Self-organizing fuzzy control of multi-variable systems using learning vector quantization network

Wei-Song Lin *, Chih-Hsin Tsai

Department of Electrical Engineering, National Taiwan University, No. 1, Sec. 4, Roosevelt Rd, Taipei 106, Taiwan, ROC

Abstract

Using learning vector quantization (LVQ) network to construct a self-organizing fuzzy controller (SOFC) for multi-variable nonlinear composite systems is developed in this paper. The LVQ network is used to provide information about the better locations of the IF-part membership functions through un-supervised learning. The generated fuzzy rule base is applied to the SOFC and updated by a self-learning procedure. Using Lyapunov stability methods, the proposed adaptive scheme is proven to provide the SOFC some degree of robust properties and guarantee uniform ultimate boundedness in the presence of disturbances, measurement noise and perturbed initialization error. The effectiveness of the proposed controller has been demonstrated numerically by applying to control a two-link manipulator. © 2001 Elsevier Science B.V. All rights reserved.

Keywords: Fuzzy control; Self-organizing; Multi-variable; Nonlinear

1. Introduction

Fuzzy logic has emerged as a practical and successful alternative in the control of complex or ill-defined systems [17]. Conventionally, the fuzzy inference is based on rules constructed according to the experience of experts [2,14]. But there are situations, due to safety, complexity or beyond reach that the experts are not allowed to experience the detailed changes of the dynamic system to extract the fuzzy rules. Or sometimes the constructed fuzzy rule base can only support the experienced conditions other than new environment of the system. Procyk and Mamdani [12] addressed this problem by introducing a linguistic self-organizing controller, which is capable of generating and modifying the control protocol by a learning process based on performance measurement. Recently, many other researchers focused on combining the learning ability of neural network with fuzzy logic system to create/adapt the proper fuzzy rule base [3,4,7,9]. Lin [3] and Lee [4]

---

* Corresponding author. Tel.: +886-2-3635251 ext. 413; fax: +886-2-3638247.

E-mail address: weisong@cc.ee.ntu.edu.tw (W.-S. Lin).

proposed a reinforcement neural-network-based fuzzy logic control system, which not only performs the fuzzy inference operations but also extracts the fuzzy rules and membership functions from learning examples. Jang [7] presented an enhanced fuzzy controller with temporal backpropagation methodology, which can either refine or automatically derive the fuzzy if-then rules. Nie and Linkens [10,11] proposed a fuzzified CMAC network and a fuzzified RBF network to act as adaptive controllers with the feature of self-organizing association cells and the further ability of self-learning the required teacher signals in real-time. Despite these considerable achievements, the development of systematic approaches, which perform self-organization of control knowledge, is still far from being completely and satisfactorily resolved. Specifically, in the cases of multi-variable nonlinear control systems, few researchers discuss the stability of the closed-loop system and the convergence of the tracking error in the procedure of self-organization and self-learning.

In this paper, a self-organizing fuzzy controller (SOFC), capable of self-organizing its own structure and self-learning the required knowledge to control multi-variable nonlinear composite systems, is proposed. Design of the SOFC is a systematic approach that uses the learning vector quantization network (LVQ) [6,8] in the self-organizing level and incorporates a robust adaptation scheme in the task execution level. A LVQ network is trained to obtain the information about the better locations of the IF-part membership functions. With the generated fuzzy rule base, the proposed robust adaptive scheme is proven to provide the SOFC some degree of robust properties and guarantee uniform ultimate boundedness in the presence of disturbances, measurement noise and perturbed initialization error.

This paper is organized as follows. In Section 2, the SOFC system and its architecture is presented. The self-organizing level using LVQ network and the coordination level are proposed in Sections 3.1 and 3.2, respectively. In Section 3.3, the task execution level is described. The robust parameter adaptation scheme is proposed in Section 3.4. Also, in this section, the robust property and the convergence of output tracking error are analyzed. In Section 4, the performance of controlling a two-link robot arm carrying a heavy load is evaluated and demonstrated. Section 5 is the conclusion.

2. The self-organizing fuzzy control system

Consider a nonlinear composite plant, such as a robot, governed by

$$
\begin{aligned}
&y^{(r)} = f(x) + G(x)u + z(x,t), \\
&\bar{x} = x + n_x, \\
&\bar{y} = y + n_y,
\end{aligned}
\tag{1}
$$

where $y = [y_1, \ldots, y_m]^T$, $\bar{y}, x = [x_1, \ldots, x_n]^T = [y_1, \ldots, y_1^{(r_1-1)}, \ldots, y_m, \ldots, y_m^{(r_m-1)}]^T$, and $\bar{x}(\cdot)$ denote the output, measured output, state and measured state vectors, respectively, $y^{(r)} \equiv [y_1^{(r_1)}, y_2^{(r_2)}, \ldots, y_m^{(r_m)}]^T$, $r = [r_1, \ldots, r_m]$ and $\sum_{i=1}^{m} r_i = n$ denote the plant relative degree, $u = [u_1, \ldots, u_m]^T$ is the plant input, $f = [f_1, \ldots, f_m]^T$, $G =$ Block diag$[g_1, \ldots, g_m]$ and f_i and g_i are smooth functions, $z(x,t) = [z_1(x,t), \ldots, z_m(x,t)]^T$ denotes the aggregation of unknown interaction and time-varying disturbance. The exogenous signals $n_x = [n_{x1}, \ldots, n_{xn}]^T$, $n_y = [n_{y1}, \ldots, n_{yn}]^T$ and $z(x,t)$ are assumed to have the properties of standard smoothness and boundedness. Additionally, the exogenous signals are assumed to satisfy $n_{yi} \in C^{r_i}$. Let y_{Mi}, and v_i denote the reference output and input, respectively. The aim of control is to make each subsystem of (1) asymptotically tracking a linear reference model of the following form:

$$
y_{Mi}^{(r_i)} = \alpha_{i1} y_{Mi} + \alpha_{i2} \dot{y}_{Mi} + \cdots + \alpha_{ir_i} y_{Mi}^{(r_i-1)} + v_i,
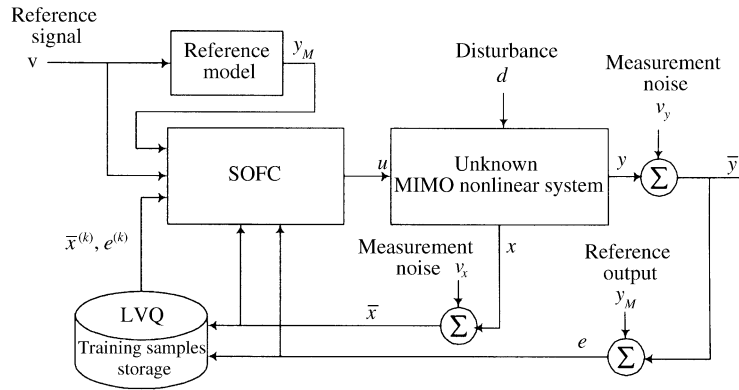\tag{2}
$$

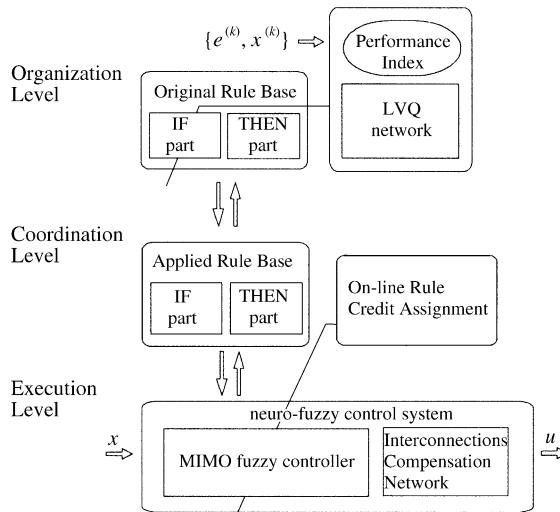Fig. 1. A robust multi-variable tracking control system using the SOFC.

Fig. 2. Diagrammatic representation of the hierarchical SOFC.

where the constants $\alpha_{i1}, \ldots, \alpha_{ir_i}$ are selected so that (2) is asymptotically stable. Since the plant operates repetitively in the presence of disturbance, measurement noise and perturbed initialization error, the ability of self-organization of the proposed controller is critical.

Fig. 1 shows the architecture of the proposed SOFC for robust tracking control of repetitive nonlinear plants described by (1). The learning process is comprised of a "training samples storage" that allows the controller to store measured state variable samples in each trial. The SOFC is a hierarchically intelligent control system [13]. Fig. 2 shows the conceptual structure of the hierarchical SOFC that is composed of three main levels of intelligence. The three levels are

(1) the organization level represents the brain of the system with functions dominated by a LVQ network to plan and make decisions about the fuzzy rule base;

(2) the coordination level is the interface between high and low levels of intelligence with functions coordinating the original rule base and the robust parameter adaptation law for the SOFC; and

(3) the execution level is the lowest level that requires high precision and has functions dominated by multi-layer fuzzy control system.

The details of each level will be described in the following sections.

3. Design of the self-organizing fuzzy controller

Since fuzzy controllers are usually realized by using digital input and output, a particular class of fuzzy systems with the singleton fuzzified, algebraic product T-norm, the sup star compositional operator [15] and the local mean-of-maximum [1] method is considered.

3.1. Organization level

3.1.1. Fuzzy rule base

A multivariable system can be controlled by the following $M + N + 1$ linguistic rules

$$R^j: \text{ IF } x_1 \text{ is } A_1^j \text{ AND} \ldots \text{AND } x_n \text{ is } A_n^j$$

$$\text{THEN } u_1 \text{ is } B_1^j \text{ AND} \ldots \text{AND } u_m \text{ is } B_m^j \quad \text{for } j = 1, \ldots, M + N + 1,$$

where M denotes the number of regular rule partitions and N denotes the number of rule partitions positioned near the trial trajectory. The fuzzy sets A_k^j and B_i^j are linguistic terms characterized by the fuzzy membership functions

$$\mu_{A_k^j}(x_k) = \exp(-(x_k - m_k^j)^2/a_k^j) \tag{3}$$

and

$$\mu_{B_i^j}(u_i) = \begin{cases} (1 + ((c_i^j - u_i)/a_{Li})^2)^{-1}, & \text{if } u_i \leqslant c_i^j, \\ (1 + ((u_i - c_i^j)/a_{Ri})^2)^{-1}, & \text{if } u_i > c_i^j, \end{cases} \tag{4}$$

where $\{a_k^j, m_k^j\}$ and $\{a_{Li}, a_{Ri}, c_i^j\}$ are referred to the premise and consequence parameters, respectively.

In most fuzzy logic control systems, the fuzzy rule space is partitioned into a number of domains of equivalent sizes, called regular rule partition as shown in Fig. 3(a), and a rule is stored in each domain. If higher tracking accuracy is desired, a partition consisting of small intervals is required. However, the problem of computational complexity will worsen as the dimensionality of the state space, n, increases. In this paper, only a few, M, fuzzy rules are constructed according to the regular rule partition to ensure that the ability to perform basic function approximation is presented in the complete space. In addition, the N rule partitions are positioned near the trial trajectory, as shown in Fig. 3(b), to achieve the desired tracking accuracy. This results in a significant reduction in the number of fuzzy rules. In this section, a new scheme using a LVQ network is proposed to accomplish this task.

3.1.2. Locating of membership functions by LVQ network

Vector quantization is a classical method that produces an approximation to a continuous probability density function $p(\mathbf{x})$ of the vector input variable $\mathbf{x}$ using a finite number of neurons with weights $\mathbf{m}^j = [m_1^j, m_2^j, \ldots, m_n^j]^T$, $j = 1, \ldots, N$. One kind of optimal placement of the $\mathbf{m}^j$ is to minimize E, the expected rth power of the reconstruction error:

$$E = \int \|\mathbf{x} - \mathbf{m}_w^j\|^r p(\mathbf{x}) \, d\mathbf{x}, \tag{5}$$
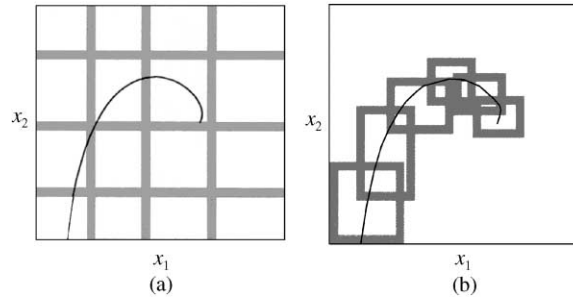
Fig. 3. (a) Regular rule partition, (b) rule partitions depend on the trial trajectory.

where $d\mathbf{x}$ is the volume differential in the $\mathbf{x}$ space, and the index $w = w(\mathbf{x})$ of the best-matching neuron vector ("winner") is a function of the input vector $\mathbf{x}$:

$$\|\mathbf{x} - \mathbf{m}_w^j\| = \min_j \{\|\mathbf{x} - \mathbf{m}^j\|\}. \tag{6}$$

Once the LVQ network is trained, the neurons will be appropriately associated with their weights $\mathbf{m}^j$, and each of which corresponds to the location of IF-part membership function as in (3). To implement this idea in the sense of discrete-time notation, consider a controlled system with measured state variable samples $\{\bar{\mathbf{x}}(T_i)\}$, which are generated at every sampling time T_i. In the first operation, one can obtain a sequence of them by the given reference model. For consecutive trials the state samples can be obtained from the "training samples storage" which is generated by the response of the controlled plant. Defining a neighborhood set N_w that shrinks monotonically with time, the updating process may read

$$\mathbf{m}^j(k+1) = \begin{cases} \mathbf{m}^j(k) + \alpha(k)(\bar{\mathbf{x}}'(k) - \mathbf{m}^j(k)), & \text{if } j \in N_w, \\ \mathbf{m}^j(k), & \text{if } j \notin N_w, \end{cases} \tag{7}$$

with $\alpha(k)$ being a suitable, monotonically decreasing sequence of scalar-valued "adaptation gain", $0 < \alpha(k) < 1$ and $\bar{\mathbf{x}}'(k)$ is the randomly selected sample from $\{\bar{\mathbf{x}}(T_i)\}$.

3.1.3. Providing samples with conditional weights

In order to achieve higher tracking accuracy, the rule space around the region of large tracking error should be partitioned with smaller intervals to obtain better function approximation. This can be done by LVQ network if each sample, $\bar{\mathbf{x}}(T_i)$, is provided with conditional weight that depend on the following performance index

$$J_i = \frac{\|\mathbf{y}_M(T_i) - \bar{\mathbf{y}}(T_i)\|^2}{\sum_i \|\mathbf{y}_M(T_i) - \bar{\mathbf{y}}(T_i)\|^2}, \tag{8}$$

where $\mathbf{y}_M(T_i) - \bar{\mathbf{y}}(T_i)$ denotes the tracking error at sampling time T_i. Then, through a roulette wheel with slots weighted in proportion to the performance index J_i, the training samples $\{\bar{\mathbf{x}}(T_i)\}$ are presented to the LVQ network. The idea of roulette wheel is inspired by the reproduction process in the genetic algorithms [5].

3.2. Coordination level

The intermediate structure, i.e. the coordination level, serves as an interface between the organization and execution levels. It dispatches an organized rule base, gives rule credit assignment, and presents robust parameter adaptation laws to the execution level. The parameter adaptation law is analyzed in Section 3.4.

3.2.1. Rule credit assignment

The basic idea of the rule credit assignment is to reward good rules by increasing the confidence of the consequent fuzzy sets and the recommended fuzzy output of this rule. Denote $\omega_i^j > 1$ (or $\omega_i^j < 1$) as a reward (or a punishment) offered to the jth rule in the ith knowledge rule base, then the consequent membership function (4) can be reshaped into

$$
\mu_{\tilde{B}_i^j}(u_i) = \begin{cases} (1 + (\omega_i^j(c_i^j - u_i)/a_{Li})^2)^{-1}, & \text{if } u_i \leqslant c_i^j, \\ (1 + (\omega_i^j(u_i - c_i^j)/a_{Ri})^2)^{-1}, & \text{if } u_i > c_i^j, \end{cases} \tag{9}
$$

and the recommended fuzzy output of each rule is determined in singleton form as follows,

$$
\omega_i^j \cdot I(\mu^j(\bar{\pmb{x}}), \mu_{\tilde{B}_i^j}(u_i)) = \begin{cases} \omega_i^j \cdot \mu^j(\bar{\pmb{x}}), & \text{for } u_i = \tilde{c}_i^j, \\ 0 & \text{otherwise,} \end{cases} \tag{10}
$$

where "$\cdot$" in the multiplication operation, I is the implication function [15], $\mu^j(\bar{\pmb{x}}) = \mu_{A_1^j}(\bar{x}_1) \cdots \mu_{A_n^j}(\bar{x}_n)$ denotes the matching degree, respectively, and $\tilde{c}_i^j$ denotes the location of the singleton implication fuzzy set defined as

$$
\tilde{c}_i^j = \text{the centroid of the set } \{u_i: \ \mu_{\tilde{B}_i^j}(u_i) \geqslant \mu^j(\bar{\pmb{x}})\}. \tag{11}
$$

Using (9) and (11) can be resolved into

$$
\tilde{c}_i^j = c_i^j - a_{LRi}\mu^j \sqrt{(\mu^j)^{-1} - 1}/\omega_i^j, \tag{12}
$$

where $a_{LRi} = (a_{Li} - a_{Ri})/2$.

The study of assigning rule credits may be complicated, where the modification of control rules is achieved by giving a credit or reward value to individual rules engaged in the problem solving process. Generally, for a fuzzy/neural system, these parameters are updated depending on its output value and the associated teacher signals. Since no direct error measurement of the neural/fuzzy system is possible in the considered control system, the teacher signals are not available and only the error information between the plant and the desired trajectory can be used. In this paper, our approach is to treat the entire problem in the context of Lyapunov-based adaptive systems theory. As shown in Section 3.4, following the context of adaptive control technique, we provide on-line tuning rules for ω_i^j.

3.3. Execution level

Using the center average defuzzification, the output response of the fuzzy controller is

$$
u_i(t) = F_i(\bar{\pmb{x}}, \omega_i^j, a_{LRi}) = \frac{\sum_{j=1}^{M+N+1} \omega_i^j \cdot \mu^j \cdot \tilde{c}_i^j}{\sum_{j=1}^{M+N+1} \omega_i^j \cdot \mu^j}. \tag{13}
$$

In the rule base, the $(M + N + 1)$th rule is chosen to be of Takagi–Sugeno type and its consequent fuzzy set B_i^{M+N+1} is singleton with support represented as the form of the synthesis input

$$
c_i' = \alpha_{i1}\bar{y}_i + \alpha_{i2}\dot{\bar{y}}_i + \cdots + \alpha_{ir_i}\bar{y}_i^{(r_i-1)} + v_i. \tag{14}
$$

The curvature control parameter, a_k^{M+N+1}, of its antecedent membership function is assumed to approach infinity so that this rule will be fired whatever $\bar{\pmb{x}}$ is. The credit assignment takes place in rules R^j, $j = 1, \ldots, M + N$ but assigned to be 1 for R^{M+N+1}. Accordingly, using (12) and (14), the analytical formulation of the multi-layer fuzzy system in Eq. (13) resolve into

$$
u = \hat{D}^{-1}(-\theta'^{\mathrm{T}}\mu' + c' - a_{LR}\phi), \tag{15}
$$

where $\hat{D} = \text{Block diag}(\omega_1^T \mu, \ldots, \omega_m^T \mu)$, ω_i and μ are $(M + N + 1) \times 1$ column vectors composed of ω_i^j and μ^j, $\theta' = [\theta'_1, \ldots, \theta'_m] \in R^{N \times m}$, θ_i and μ' are $(M + N) \times 1$ column vectors composed of $\omega_i^j c_i^j$ and μ^j, $c' = [c'_1, \ldots, c'_m]^T$, $a_{LR} = [a_{LR1}, \ldots, a_{LRm}]^T$, and $\phi = \sum_{j=1}^{M+N+1} \mu^j \sqrt{(\mu^j)^{-1} - 1}$.

3.4. Parameters adaptation and performance analysis

Let $\theta_i = [\theta_i'^T, \omega_i^T]^T$, $\boldsymbol{M}_{\theta_i} = \{\boldsymbol{\theta}_i(t): |\boldsymbol{\theta}_i(t)| \leqslant \boldsymbol{\theta}_{i,\text{Max}}\}$ be the bounds of θ_i, and

$$\theta_i'^* \equiv \arg \min_{\theta'_i \in \boldsymbol{M}_{\theta_i}} [\sup |f_i(\boldsymbol{x}) - \theta_i'^T \mu'(\boldsymbol{x})|].$$

$$\omega_i^* \equiv \arg \min_{\omega_i \in \boldsymbol{M}_{\theta_i}} [\sup |g_i(\boldsymbol{x}) - \omega_i^T \mu(\boldsymbol{x})|],$$

be the best function approximation parameters. The adjustable parameter a_{LRi} in (15) represents the difference between the left and right spreads of the consequent membership functions. In the conventional fuzzy logic control systems, a_{Li} is set to be equivalent to a_{Ri} or the consequent membership is just in singleton form [15]. In this paper, the SOFC is developed to facilitate robust property by tuning the parameter a_{LRi}, i.e. tuning the consequent membership functions of the fuzzy system. By this on-line tuning mechanism, the fuzzy system can effectively deal with the lump disturbances, measurement noise, and interconnection compensation among subsystems that is discussed in greater detail in [16]. The parameter a_{LRi} is chosen as $a_{LRi}(\vartheta_i) = \vartheta_i \tanh(b_i^T P_i e_i \phi(\bar{\boldsymbol{x}})/\varepsilon)$ where ϑ_i is an auxiliary parameter and ε is a small positive constant.

Assumption 1. *There exists the smallest non-negative parameter values $\vartheta_i^* \geqslant 0$ such that for all $\bar{\boldsymbol{x}} \in \Re^n$ and $t \in \Re_+$*

$$|\zeta_i| \leqslant \vartheta_i^* \phi(\bar{\boldsymbol{x}}) \tag{16}$$

where ζ_i (see Appendix A) denotes the equivalent uncertainties which lump disturbance, measurement noise and interconnection effect among subsystems together.

Let $\boldsymbol{M}_{\vartheta_i} = \{\vartheta_i: |\vartheta_i| < \vartheta_{i,\text{max}}\}$ be the bound of ϑ_i, $\boldsymbol{M}_{\vartheta_i}^\varepsilon$ (or $\boldsymbol{M}_{\theta i}^\varepsilon$) be the union of $\boldsymbol{M}_{\vartheta_i}$ (or $\boldsymbol{M}_{\theta i}$) and its boundary layer of thickness ε_ϑ (or ε_θ), $\theta_{i\perp} = \theta_i/|\theta_i|$ be the unit normal vector, and the prefix ∂ denotes the boundary. Denote $e_i = [y_i - y_{id}, \dot{y} - \dot{y}_{id}, \ldots, y_i^{(r_i-1)} - y_{id}^{(r_i-1)}]^T$ and $\boldsymbol{e} = [e_1^T, \ldots, e_m^T]^T$. A robust adaptive algorithm for θ_i and ϑ_i motivated by an attempt to provide treatment to the equivalent uncertainties are proposed as follows

$$\dot{\theta}_i(t) = \begin{cases} 0, & \text{if } \boldsymbol{e}^T \boldsymbol{P} \boldsymbol{b} \boldsymbol{b}^T \boldsymbol{P} \boldsymbol{e} \leqslant d_0^2, \\ (I - d_{\theta_i} \theta_{i\perp} \theta_{i\perp}^T) R_i^{-1} [b_i^T P_i e_i \boldsymbol{w} - \sigma_1(\theta_i - \theta_{i0})], & \text{otherwise}, \end{cases} \tag{17}$$

with

$$d_{\theta_i} = \begin{cases} 0, & \text{if } \theta_{i\perp}^T [b_i^T P_i e_i \boldsymbol{w} - \sigma_1(\theta_i - \theta_{i0})] \leqslant 0, \\ \min[1, \text{dist}(\theta_i, \boldsymbol{M}_{\theta_i})/\varepsilon_\theta], & \text{otherwise}, \end{cases} \tag{18}$$

and

$$\dot{\vartheta}_i(t) = \begin{cases} 0, & \text{if } \boldsymbol{e}^T \boldsymbol{P} \boldsymbol{b} \boldsymbol{b}^T \boldsymbol{P} \boldsymbol{e} \leqslant d_0^2, \\ (1 - d_{\vartheta_i}) r_{\vartheta i}^{-1} [w'_i b_i^T P_i e_i - \sigma_2(\vartheta_i - \vartheta_{i0})], & \text{otherwise}, \end{cases} \tag{19}$$

with

$$d_{\vartheta_i} = \begin{cases} 0, & \text{if } \vartheta_i[b_i^{\mathrm{T}}P_ie_iw_i' - \sigma_2(\vartheta - \vartheta_0)] \leqslant 0, \\ \min[1, \mathrm{dist}(\vartheta_i, \boldsymbol{M}_{\vartheta_i})/\varepsilon_\vartheta], & \text{otherwise,} \end{cases} \tag{20}$$

$$w_i' = \phi \tanh\left(\frac{b_i^{\mathrm{T}}P_ie_i\phi(\bar{\boldsymbol{x}})}{\varepsilon}\right), \tag{21}$$

where P_i is a symmetric positive-definite matrix satisfying the Lyapunov equation $A_i^{\mathrm{T}}P_i + P_iA_i = -Q_i$, with the design parameters $Q_i > 0$, and σ_1 and σ_2 are chosen small but positive constant to keep θ_i and ϑ_i from growing unbounded.

Theorem 1. *Consider the nonlinear composite system* (1) *with the SOFC* (15), *the parameter adaptation schemes* (17) *and* (19) *operating in the bounded state* $\boldsymbol{x} \in \Omega$. *Then*
(1) θ_i, ϑ_i *and the control input* $\boldsymbol{u}$ *are uniformly ultimately bounded.*
(2) *Given any* ρ *satisfying* $\rho^* < \rho$, *where*

$$\rho^* = \frac{\sum_{i=1}^m[\sigma_1(\theta_i^* - \theta_{i0})^{\mathrm{T}}(\theta_i^* - \theta_{i0}) + \sigma_2(\vartheta_i^* - \vartheta_{i0})^2 + 2\kappa\vartheta_i^M\varepsilon]}{\min_i \min\{\lambda_{\min}(Q_i')/\lambda_{\max}(P_i), \sigma_1/\lambda_{\max}(R_i), \sigma_2/\gamma_\vartheta\}} \tag{22}$$

with $\vartheta_i^M \equiv \max\{\vartheta_i^*, \vartheta_{i0}\}$ *and* κ *being a constant that satisfies* $\kappa = \mathrm{e}^{-(\kappa+1)}$, *i.e.* $\kappa = 0.2785$, *there exists* T *such that for* $T \leqslant t \leqslant \infty$ *the tracking error* $\boldsymbol{e}$ *converges to the residual set*

$$\{\boldsymbol{e}: \boldsymbol{e}^{\mathrm{T}}\boldsymbol{Pe} \leqslant \rho \text{ or } \boldsymbol{e}^{\mathrm{T}}\boldsymbol{Pbb}^{\mathrm{T}}\boldsymbol{Pe} \leqslant d_0^2\}. \tag{23}$$

Proof. Refer to Appendix A for details. □

Remark 1. From (22), the tracking error residual is determined by the design parameter ρ^*. If the design constants ε, σ_1, σ_2, γ_ϑ, Q_i, P_i and R_i are appropriately chosen, it is possible to make ρ^* as small as desired and therefore better tracking performance can be achieved.

4. Simulation

A two-link robot manipulator is simulated to show the effectiveness of the proposed SOFC. The equations of motion of the robot can be expressed in matrix form as follows:

$$\begin{bmatrix} (m_1 + m_2)r_1^2 + m_2r_2^2 + 2m_2r_1r_2c_2 + J_1 & m_2r_2^2 + m_2r_1r_2c_2 \\ m_2r_2^2 + m_2r_1r_2c_2 & m_2r_2^2 + J_2 \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix}$$

$$+ \begin{bmatrix} -m_2r_1r_2s_2\dot{q}_1(\dot{q}_1 + \dot{q}_2) \\ m_2r_1r_2s_2\dot{q}_2^2 \end{bmatrix} + \begin{bmatrix} ((m_1 + m_2)l_1c_2 + m_2l_2c_{12})g \\ (m_2l_2c_{12})g \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \begin{bmatrix} d_1 \\ d_2 \end{bmatrix}, \tag{24}$$

where m_1, m_2, J_1, J_2, $r_1 = 0.5l_1$, and $r_2 = 0.5l_2$ are the mass, the moment of inertia, the half-length of link 1 and 2, and $c_1 \equiv \cos(q_1)$, $s_{12} \equiv \sin(q_1 + q_2)$, etc. The combined effects of friction and the external torque disturbance are given by

$$d_1 = 2.0 \sin(\dot{q}_1) + 2.5 \sin(\dot{q}_2) + 0.5 \sin(t),$$
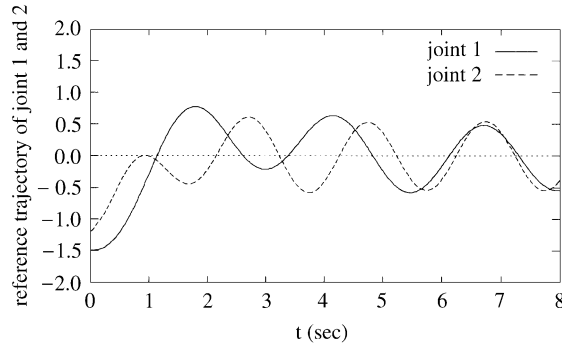$$d_2 = 5.0 \sin(\dot{q}_1) + 4.0 \sin(\dot{q}_2) + 0.4 \sin(t). \tag{25}$$

Fig. 4. Reference outputs of joints 1 and 2.

In the control experiments described below, the kinematics and inertial parameters of the arm are chosen as $l_1 = 2.04$ m, $l_2 = 1.66$ m, $J_1 = J_2 = 4.5$ kg m, $m_1 = 0.60$ kg, $m_2 = 7.02$ kg, respectively. The excessive ratio between m_1 and m_2 is to emphasize the load effect. As shown in Fig. 4, the trajectory to be followed by the ith joint is given by two decoupled linear systems as

$$\ddot{q}_{Mi} = \alpha_{i1} q_{Mi} + \alpha_{i2} \dot{q}_{Mi} + v_i, \quad i = 1, 2. \tag{26}$$

The model parameters are chosen as follows: $\alpha_{11} = -1.0$, $\alpha_{12} = -1.0$, $\alpha_{21} = -4.0$, $\alpha_{22} = -2.0$ and the driving inputs to the reference model are sinusoidal functions $v_1 = \pi \sin(0.8\pi t)$, $v_2 = 1.5\pi \cos(\pi t)$. In (17) and (19), the design parameters are given by $Q_1 = Q_2 = 10I_{2\times2}$, $\boldsymbol{R}_1 = $ Block diag $[0.01I_{256\times256}, 20\,000I_{256\times256}]$, $\boldsymbol{R}_2 = $ Block diag $[0.025I_{256\times256}, 20\,000I_{256\times256}]$, $r_{\vartheta_1} = r_{\vartheta_2} = 0.025$, $\sigma_1 = \sigma_2 = 0.002$, and $\varepsilon = 0.005$. The elements in θ_i' and ω_i are chosen randomly within the interval $(-10, 10)$ and $(-2, 2)$, respectively. The membership functions of state q_1, $\dot{q}_1$, q_2, and $\dot{q}_2$ (represented by generic variable x_k) for $M = 3^4 = 81$ regular rule partitions are defined as {NB, ZE, PB} where NB: $\mu_{A_k^j}(x_k) = \exp(-4(x_k + 1.8)^2)$, ZE: $\mu_{A_k^j}(x_k) = \exp(-4x_k^2)$, PB: $\mu_{A_k^j}(x_k) = \exp(-4(x_k - 1.8)^2)$. Meanwhile, $N = 12$, i.e. 12 rule partitions positioned near the trial trajectory are defined as $\mu_{A_k^j}(x_k) = \exp(-4(x_k - m_k^j)^2)$. The locations of these membership functions, $\boldsymbol{m}^j = [m_1^j, \ldots, m_n^j]^{\mathrm{T}}$, $j = 1, \ldots, N$, are self-organized through the LVQ network. The corresponding adaptation gain $\alpha(k)$ and the radius of N_w decreased linearly with time from 0.9 and 0.1 to zero, respectively. Two-hundred state variable samples, $\bar{\boldsymbol{x}}(T_i)$, $i = 1, \ldots, 200$ along the state trajectories were used in the training process. In the simulation, for the purpose of demonstrating the adaptability and robustness of the SOFC, preceding systematic development is applied to the robot manipulator considering environments without and with initial state error and measurement noise. For both of the two environments, three trial trajectories are run and denoted by trial A, B, C, and trial A', B', C', respectively. The results of the organization level for locating the membership functions are shown in Fig. 5. The squares denote the locations of the IF-part membership functions, $\boldsymbol{m}^j$, $j = 1, \ldots, N$. In Fig. 5, the solid lines of (b) denote the reference state trajectories, while the solid lines of (c) and (c') denote the plant state trajectories.

Fig. 5(a) shows the locations of the membership functions, $\boldsymbol{m}^j$, before applying the self-organization process and they were applied in trials A and A' for cases 1 and 2 simulations; (b) shows that the LVQ network is first trained to adjust $\boldsymbol{m}^j$ base on the state trajectory of the reference model for trial B and B'. Fig. 5(c) and (c') shows that the LVQ network is trained again for trial C and C' base on the plant state trajectory generated by trial B and B', respectively. In Fig. 5(c) and (c'), the training samples are provided with conditional weights according to the performance index J_i. The dense partition phenomenon in the larger tracking error region through learning becomes apparent when Figs. 5(b) and (c) or (c') are compared.
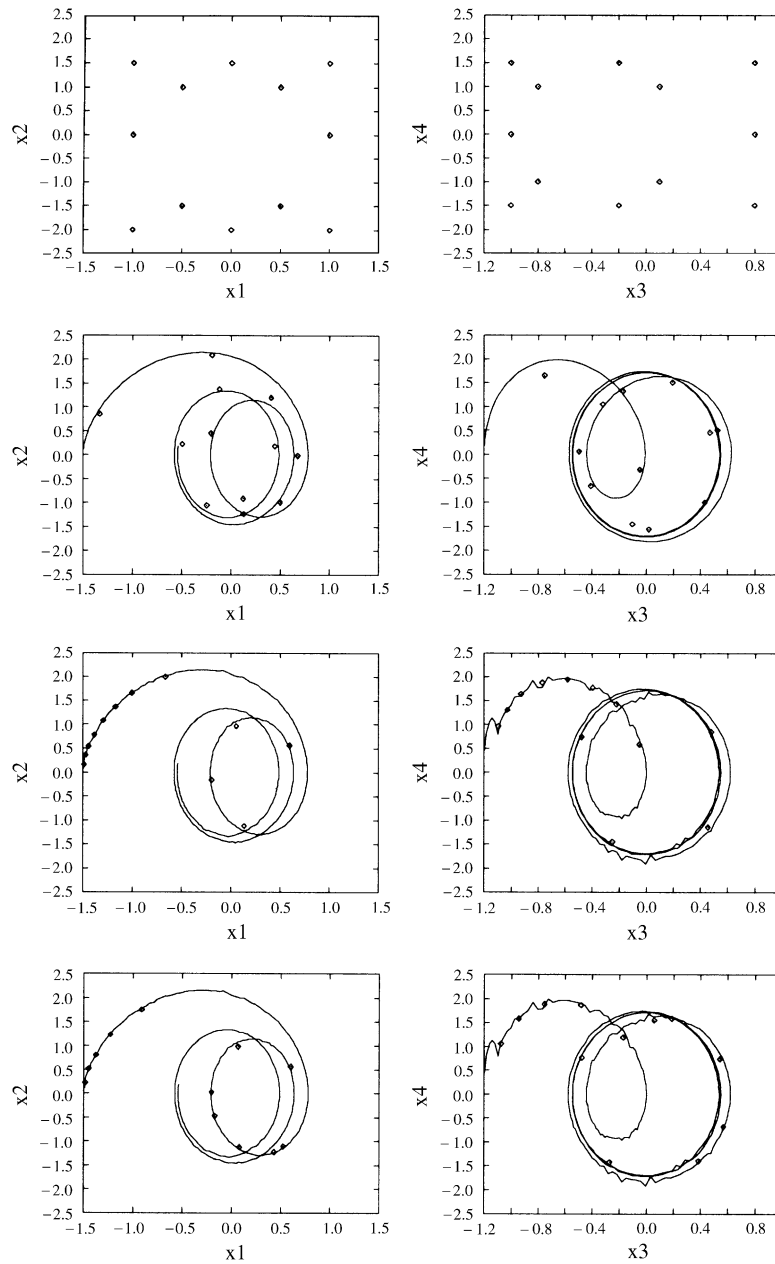
Fig. 5. Locating membership functions by LVQ network.

Case 1: In Fig. 6, the simulation result of trials *A*, *B*, and *C* which are in the situation characterized by the same initial conditions on the reference model and the plant without measurement noise are presented. The initial conditions are set to be $q_1(0) = -1.5$ rad, $q_2(0) = -1.2$ rad, $\dot{q}_1(0) = 0$ rad/s, $\dot{q}_2(0) = 0$ rad/s.

Case 2: In Fig. 7, the re-initialization error setting is satisfied within an admissible deviation level, $q_1(0) - q_{M1}(0) = -0.05$, and $q_2(0) - q_{M2}(0) = 0.05$ for each repetitive motion. The noise $\bar{q}_1 = q_1 + 0.01\cos(10t)$ and
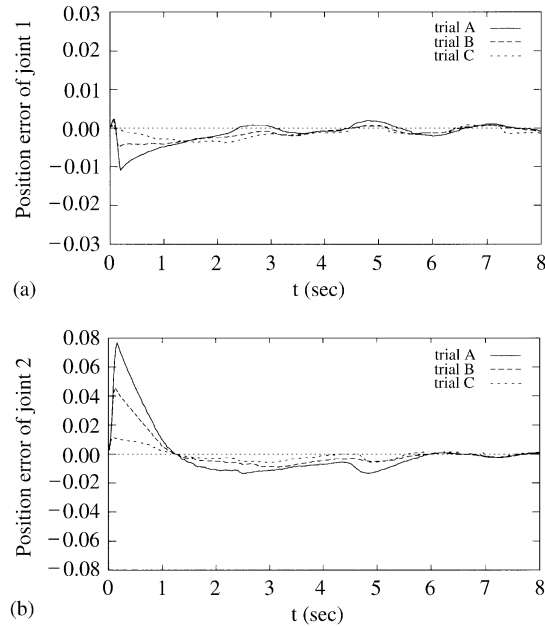
Fig. 6. Tracking error of (a) joint 1, and (b) joint 2 without initial state error and measurement noise.

$\bar{q}_2 = q_2 + 0.01\cos(10t)$ are applied. From the variation of the tracking error of the SOFC for different trials, we find that (i) For a trajectory tracking or model-reference control problem, the SOFC can be trained base on the desired trajectory or reference model (like trials B and B') to improve the tracking performance; (ii) The SOFC is particularly suitable for the control of repetitive operating processes that the state trajectories in the preceding period can be used for the training and more significant improvement of the tracking performance will be obtained (like trials C and C'). The simulation results of cases 1 and 2 also demonstrated the robustness and adaptability of the SOFC to initial state error and state measurement noise.

5. Conclusion

A hierarchical self-organizing fuzzy logic controller using LVQ network has been proposed for the robust tracking control of unknown nonlinear composite systems. A complementary algorithm for self-learning has been developed and demonstrated. Firstly, a set of regular rule partition makes it possible to avoid certain improper rule modifications. Secondly, the information about the better location of the IF-part membership function is obtained through an LVQ network. It has been proven that the overall fuzzy control system is able to guarantee the output tracking error to converge to a residual set ultimately. In addition, the system is also robust to the disturbances, measurement noise, and perturbed initialization error. The simulation results of robot control show that better results are obtained if the LVQ network is used for self-organizing the rule base. Consequently, in the context of organization level of the SOFC, by providing conditional weights according to the performance index to the training samples, the output trajectories follow the required path more closely and smoothly.

For a model-reference/-following, trajectory tracking or repetitive control problem, the information about the better location of the IF-part membership function can be obtained in the context of the organization level of the SOFC. However, the limitation of the proposed technique is mainly in operating the set-point control
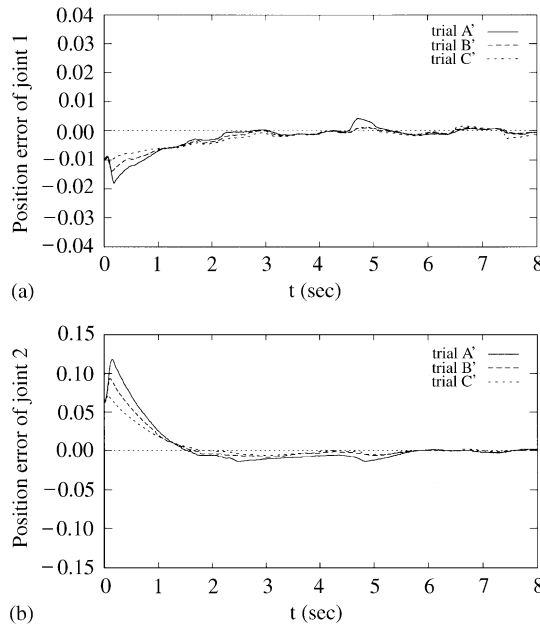
Fig. 7. Tracking error of (a) joint 1, and (b) joint 2 with admissible initial state error and measurement noise.

problem. To lift this difficulty, our further work is to design the organization level of the SOFC in which the locations of the membership functions are trained according to the moving direction of the process state instead of the position of the process state. A future work about the fuzzy systems is to determine initially the numbers of fuzzy rules to ensure a pre-specified uniform approximation capability. To adaptively modify the widths of the IF-part membership functions may be an alternative of modifying the locations of them; it is worthy to further investigate this trade-off.

Appendix A

Proof of Theorem 1. Eq. (1) can be rewritten in terms of the measured output $\bar{y}$ and the ith component as

$$
\begin{aligned}
\bar{y}_i^{(r_i)} &= f_i(\boldsymbol{x}) + g_i(\boldsymbol{x})u_i + z_i(\boldsymbol{x},t) + n_{yi}^{(r_i)} \\
&= \theta_i'^{*\mathrm{T}}\mu'(\bar{\boldsymbol{x}}) + \zeta_i^f + (\omega_i^{*\mathrm{T}}\mu(\bar{\boldsymbol{x}}) + \zeta_i^g)u_i + z_i(\boldsymbol{x},t) + n_{yi}^{(r_i)},
\end{aligned}
\tag{A.1}
$$

where

$$
\begin{aligned}
\zeta_i^f &= f_i(\boldsymbol{x}) - \theta_i'^{*\mathrm{T}}\mu'(\boldsymbol{x}) - \theta_i'^{*\mathrm{T}}\Delta\mu'(\boldsymbol{x},\boldsymbol{n}_x), \\
\zeta_i^g &= g_i(\boldsymbol{x}) - \omega_i^{*\mathrm{T}}\mu(\boldsymbol{x}) - \omega_i^{*\mathrm{T}}\Delta\mu(\boldsymbol{x},\boldsymbol{n}_x),
\end{aligned}
\tag{A.2}
$$

and

$$\Delta\mu'(\boldsymbol{x},\boldsymbol{n}_x) = \mu'(\bar{\boldsymbol{x}}) - \mu'(\boldsymbol{x}),$$

$$\Delta\mu(\boldsymbol{x},\boldsymbol{n}_x) = \mu(\bar{\boldsymbol{x}}) - \mu(\boldsymbol{x}) \qquad (A.3)$$

are measures of the sensitivity of the nominal model $\boldsymbol{z}(\boldsymbol{x},t) \equiv \boldsymbol{n}_x \equiv \boldsymbol{n}_y \equiv 0$ with respect to the measurement noise $\boldsymbol{n}_x$. By (2) and (A.1), it is then possible to derive the error equation as

$$\bar{y}_i^{(r_i)} - y_{Mi}^{(r_i)} = -\alpha_{i1}y_{Mi} - \alpha_{i2}\dot{y}_{Mi} - \cdots - \alpha_{ir_i}y_{Mi}^{(r_i-1)} - v_i$$

$$+\theta_i'^{*\mathrm{T}}\mu'(\bar{\boldsymbol{x}}) + \omega_i^{*\mathrm{T}}\mu(\bar{\boldsymbol{x}})u_i + \zeta_i, \qquad (A.4)$$

where

$$\zeta_i = \zeta_i^f + \zeta_i^g u_i + z_i(\boldsymbol{x},t) + n_{y,i}^{(r_i)}.$$

By (15), subtracting $\omega_i^{\mathrm{T}}\mu(\bar{\boldsymbol{x}})u_i$ from and adding $-\theta_i'^{\mathrm{T}}\mu'(\bar{\boldsymbol{x}}) + \alpha_{i1}\bar{y}_{Mi} + \alpha_{i2}\dot{\bar{y}}_{Mi} + \cdots + \alpha_{ir_i}\bar{y}_{Mi}^{(r_i-1)} - a_{LRi}\phi(\bar{\boldsymbol{x}})$ to the right-hand side of (A.4) obtains

$$\bar{y}_i^{(r_i)} - y_{Mi}^{(r_i)} = \alpha_{i1}(\bar{y}_i - y_{Mi}) + \alpha_{i2}(\dot{\bar{y}}_i - \dot{y}_{Mi}) + \cdots + \alpha_{ir_i}(\bar{y}_i^{(r_i-1)} - y_{Mi}^{(r_i-1)})$$

$$+(\theta_i'^{*\mathrm{T}} - \theta_i'^{\mathrm{T}})\mu'(\bar{\boldsymbol{x}}) + (\omega_i^{*\mathrm{T}} - \omega_i^{\mathrm{T}})\mu(\bar{\boldsymbol{x}})u_i + \zeta_i - a_{LRi}\phi(\bar{\boldsymbol{x}}) \qquad (A.5)$$

or

$$\dot{e}_i = A_i e_i - b_i\tilde{\theta}_i^{\mathrm{T}}\boldsymbol{w} + b_i(\zeta_i - a_{LRi}\phi), \qquad (A.6)$$

where $\tilde{\theta}_i = \theta_i - \theta_i^*$ denotes the parameter estimation error and

$$A_i = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_{i1} & -a_{i2} & -a_{i3} & \cdots & -a_{ir_i} \end{bmatrix}, \quad b_i = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}, \quad \boldsymbol{w} = \begin{bmatrix} \mu' \\ \mu \end{bmatrix}. \qquad (A.7)$$

Let V_θ and V_ϑ be positive-definite functions of the forms $V_\theta = \frac{1}{2}\sum_{i=1}^m(\theta_i^{\mathrm{T}}\theta_i)$ and $V_\vartheta = \frac{1}{2}\sum_{i=1}^m\vartheta^2$, respectively, and their time derivatives are $\dot{V}_\theta = \sum_{i=1}^m\theta_i^{\mathrm{T}}\dot{\theta}_i$ and $\dot{V}_\vartheta = \sum_{i=1}^m\vartheta_i^{\mathrm{T}}\dot{\vartheta}_i$. If the first line of (19) is true then $d_{\theta_i} = 0$, and the conclusion $\dot{V}_\theta \leqslant 0$ is trivial. If the second line of (19) is true then $d_{\theta_i} < 1$ and $\theta_i \in M_{\theta_i}^\varepsilon$. Therefore, either $\dot{V}_\theta \leqslant 0$ or $\theta_i \in M_{\theta_i}^\varepsilon$ is obtained. Similarly, one has either $\dot{V}_\vartheta < 0$ or $\vartheta_i \in M_{\vartheta_i}^\varepsilon$. Therefore, the boundedness of θ_i, ϑ_i, and $\boldsymbol{u}$ is guaranteed. To show the performance of the closed-loop system formed by (1), (15), (18), and (20), one can choose the following positive-definite function:

$$\boldsymbol{V} = V_1 + \cdots + V_m, \qquad (A.8)$$

where

$$V_i(t) = \begin{cases} \frac{1}{2}d_0^2 + \frac{1}{2}\tilde{\theta}_i^{\mathrm{T}}\boldsymbol{R}_i\tilde{\theta}_i + \frac{1}{2}\gamma_{\vartheta_i}\tilde{\vartheta}_i^2, & \text{if } \boldsymbol{e}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{b}\boldsymbol{b}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{e} \leqslant d_0^2, \\ \frac{1}{2}e_i^{\mathrm{T}}P_i e_i + \frac{1}{2}\tilde{\theta}_i^{\mathrm{T}}\boldsymbol{R}_i\tilde{\theta}_i + \frac{1}{2}\gamma_{\vartheta_i}\tilde{\vartheta}_i^2, & \text{otherwise}, \end{cases}$$

where $\tilde{\vartheta}_i = \vartheta_i - \vartheta_i^M$ is the auxiliary adjustable parameter error and $\vartheta_i^M \equiv \max\{\vartheta_i^*, \vartheta_{i0}\}$. Taking the derivative of V_i and considering (A.6), (18), and (20), one obtains: $\dot{V}_i = 0$ for $e^T P b b^T P e \leqslant d_0^2$, and

$$
\begin{aligned}
\dot{V}_i(t) &= e_i^T P_i (A_i e_i - b_i \tilde{\theta}_i^T w + b_i(\zeta_i - \alpha_{LRi}\phi)) + \tilde{\theta}_i^T (I - d_\theta \theta_{i\perp} \theta_{i\perp}^T) \\
&\quad [w b_i^T P_i e_i - \sigma_1(\theta_i - \theta_{i0})] + \tilde{\vartheta}_i (1 - d_\vartheta)[w_i' b_i^T P_i e_i - \sigma_2(\vartheta_i - \vartheta_{i0})] \\
&= \tfrac{1}{2} e_i^T (A_i^T P_i + P_i A_i) e_i - e_i^T P_i b_i \tilde{\theta}_i^T w + e_i^T P_i b_i(\zeta_i - \vartheta_i w_i') \\
&\quad + \tilde{\theta}_i^T w b_i^T P_i e_i - \sigma_1 \tilde{\theta}_i^T (\theta_i - \theta_{i0}) + \tilde{\vartheta}_i w_i' b_i^T P_i e_i - \sigma_2 \tilde{\vartheta}_i(\vartheta_i - \vartheta_{i0}) \\
&\quad - d_\vartheta \tilde{\vartheta}_i [w_i' b_i^T P_i e_i - \sigma_2(\vartheta_i - \vartheta_{i0})] - d_\theta \tilde{\theta}_i^T \theta_{i\perp} \theta_{i\perp}^T [w b_i^T P_i e_i - \sigma_1(\theta_i - \theta_{i0})]
\end{aligned}
\tag{A.9}
$$

for $e^T P b b^T P e > d_0^2$. By (19), if $\theta_{i\perp}^T [w b_i^T P_i e_i - \sigma_1(\theta_i - \theta_{i0})] \leqslant 0$, one has $d_{\theta_i} = 0$ and the last term of the above equation is equal to zero. When $\theta_{i\perp}^T [w b_i^T P_i e_i - \sigma_1(\theta_i - \theta_{i0})] > 0$, one also has $d_{\theta_i} = 0$ for $\theta_i \in M_{\theta_i}$ and the above conclusion holds. If $\theta_i \notin M_{\theta_i}$ and suppose that M_{θ_i} and M_{ϑ_i} are appropriately selected such that θ_i^* and ϑ_i^* are in the interior of M_{θ_i} and M_{ϑ_i}, respectively, one obtains

$$
\begin{aligned}
\tilde{\theta}_i^T \theta_{i\perp} &= (\theta_i - \theta_i^*)^T \theta_i / |\theta_i| \\
&= \tfrac{1}{2}[(\theta_i - \theta_i^*)^T (\theta_i - \theta_i^*) + \theta_i^T \theta_i - \theta_i^{*T} \theta_i^*] / |\theta_i| \\
&\geqslant 0,
\end{aligned}
\tag{A.10}
$$

or

$$
\tilde{\theta}_i^T \theta_{i\perp} \theta_{i\perp}^T [w b_i^T P_i e_i - \sigma_1(\theta_i - \theta_{i0})] \geqslant 0.
\tag{A.11}
$$

In a similar way, it can be shown that

$$
\tilde{\vartheta}_i [w_i' b_i^T P_i e_i - \sigma_2(\vartheta_i - \vartheta_{i0})] \geqslant 0.
\tag{A.12}
$$

Therefore,

$$
\begin{aligned}
\dot{V}_i &\leqslant \tfrac{1}{2} e_i^T (A_i^T P_i + P_i A_i) e_i + e_i^T P_i b_i(\zeta_i - \vartheta^M w_i') \\
&\quad - \sigma_1 \tilde{\theta}_i^T (\theta_i - \theta_{i0}) - \sigma_2 \tilde{\vartheta}_i(\vartheta_i - \vartheta_{i0}).
\end{aligned}
\tag{A.13}
$$

Using Assumption 1, the second term on the right-hand side satisfies the inequality

$$
\begin{aligned}
e_i^T P_i b_i(\zeta_i - \vartheta^M w_i') &\leqslant |e_i^T P_i b_i| \vartheta_i^* \phi - e_i^T P_i b_i \vartheta^M w_i' \\
&\leqslant \vartheta_i^M (|e_i^T P_i b_i| \phi - e_i^T P_i b_i w_i') \\
&= \vartheta_i^M (|e_i^T P_i b_i \phi| - e_i^T P_i b_i \phi \tanh(e_i^T P_i b_i \phi / \varepsilon)) \\
&\leqslant \vartheta_i^M \kappa\varepsilon.
\end{aligned}
\tag{A.14}
$$

Since the following fact can be shown easily by straightforward algebraic manipulation.
   Claim.

$$
0 \leqslant |\gamma| - \gamma \tanh\left(\frac{\gamma}{\varepsilon}\right) \leqslant \kappa\varepsilon
\tag{A.15}
$$

for any $\gamma \in R$, and it can be readily shown that

$$\tilde{\theta}_i^{\mathrm{T}}(\theta_i - \theta_{i0}) = \tfrac{1}{2}\tilde{\theta}_i^{\mathrm{T}}\tilde{\theta}_i + \tfrac{1}{2}(\theta_i - \theta_{i0})^{\mathrm{T}}(\theta_i - \theta_{i0}) - \tfrac{1}{2}(\theta_i^* - \theta_{i0})^{\mathrm{T}}(\theta_i^* - \theta_{i0}),$$

$$\tilde{\vartheta}_i(\vartheta_i - \vartheta_{i0}) = \tfrac{1}{2}\tilde{\vartheta}_i^2 + \tfrac{1}{2}(\vartheta_i - \vartheta_{i0})^2 - \tfrac{1}{2}(\vartheta_i^* - \vartheta_{i0})^2. \tag{A.16}$$

Therefore,

$$\dot{V}_i \leqslant -\tfrac{1}{2}e_i^{\mathrm{T}}(Q_i')e_i - \frac{\sigma_1}{2}\tilde{\theta}_i^{\mathrm{T}}\tilde{\theta}_i - \frac{\sigma_2}{2}\tilde{\vartheta}_i^2 + \frac{\sigma_1}{2}(\theta_i^* - \theta_{i0})^{\mathrm{T}}(\theta_i^* - \theta_{i0}) + \frac{\sigma_2}{2}(\vartheta_i^* - \vartheta_{i0})^2 + \vartheta_i^M \kappa\varepsilon$$

$$\leqslant -a_i V_i + \lambda_i, \tag{A.17}$$

where

$$a_i \equiv \min\left\{ \frac{\lambda_{\min}(Q_i')}{\lambda_{\max}(P_i)}, \frac{\sigma_1}{\lambda_{\max}(R_i)}, \frac{\sigma_2}{\gamma_{\vartheta_i}} \right\}$$

and

$$\lambda_i = \frac{\sigma_1}{2}(\theta_i^* - \theta_{i0})^{\mathrm{T}}(\theta_i^* - \theta_{i0}) + \frac{\sigma_2}{2}(\vartheta_i^* - \vartheta_{i0})^2 + \vartheta_i^M \kappa\varepsilon \tag{A.18}$$

or

$$\dot{V} \leqslant -aV + \lambda, \tag{A.19}$$

where $a = \min_i a_i$ and $\lambda = \sum_{i=1}^m \lambda_i$. The differential inequality (A.19) satisfies

$$0 \leqslant V(t) \leqslant \frac{\lambda}{\alpha} + \left(V(0) - \frac{\lambda}{\alpha} \right) \mathrm{e}^{-at}. \tag{A.20}$$

Therefore, e_i, θ_i, and ϑ_i are uniformly ultimately bounded. Let $\rho^* = 2\lambda/a$ then from (A.20), (24) is readily obtained.

References

[1] H.R. Berenji, P.S. Khefkar, Learning and tuning fuzzy logic controllers through reinforcements, IEEE Trans. Neural Networks 3 (1992) 725–740.

[2] M. Braae, D.A. Rutherford, Theoretical and linguistic aspects of the fuzzy logic controller, Automatica 15 (1979) 553–557.

[3] Chin-Teng Lin, C.S. George Lee, Neural-network-based fuzzy logic control and decision system, IEEE Trans. Comput. 40 (1991) 1320–1336.

[4] Chin-Teng Lin, C.S. George Lee, Reinforcement structure/parameter learning for neural-network-based fuzzy logic control systems, IEEE Trans. Fuzzy Systems 2 (1994) 46–63.

[5] D.E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading, MA, 1989, pp. 62–65.

[6] R.M. Gray, Vector quantization, IEEE ASSP Mag. 1 (1984) 4–29.

[7] J.-S.R. Jang, Self-learning fuzzy controllers based on temporal backpropagation, IEEE Trans. Neural Networks 3 (1992) 714–723.

[8] Y. Linde, A. Buzo, R.M. Gray, An algorithm for vector quantization, IEEE Trans. Comm. 28 (1980) 84–95.

[9] D.A. Linkens, H.O. Nyongesa, Learning systems in intelligent control: an appraisal of fuzzy, neural and genetic algorithm control applications, IEE Proc.: Control Theory Appl. 143 (1996) 367–386.

[10] J. Nie, D.A. Linkens, Learning control using fuzzified self-organizing radial basis function network, IEEE Trans. Fuzzy Systems 1 (1993) 280–287.

[11] J. Nie, D.A. Linkens, FCMAC: a fuzzified cerebellar model articulation controller with self-organizing capacity, Automatica 30 (1994) 655–664.

[12] T.J. Procyk, E.H. Mamdani, A linguistic self-organizing process controller, Automatica 15 (1979) 15–30.

[13] G.N. Saridis, Self-Organizing Controls of Stochastic Systems, Marcel Dekker, New York, 1977.

[14] S. Tzafestas, N. Papanikolopoulos, Incremential fuzzy expert PID control, IEEE Trans. Ind. Electron. 37 (1990) 365–371.

[15] L.X. Wang, Adaptive Fuzzy Systems and Control: Design and Stability Analysis, Prentice-Hall, Englewood Cliffs, NJ, 1994.

[16] Chih-Hsin Tsai, Chi-Hsiang Wang, Wei-Song Lin, Robust fuzzy model-following control of robot manipulators, IEEE Trans. Fuzzy Systems 8 (4) (2000) 462–469.

[17] L.A. Zadeh, Outline of a new approach to the analysis of complex systems and decision processes, IEEE Trans. Systems Man Cybernet. SMC-3 (1973) 28–44.