1270, 2006.

[13] F. C. Sun, Z. Q. Sun, and B. Zhang, *Theory and Approaches for Stable Adaptive Control of Robotic Manipulators Using Neural Networks*. Beijing, China: Higher Education Press, 2005.

[14] F. Sun, Z. Sun, and P.-Y. Woo, "Neural network-based adaptive controller design of robotic manipulators with an observer," *IEEE Trans. Neural Netw.*, vol. 12, no. 1, pp. 54–67, Jan. 2001.

[15] Y. Wei, J. Qiu, and H. R. Karimi, "Reliable output feedback control of discrete-time fuzzy affine systems with actuator faults," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 64, no. 1, pp. 170–181, Jan. 2017.

[16] Y. Wei, J. Qiu, H. K. Lam, and L. Wu, "Approaches to T–S fuzzy-affine-model-based reliable output feedback control for nonlinear Ito stochastic systems," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 3, pp. 569–583, Jun. 2017.

[17] R.-J. Wai and C. J. Chang, "Tracking control based on neural network strategy for robot manipulator," *Neurocomputing*, vol. 51, no. 7, pp. 425–445, 2003.

[18] R.-J. Lian, "Adaptive self-organizing fuzzy sliding-mode radial basis-function neural-network controller for robotic systems," *IEEE Trans. Ind. Electron.*, vol. 61, no. 3, pp. 1493–1503, Mar. 2014.

[19] J. S. Albus, "A new approach to manipulator control: The cerebellar model articulation controller (CMAC)," *J. Dyn. Syst., Meas. Control*, vol. 97, no. 3, pp. 220–233, 1975.

[20] C.-S. Chen, "Sliding-mode-based fuzzy CMAC controller design for a class of uncertain nonlinear system," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, San Antonio, TX, USA, Oct. 2009, pp. 3030–3034.

[21] F. L. Lewis, D. M. Dawson, and C. T. Abdallah, *Robot Manipulator Control: Theory and Practice*. Boca Raton, FL, USA: CRC Press, 2003.

[22] F. L. Lewis, K. Liu, and A. Yesildirek, "Neural net robot controller with guaranteed tracking performance," *IEEE Trans. Neural Netw.*, vol. 6, no. 3, pp. 703–715, May 1995.

[23] R.-J. Wai and L.-J. Chang, "Stabilizing and tracking control of nonlinear dual-axis inverted-pendulum system using fuzzy neural network," *IEEE Trans. Fuzzy Syst.*, vol. 14, no. 1, pp. 145–168, Feb. 2006.

[24] C.-J. Lin, C.-H. Chen, and C.-T. Lin, "A hybrid of cooperative particle swarm optimization and cultural algorithm for neural fuzzy networks and its prediction applications," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 39, no. 1, pp. 55–68, Jan. 2009.

[25] S.-H. Chen, W.-H. Ho, and J.-H. Chou, "Robust controllability of T–S fuzzy-model-based control systems with parametric uncertainties," *IEEE Trans. Fuzzy Syst.*, vol. 17, no. 6, pp. 1324–1335, Dec. 2009.

[26] P. C. Chang and C. Y. Fan, "A hybrid system integrating a wavelet and TSK fuzzy rules for stock price forecasting," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 38, no. 6, pp. 802–815, Nov. 2008.

[27] Z. Xi, G. Feng, and T. Hesketh, "Piecewise sliding-mode control for T–S fuzzy systems," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 4, pp. 707–716, Aug. 2011.

[28] Z. Zhong, R. Wai, Z. Shao, and M. Xu, "Reachable set estimation and decentralized controller design for large-scale nonlinear systems with time-varying delay and input constraint," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 6, pp. 1629–1643, Dec. 2017, doi: 10.1109/TFUZZ.2016.2617366.

[29] C.-M. Lin and Y.-F. Peng, "Adaptive CMAC-based supervisory control for uncertain nonlinear systems," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 2, pp. 1248–1260, Apr. 2004.

[30] C.-M. Lin, L.-Y. Chen, and C.-H. Chen, "RCMAC hybrid control for MIMO uncertain nonlinear systems using sliding-mode technology," *IEEE Trans. Neural Netw.*, vol. 18, no. 3, pp. 708–720, May 2007.

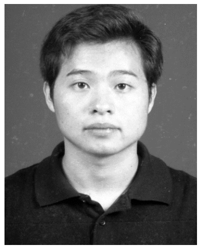
[31] J. S. Guan, L. Y. Lin, and G. L. Ji, "Breast tumor computer-aided diagnosis using self-validating cerebellar model neural networks," *Acta Polytechn. Hungarica*, vol. 13, no. 4, pp. 39–52, 2016.

[32] J. Qiu, Y. Wei, and L. Wu, "A novel approach to reliable control of piecewise affine systems with actuator faults," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 64, no. 8, pp. 957–961, Aug. 2017.

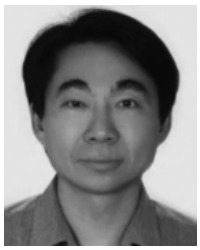
[33] Y. Wei, J. Qiu, H. R. Karimi, and M. Wang, "New results on H_∞ dynamic output feedback control for Markovian jump systems with time-varying delay and defective mode information," *Optim. Control Appl. Methods*, vol. 35, no. 6, pp. 656–675, 2014.

[34] Z. Zhong, C. Lin, Z. Shao, and M. Xu, "Decentralized event-triggered control for large-scale networked fuzzy systems," *IEEE Trans. Fuzzy Syst.*, to be published, doi: 10.1109/TFUZZ.2016.2634090.

[35] Z. Zhong, Y. Zhu, and H.-K. Lam, "Asynchronous piecewise output-feedback control for large-scale fuzzy systems via distributed event-triggering schemes," *IEEE Trans. Fuzzy Syst.*, doi: 10.1109/TFUZZ.2017.275100.



JIANSHENG GUAN was born in China in 1981. He received the B.S. and M.S. degrees from the Department of Automation, Xiamen University, where he is currently pursuing the Ph.D. degree in control science and engineering with the Institute of Aeronautics and Astronautics. He is currently an Associate Professor with the Xiamen University of Technology. His research interests include intelligent control systems, neural networks, cerebellar model articulation controllers, and complex-process control systems.

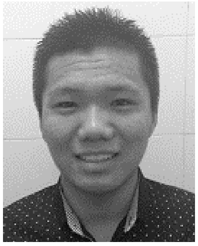


CHIH-MIN LIN (M'87–SM'99–F'10) was born in Taiwan in 1959. He received the B.S. and M.S. degrees from the Department of Control Engineering and the Ph.D. degree from the Institute of Electronics Engineering, National Chiao Tung University, Hsinchu, Taiwan, in 1981, 1983, and 1986, respectively. He is currently a Chair and the Dean of the College of Electrical and Communication Engineering at Yuan Ze University, Zhongli, Taiwan. He has published 158 journal papers and 159 conference papers. His current research interests include fuzzy neural networks, cerebellar model articulation controllers, intelligent control systems, and signal processing.

Dr. Lin was a recipient of the Honorary Research Fellowship at the University of Auckland, Auckland, New Zealand, from 1997 to 1998. He also serves as an Associate Editor of the IEEE TRANSACTIONS ON CYBERNETICS, the *Asian Journal of Control*, the *International Journal of Fuzzy Systems*, and the *International Journal of Machine Learning and Cybernetics*.

GUO-LI JI received the B.S. degree in automation control and the M.S. degree in systems engineering from Xi'an Jiaotong University, Xi'an, China, in 1982 and 1986, respectively.

He is currently a Full Professor with the Department of Automation, Xiamen University, Xiamen, China. He is also the President of the Institute of Systems Engineering and Xiamen Association of Systems Engineering. He has led and participated in many research projects from the Natural Science Foundation of China, the Natural Science Foundation of Fujian, and others. Recently, he has published papers in the IEEE TRANSACTIONS ON HUMAN–MACHINE SYSTEMS, *Expert Systems With Applications*, *Knowledge-Based Systems*, the *Journal of Process Control*, *Bioinformatics*, *BMC Biotechnology*, *Genome Research*, *Proceedings of the National Academy of Sciences* of the USA, and *Cell Research*.



LING-WU QIAN was born in China in 1994. He received the B.S. degree from the Department of Automation, School of Electrical Engineering and Automation, Xiamen University of Technology. He is currently pursuing the M.S. degree in control engineering with the Institute of Aeronautics and Astronautics, Xiamen University. His research interests are bioinformatics and information science.

YI-MIN ZHENG was born in China in 1977. He received the M.S. degree from the Institute of Mechanical and Electrical Engineering, Huaqiao University. He is currently pursuing the Ph.D. degree in control science and engineering at the Institute of Aeronautics and Astronautics, Xiamen University. He is currently a Lecturer at Huaqiao University. His research interests include system identification, industrial process control, and decoupling control.