International Journal of Innovative Computing, Information and Control
Volume **8**, Number **2**, February **2012**

ICIC International ©2012 ISSN 1349-4198
pp. 1419–1437

OBSERVER-BASED ADAPTIVE FUZZY ROBUST CONTROLLER WITH SELF-ADJUSTED MEMBERSHIP FUNCTIONS FOR A CLASS OF UNCERTAIN MIMO NONLINEAR SYSTEMS: A PSO-SA METHOD

Kai-Shiuan Shih[1], Tzuu-Hseng S. Li[1] and Shun-Hung Tsai[2,*]

[1]aiRobots Laboratory
Department of Electrical Engineering
National Cheng Kung University
No. 1, University Rd., Tainan 70101, Taiwan
{ n2894115; thsli }@mail.ncku.edu.tw

[2]Graduate Institute of Automation Technology
National Taipei University of Technology
No. 1, Sec. 3, Chung-hsiao E. Rd., Taipei 10608, Taiwan
*Corresponding author: shtsai@ntut.edu.tw

Received November 2010; revised March 2011

Abstract. *In this paper, a novel adaptive fuzzy robust controller with a state observer approach based on the hybrid particle swarm optimization-simulated annealing (PSO-SA) technique for a class of multi-input multi-output (MIMO) nonlinear systems with disturbances is proposed. Firstly, particle swarm optimization-simulated annealing (PSO-SA) is used to adjust the fuzzy membership functions, while adaptive laws are used to approximate nonlinear functions and the unknown upper bounds of disturbances, respectively. Secondly, a state observer is applied for estimating all states which are not available for measurement in the system. By using the strictly-positive-real (SPR) stability theorem, the proposed adaptive fuzzy robust controller not only guarantees the stability of a class of MIMO nonlinear systems, but also maintains good tracking performance. Thirdly, we propose a novel auxiliary compensation, the item is designed to suppress the influence of external disturbance and remove fuzzy approximation error, respectively. The intelligence algorithm consists of the adaptive fuzzy robust method, the individual enhancement scheme and particle swarm optimization-simulated annealing structure which generates new optimal parameters for the control scheme. Finally, one simulation example is given to illustrate the effectiveness of the proposed approach.*

Keywords: MIMO, Adaptive fuzzy control, Particle swarm optimization, Robust, Simulated annealing, Strictly-positive-real (SPR) stability theorem

1. **Introduction.** Adaptive fuzzy controllers provide a methodical and powerful framework for nonlinear control problems. In [1], an adaptive fuzzy control scheme has been developed for uncertain nonlinear systems. The stability of uncertain nonlinear systems has been addressed using adaptive fuzzy control approaches [2-14]. However, a significant constraint is that the system state variables must be measured. For many real systems, the state vector is rarely fully measured. There is thus a need to implement a state observer [15-21].

For a class of SISO systems, an adaptive fuzzy controller with a state observer has been proposed in [22,23]. According to the results, many observer-based adaptive fuzzy design schemes have been proposed in [24-32], where the fuzzy (neural network) can be used to approximate any nonlinear function. For example, some observer-based indirect adaptive fuzzy control schemes have been studied in [24,30] while direct control schemes were addressed in [26,28,30]. However, in actuality, most engineering systems are multiple-input multiple-output (MIMO) systems [33-36]. MIMO systems can be found in nature, such as manipulators and chaotic systems, and are often composed of a set of subsystems [2,6,16,37-39]. Recently, adaptive fuzzy (neural network) control schemes that use universal approximators with adjustable parameters have been developed for MIMO nonlinear systems [4-7,40-44]. For example, several adaptive fuzzy control schemes were developed in [41-43] while an adaptive neural controller was designed in [44] to adjust parameters on-line.

Particle swarm optimization (PSO), proposed by Kennedy and Eberhart [45], is a new evolutionary algorithm that may be used to find optimal or near optimal solutions in a large search space. The PSO algorithms for tuning the membership function parameters of fuzzy logic controllers have been studied extensively in the literature [46-49]. However, for the basic PSO algorithm, all particles are usually trapped into the local minimum, and the optimal value found is thus often a local minimum instead of a global minimum. To overcome the shortcomings of the basic PSO algorithm, our method combines the PSO algorithm with the simulated annealing (SA) algorithm. SA is a kind of stochastic method that is well known for effectively bypassing the local minimum trap.

The control theory for nonlinear multivariable systems has universal applications. The control of multivariable systems is a difficult problem due to the coupling between the control inputs and the subsystems. Meanwhile, the states of multivariable systems are unmeasured; therefore, the control problem becomes more challenging. This paper proposes an observer-based adaptive fuzzy robust control scheme for a class of MIMO nonlinear systems with external disturbances. A state observer is introduced to estimate the unmeasured states. In the controller design, we assume that the systems are controllable and propose a novel auxiliary compensation; the control is designed to suppress the influence of external disturbance and remove the fuzzy approximation error, respectively. Compared with existing designs, the proposed method has four main contributions: (1) an adaptive fuzzy robust tracking control method for a class of MIMO systems is designed; (2) the controller does not require a priori knowledge of the sign of the control coefficient; (3) the PSO-SA algorithm is used to self-adjust the controller's coefficient for the optimal solution; and (4) a novel auxiliary compensation is introduced to eliminate external disturbance and fuzzy approximation error. All parameter adaptive laws and robust control terms are derived based on Lyapunov's stability analysis, so that convergence to zero tracking error and boundedness of all signals are guaranteed.

The rest of this paper is organized as follows: in Section 2, the system description and the design of the fuzzy logic system are presented; the adaptive fuzzy robust controller based on the PSO-SA algorithm combined with an observer for a MIMO nonlinear system is described in Section 3; simulation results are presented to validate the proposed control scheme in Section 4; and finally, the conclusions are given in Section 5.

2. **Problem Statement and Preliminaries.**

In this section, we shall describe the plant and some basic concepts of fuzzy set theory and fuzzy logic used in this paper.

2.1. **System description.**

Consider a class of MIMO nonlinear systems as represented by the following differential equations:

$$\begin{aligned} \dot{x}_{11}(t) &= x_{12}(t), \\ &\vdots \\ \dot{x}_{1(n_1-1)} &= x_{1n_1}(t), \end{aligned}$$

$$\begin{aligned}
\dot{x}_{1n_1}(t) &= f_1(\boldsymbol{x}(t)) + \sum_{j=1}^{p} g_{1j}\left(\boldsymbol{x}(t)\right) u_j(t) + \Delta d_1\left(\boldsymbol{x}(t)\right), \\
\dot{x}_{p1}(t) &= x_{p2}(t), \\
&\vdots \\
\dot{x}_{p(n_p-1)}(t) &= x_{pn_p}(t),
\end{aligned} \tag{1a}$$

$$\dot{x}_{pn_p}(t) = f_p(\boldsymbol{x}(t)) + \sum_{j=1}^{p} g_{pj}\left(\boldsymbol{x}(t)\right) u_j(t) + \Delta d_p\left(\boldsymbol{x}(t)\right), \tag{1b}$$

$$y_1(t) = x_{11}(t), \tag{1c}$$

$$\vdots$$

$$y_p(t) = x_{p1}(t). \tag{1d}$$

(1a)-(1d) can be rewritten as:

$$\begin{aligned}
&\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\left[\boldsymbol{F}\left(\boldsymbol{x}(t)\right) + \boldsymbol{G}\left(\boldsymbol{x}(t)\right)\boldsymbol{u}(t) + \Delta\boldsymbol{d}\left(\boldsymbol{x}(t)\right)\right], \\
&\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t).
\end{aligned} \tag{2}$$

Define:

$$\begin{aligned}
&\boldsymbol{F}(\boldsymbol{x}) = \left[f_1(\boldsymbol{x}), \cdots, f_p(\boldsymbol{x})\right]^T, \quad \boldsymbol{G}_i(\boldsymbol{x}) = \left[g_{1i}(\boldsymbol{x}), \cdots, g_{pi}(\boldsymbol{x})\right]^T, \quad i = 1, \ldots, p. \\
&\boldsymbol{G}(\boldsymbol{x}) = \left[\boldsymbol{G}_1(\boldsymbol{x}), \cdots, \boldsymbol{G}_p(\boldsymbol{x})\right], \\
&\boldsymbol{A}_i = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{n_i \times n_i}, \quad \boldsymbol{B}_i = \begin{bmatrix} 0 \\ 0 \\ \cdots \\ 1 \end{bmatrix}_{n_i \times 1}, \\
&\boldsymbol{C}_i = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}_{1 \times n_i}, \\
&\boldsymbol{A} = \operatorname{diag}\left[\boldsymbol{A}_1, \cdots, \boldsymbol{A}_p\right], \quad \boldsymbol{B} = \operatorname{diag}\left[\boldsymbol{B}_1, \cdots, \boldsymbol{B}_p\right], \quad \boldsymbol{C} = \operatorname{diag}\left[\boldsymbol{C}_1, \cdots, \boldsymbol{C}_p\right],
\end{aligned}$$

where $\boldsymbol{x} = \left[x_{11}, \cdots, x_{1n_1}, \cdots, x_{p1}, \cdots, x_{pn_p}\right]^T \in \boldsymbol{R}^n$ is the state vector, which is assumed to be unavailable for measurement, and $\boldsymbol{u} = [u_1, \cdots, u_p]^T \in \boldsymbol{R}^p$ and $\boldsymbol{y} = [y_1, \cdots, y_p]^T \in \boldsymbol{R}^p$ are the state vector, the control input and the output of the system, respectively, while $n = n_1 + \cdots + n_p$. $f_i\left(\boldsymbol{x}(t)\right)$ and $g_{ij}\left(\boldsymbol{x}(t)\right)$ are smooth nonlinear functions, and $\Delta\boldsymbol{d} = [\Delta\boldsymbol{d}_1, \cdots, \Delta\boldsymbol{d}_p]^T$ is the vector of external disturbance. The following assumptions are made for the controller design:

Assumption 2.1. **[50]** *The matrix $\boldsymbol{G}(\boldsymbol{x})$ as previously defined is nonsingular, i.e., $\boldsymbol{G}^{-1}(\boldsymbol{x})$ exists, and is bounded for all $\boldsymbol{x} \in U$, where $U \in \boldsymbol{R}^n$ is a compact set.*

Control objectives. To design a stabilizing controller for the system described by Equation (2) allowing the tracking error to converge to zero asymptotically. The stabilizing controller makes the system output track the given desired signal y_{pd}.

Then, the derivation of the output can be expressed as:

$$\begin{aligned}
y_i(t) &= x_{i1}(t), \quad i = 1, \cdots, p \\
&\vdots \\
y_i^{(n_i-1)}(t) &= x_{in_i}(t), \\
y_i^{(n_i)}(t) &= \dot{x}_{in_i}(t)
\end{aligned}$$

$$= f_i\left(\boldsymbol{x}(t)\right) + \sum_{j=1}^{p} g_{ij}\left(\boldsymbol{x}(t)\right) u_j(t) + \Delta d_i\left(\boldsymbol{x}(t)\right), \tag{3}$$

where $i = 1, \cdots, p$. Let $y_{1d}, \cdots, y_{pd}$ denote the desired bounded output signal that contains finite derivatives up to the nth order. The tracking errors of the system are defined as follows:

$$\begin{aligned} e_{i\ell}(t) &= y_i^{(\ell-1)}(t) - y_{id}^{(\ell-1)}(t), \quad i = 1, \cdots, p, \quad \ell = 1, \cdots, n_i, \\ &\vdots \\ \dot{e}_{i(n_i-1)}(t) &= y_i^{(n_i-1)}(t) - y_{id}^{(n_i-1)}(t), \\ \dot{e}_{in_i}(t) &= y_i^{(n_i)}(t) - y_{id}^{(n_i)}(t) \\ &= f_i(\boldsymbol{x}(t)) + \sum_{j=1}^{p} g_{ij}(\boldsymbol{x}(t)) u_j(t) - y_{id}^{(n_i)}(t) + \Delta d_i(\boldsymbol{x}). \end{aligned} \tag{4}$$

Equation (4) can be shown as:

$$\dot{\boldsymbol{e}}_i(t) = \boldsymbol{A}_i \boldsymbol{e}_i(t) + \boldsymbol{B}_i \left[f_i\left(\boldsymbol{x}(t)\right) + \sum_{j=1}^{p} g_{ij}\left(\boldsymbol{x}(t)\right) u_j(t) - y_{id}^{(n_i)}(t) + \Delta d_i\left(\boldsymbol{x}\right) \right], \quad i = 1, \cdots, p. \tag{5}$$

Equation (5) can be rewritten as:

$$\begin{aligned} \dot{\boldsymbol{e}}(t) &= \boldsymbol{A}\boldsymbol{e}(t) + \boldsymbol{B}\left[\boldsymbol{F}\left(\boldsymbol{x}(t)\right) + \boldsymbol{G}\left(\boldsymbol{x}(t)\right) \boldsymbol{u}(t) - \boldsymbol{y}_d^{(n)}(t) + \Delta \boldsymbol{D}(\boldsymbol{x}) \right], \\ \boldsymbol{e}_q(t) &= \boldsymbol{C}\boldsymbol{e}(t), \end{aligned} \tag{6}$$

where $\boldsymbol{e}_i = [e_{i1}, e_{i2}, \cdots, e_{in_i}]^T$, $\boldsymbol{e} = [\boldsymbol{e}_1, \cdots, \boldsymbol{e}_p]^T$, $\boldsymbol{e}_q = [e_{11}, e_{21}, \cdots, e_{p1}]^T$, $\Delta\boldsymbol{D} = [\Delta d_1, \cdots, \Delta d_p]^T$ and $\boldsymbol{y}_d^{(n)} = \left[y_{1d}^{(n_1)}, \cdots, y_{pd}^{(n_p)} \right]^T$.

For later derivations, the following assumption should be satisfied.

Assumption 2.2.

$$\|\Delta \boldsymbol{D}(\boldsymbol{x})\| \leq \boldsymbol{g}_d(\boldsymbol{x}), \tag{7}$$

where $\boldsymbol{g}_d(\boldsymbol{x})$ is the unknown positive smooth continuous function.

If the nonlinear functions $f_i\left(\boldsymbol{x}(t)\right)$ and $g_{ij}\left(\boldsymbol{x}(t)\right)$ are exactly known, and $\boldsymbol{G}\left(\boldsymbol{x}(t)\right) \neq 0$ based on the certainty equivalent approach, the ideal control law can be obtained as:

$$\boldsymbol{u}(t) = \boldsymbol{G}^{-1}\left(\boldsymbol{x}(t)\right) \left[-\boldsymbol{F}\left(\boldsymbol{x}(t)\right) + \boldsymbol{y}_d^{(n)} - \boldsymbol{K}_c \boldsymbol{e}(t) + \boldsymbol{u}_a(t) \right], \tag{8}$$

where $\boldsymbol{K}_c = [K_{c1}, \cdots, K_{cp}]$ is a control gain vector and $\boldsymbol{u}_a(t)$ is the flexible auxiliary compensation, which is designed for compensating the fuzzy approximation errors and the external disturbance.

2.2. **Fuzzy system description.** The basic configuration of the fuzzy logic system consists of four main components: fuzzy rule base, fuzzy inference engine, fuzzifier and defuzzifier. The fuzzy rule base consists of a collection of fuzzy IF-THEN rules:

$$\begin{aligned} R^{(l)} &: \text{If } x_1 \text{ is } F_1^l \text{ and } x_2 \text{ is } F_2^l \ldots \text{ and } x_n \text{ is } F_n^l,\ l = 1, 2, \ldots, n, \\ &\quad \text{then } y_s \text{ is } G_s^l,\ s = 1, 2, \ldots, p, \end{aligned} \tag{9}$$

where $\boldsymbol{x} = [x_1, \cdots, x_n]^T$ is the fuzzy system input; y is the fuzzy system output; and F_n^l and G_s^l are fuzzy sets. Through singleton fuzzification, center average defuzzification and product inference, the output of the fuzzy logic system can be expressed as:

$$y_s = \theta_s^T \xi(x), \quad s = 1, \cdots, p, \tag{10}$$

where $\theta_s^T = \left[\theta_s^1, \cdots, \theta_s^N\right]$, with each variable θ_s^l being the point at which the fuzzy membership function of G_s^l achieves the maximum value; and $\xi(x) = [\xi_1(x), \cdots, \xi_n(x)]^T$ with each variable ξ_l being the fuzzy basis function defined as:

$$\xi_l(x) = \prod_{i=1}^{n} \mu_{F_i^l}(x_i) \Bigg/ \sum_{l=1}^{N} \left(\prod_{i=1}^{n} \mu_{F_i^l}(x_i) \right), \tag{11}$$

where $\mu_{F_i^l}(x_i)$ is the membership function of the fuzzy set. Allow $\boldsymbol{\theta}^T = \text{diag}\left[\theta_1^T, \cdots, \theta_p^T\right]$ and $\boldsymbol{\Phi}(\boldsymbol{x}) = \text{diag}\left[\xi(\boldsymbol{x}), \cdots, \xi(\boldsymbol{x})\right]$.

MIMO fuzzy systems can be rewritten as:

$$\boldsymbol{y} = \boldsymbol{\theta}^T \boldsymbol{\Phi}(\boldsymbol{x}), \tag{12}$$

Define the nonlinear functions $f_i(\boldsymbol{x}(t))$ and $g_{ij}(\boldsymbol{x}(t))$ to be approximated by fuzzy logic systems as:

$$f_i\left(\boldsymbol{x} \left| \theta_{f_i} \right.\right) = \theta_{f_i}^T \xi(\boldsymbol{x}), \quad i = 1, 2, \ldots, p,$$

$$g_{ij}\left(\boldsymbol{x} \left| \theta_{g_{ij}} \right.\right) = \theta_{g_{ij}}^T \xi(\boldsymbol{x}), \quad i, j = 1, 2, \ldots, p,$$

$$\hat{\boldsymbol{F}}\left(\boldsymbol{x} \left| \theta_f \right.\right) = \boldsymbol{\theta}_f^T \xi_f(\boldsymbol{x}), \tag{13}$$

$$\hat{\boldsymbol{G}}\left(\boldsymbol{x} \left| \theta_g \right.\right) = \boldsymbol{\theta}_g^T \xi_g(\boldsymbol{x}). \tag{14}$$

Based on approximation theory, fuzzy inference systems with specific operations are universal approximators that can be used to model the nonlinear functions of the systems.

3. Main Results.

In this section, the adaptive fuzzy controller with the PSO-SA design method will be described.

3.1. Basic PSO algorithm.

PSO is a stochastic optimization algorithm [51-56]. The main concept of PSO is the mathematical modeling and simulation of the food searching activities of a flock of birds. The PSO algorithm includes two parts: (1) Cognitive component; and (2) Social component. First, the cognitive component models the memory of the each particle about its previous best position. Second, the social component models the memory of the each particle about the best position among the particles. From the above discussion, the mathematical model for the PSO algorithm can be written as:

$$\begin{aligned} V_i(k+1) &= V_i(k) + c_1 \cdot rand_1(\cdot) \times (Pbest_i - X_i(k)) \\ &\quad + c_2 \cdot rand_2(\cdot) \times (Gbest_n - X_i(k)), \\ X_i(k+1) &= X_i(k) + V_i(k+1), \quad i = 1, \cdots, N_{swarm}, \end{aligned} \tag{15}$$

where V_i is the velocity of particle i; X_i is the position of particle i; $Pbest_i$ and $Gbest_n$ are the best historical position of particle i and the best historical position of the neighbor of particle i with objective values F_{p_i} and F_{p_n}, respectively. $rand_1(\cdot)$ and $rand_2(\cdot)$ are random numbers between 0 and 1, while constants c_1 and c_2 are the weighting factors. Usually $Gbest_n$ is often replaced by $Gbest_g$, which is the global best position. The method has the drawback of being easily trapped in local minima. The following intelligence algorithm was thus proposed to overcome this shortcoming.

3.2. **Simulated annealing.** The simulated annealing (SA) algorithm is introduced to prevent particles from becoming trapped in a local optimum and to increase the diversity of the particle swarm. It was first put forward by Metropolis and successfully applied to optimization problems by Kirkpatrick [57].

The most significant character of SA is the probabilistic jumping property; i.e., a bad solution has a probability of being accepted as the new solution. Moreover, by adjusting the temperature, the jumping probability can be controlled. In particular, the probability decreases with decreasing temperature; and when the temperature moves toward zero, the probability approaches zero. In this scenario, better solutions are accepted. It has been proven that the SA algorithm is globally convergent with a probability of 1. In this section, we incorporate the mechanism of SA into PSO to create a hybrid optimization strategy, named PSO-SA.

As just mentioned, in PSO, $Gbest_g$ is an element of the set of all $Pbest_i$, which can be considered as a set of local optima. Therefore, we modify the selection of $Gbest_g$ to overcome premature convergence. We utilize the jumping mechanism of the SA algorithm for the selection of $Gbest_n$ instead of $Gbest_g$. When $Pbest_i$ has better features, it should have a larger probability of being selected for $Gbest_g$. By employing the mechanism of the SA algorithm, all other $Pbest_i$ can be considered as solutions worse than $Gbest_g$. Meanwhile, we can set $e^{-\left(F_{Pbest_i}-F_{Gbest_g}\right)/t}$ as the fitness function of each $Pbest_i$ to change $Gbest_g$ at a temperature t. The largest fitness value belongs to $Gbest_g$ and it's value is 1, with other fitness values within the interval $(0, 1]$. Hence, the probability of selecting $Pbest_i$ as $Gbest_g$ can be described as:

$$\left(e^{-(F_{Pbest_i}-F_{Gbest_g})/t}\right) \Bigg/ \sum_{j=1}^{N_{swarm}} e^{-(F_{Pbest_i}-F_{Gbest_g})/t}. \tag{16}$$

Then, roulette wheel selection can be used to randomly determine which $Pbest_i$ to select as $Gbest_g$.

3.3. **Optimization of fuzzy logic control with PSO-SA.** The fuzzy logic controller has 60 parameters to be adjusted. Now, the particles are chosen as:

$$P_i = \left[{}^i\sigma_{11}\ {}^ic_{11}\ \cdots\ {}^i\sigma_{15}\ {}^ic_{15}\right], \quad i = 1, 2, 3, \tag{17}$$

where parameters σ and c represent the center and deviation of the Gaussian membership functions, respectively. The whole procedure of the PSO-SA algorithm can be written as follows:

Step 1: Let $k = 0$, and randomly initialize $X_i(0)$ and $V_i(0)$, $i = 1, \cdots, N_{swarm}$. Then, evaluate objective function F_i for all particles.

Step 2: Initialize $Pbest_i$ with a copy of $X_i(0)$, $i = 1, \cdots, N_{swarm}$. Meanwhile, initialize $Gbest_g$ with a copy of the best $Pbest_i$, and define the temperature $t(0) = t_0$.

Step 3: Repeat until the stopping criterion is satisfied.
Calculate the fitness:

$$\text{Fitness}_i = \exp\left[-(F_{Pbest_i} - F_{Gbest_g})/t(k)\right], \tag{18}$$

$$\text{where } F_{P_{i,g}} = \frac{1}{N_{swarm}} \sum_{j=1}^{N_{swarm}} \sqrt{e^2(j)}. \tag{19}$$

Step 4: For every particle i, apply roulette wheel selection from the set of all $Pbest_i$, and then update V_i and X_i via the following equations:

$$\begin{aligned} V_i(k+1) &= \chi \times \{V_i(k) + c_1 \cdot rand_1(\cdot) \times (Pbest_i - X_i(k)) \\ &\quad + c_2 \cdot rand_2(\cdot) \times (Gbest_n - X_i(k))\}, \\ X_i(k+1) &= X_i(k) + V_i(k+1), \end{aligned} \tag{20}$$

where χ is a constriction coefficient that is defined as:

$$\chi = \frac{2}{\left|2 - C - \sqrt{C^2 - 4C}\right|}, \quad C = c_1 + c_2. \tag{21}$$

Step 5: Evaluate the new objective value F_i for all particles, and then update $Pbest_i$ and $Gbest_g$ (including position values and objective values).
Step 6: Set the annealing temperature to obtain $t(k+1)$, and let $k = k+1$.
Step 7: The best solution $Gbest_g$ is then obtained.

Remark 3.1. *The mechanism of the SA algorithm is combined with the PSO algorithm in this paper. The main purpose of employing the SA algorithm is to balance exploration and exploitation; the evolving process may be prolonged as a result of overcoming premature convergence.*

Remark 3.2. *The initial temperature t_0 and the method of annealing play important roles in the algorithm, so the initial value affects the performance of the PSO-SA algorithm. The initial temperature can be described as:*

$$t_0 = -F_{Gbest_g} / \ln(0.8), \tag{22}$$

For the annealing algorithm, an exponential annealing function, i.e., $k = k+1 = \lambda \cdot t(k)$, is employed, where $0 < \lambda < 1$ denotes the annealing rate.

The controller design procedure is discussed in the next section.

3.4. **Observer-based adaptive fuzzy robust controller design.** Utilizing fuzzy logic systems to approximate $f_i(\boldsymbol{x}(t))$, $g_{ij}(\boldsymbol{x}(t))$ and $\boldsymbol{g}_d(\boldsymbol{x})$:

$$\hat{\boldsymbol{F}}(\boldsymbol{x} \,|\boldsymbol{\theta}_f) = \boldsymbol{\theta}_f^T \boldsymbol{\xi}_f(\boldsymbol{x}). \tag{23}$$

$$\hat{\boldsymbol{G}}(\boldsymbol{x} \,|\boldsymbol{\theta}_g) = \boldsymbol{\theta}_g^T \boldsymbol{\xi}_g(\boldsymbol{x}). \tag{24}$$

$$\hat{\boldsymbol{g}}_d(\boldsymbol{x} \,|\boldsymbol{\theta}_{g_d}) = \boldsymbol{\theta}_{g_d}^T \boldsymbol{\xi}_{g_d}(\boldsymbol{x}). \tag{25}$$

The fuzzy control law in Equation (8) can be rewritten as:

$$\boldsymbol{u}(t) = \hat{\boldsymbol{G}}^{-1}(\boldsymbol{x}(t) \,|\boldsymbol{\theta}_g) \left[-\hat{\boldsymbol{F}}(\boldsymbol{x}(t) \,|\boldsymbol{\theta}_f) + \boldsymbol{y}_d^{(n)} - \boldsymbol{K}_c \boldsymbol{e}(t) + \boldsymbol{u}_a(t)\right], \tag{26}$$

where $\boldsymbol{K}_c = [K_{c1}, \ \cdots, \ K_{cp}]$ is the control gain vector, and

$$\boldsymbol{u}_a = -\hat{w}_c - \frac{\boldsymbol{B}_c^T \boldsymbol{P} \tilde{\boldsymbol{e}}_c}{\|\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c\|} (\hat{\boldsymbol{g}}_d(\hat{\boldsymbol{x}} \,|\boldsymbol{\theta}_{g_d})) - \frac{\boldsymbol{B}_c^T \boldsymbol{P} \tilde{\boldsymbol{e}}_c}{\|\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c\|^2} \left[\hat{\boldsymbol{e}}^T \boldsymbol{P}_2 \boldsymbol{K}_o \tilde{\boldsymbol{e}}_1\right]. \tag{27}$$

The parameter update laws are as follows:

$$\dot{\boldsymbol{\theta}}_f = \gamma_f \boldsymbol{\xi}_f(\hat{\boldsymbol{x}}) \tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c, \tag{28}$$

$$\dot{\boldsymbol{\theta}}_g = \gamma_g \boldsymbol{\xi}_g(\hat{\boldsymbol{x}}) \boldsymbol{u} \tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c, \tag{29}$$

$$\dot{\boldsymbol{\theta}}_{g_d} = \gamma_{g_d} \left\|\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c\right\| \boldsymbol{\xi}_{g_d}(\hat{\boldsymbol{x}}), \tag{30}$$

$$\dot{\hat{\boldsymbol{w}}}_c = \gamma_c \tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c, \tag{31}$$

where $\gamma_f > 0$, $\gamma_g > 0$, $\gamma_{g_d} > 0$ and $\gamma_c > 0$ are positive adaptive gains to be determined.

In order to constrain the parameters $\boldsymbol{\theta}_f$, $\boldsymbol{\theta}_g$ and $\boldsymbol{\theta}_{gd}$ within the sets M_f, M_g and M_{gd}, respectively, the following parameter projection algorithm was used. Therefore, the proposed adaptive laws (28)-(30) can be modified as follows:

$$\dot{\boldsymbol{\theta}}_f = \begin{cases} \gamma_f \boldsymbol{\xi}_f(\hat{\boldsymbol{x}})\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c & \text{if } (\|\boldsymbol{\theta}_f\| \le M_f) \\ & \text{or } \left(\|\boldsymbol{\theta}_f\| = M_f \text{ and } \tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c \boldsymbol{\theta}_1^T \boldsymbol{\xi}_f(\hat{\boldsymbol{x}}) \le 0\right) \\ P\left\{\gamma_f \boldsymbol{\xi}_f(\hat{\boldsymbol{x}})\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c\right\} & \text{if } \left(\|\boldsymbol{\theta}_f\| = M_f \text{ and } \tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c \boldsymbol{\theta}_1^T \boldsymbol{\xi}_f(\hat{\boldsymbol{x}}) > 0\right) \end{cases} \tag{32}$$

where $P\left\{\gamma_f \boldsymbol{\xi}_f(\hat{\boldsymbol{x}})\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c\right\} = \gamma_f \boldsymbol{\xi}_f(\hat{\boldsymbol{x}})\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c - \gamma_f \dfrac{\boldsymbol{\theta}_f \boldsymbol{\theta}_f^T}{\|\boldsymbol{\theta}_f\|} \boldsymbol{\xi}_f(\hat{\boldsymbol{x}})\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c$.

$$\dot{\boldsymbol{\theta}}_g = \begin{cases} \gamma_g \boldsymbol{\xi}_g(\hat{\boldsymbol{x}})\boldsymbol{u}\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c & \text{if } (\|\boldsymbol{\theta}_g\| \le M_g) \\ & \text{or } \left(\|\boldsymbol{\theta}_g\| = M_g \text{ and } \tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c \boldsymbol{\theta}_2^T \boldsymbol{\xi}_g(\hat{\boldsymbol{x}})\boldsymbol{u} \le 0\right) \\ P\left\{\gamma_g \boldsymbol{\xi}_g(\hat{\boldsymbol{x}})\boldsymbol{u}\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c\right\} & \text{if } \left(\|\boldsymbol{\theta}_g\| = M_g \text{ and } \tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c \boldsymbol{\theta}_g^T \boldsymbol{\xi}_g(\hat{\boldsymbol{x}})\boldsymbol{u} > 0\right) \end{cases} \tag{33}$$

where $P\left\{\gamma_g \boldsymbol{\xi}_g(\hat{\boldsymbol{x}})\boldsymbol{u}\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c\right\} = \gamma_g \boldsymbol{\xi}_g(\hat{\boldsymbol{x}})\boldsymbol{u}\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c - \gamma_g \dfrac{\boldsymbol{\theta}_g \boldsymbol{\theta}_g^T}{\|\boldsymbol{\theta}_g\|} \boldsymbol{\xi}_g(\hat{\boldsymbol{x}})\boldsymbol{u}\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c$.

$$\dot{\boldsymbol{\theta}}_{g_d} = \begin{cases} \gamma_{g_d} \left\|\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c\right\| \boldsymbol{\xi}_{g_d}(\hat{\boldsymbol{x}}) & \text{if } (\|\boldsymbol{\theta}_{g_d}\| \le M_{\theta g_d}) \\ & \text{or } \left(\|\boldsymbol{\theta}_{g_d}\| = M_{\theta_{g_d}} \text{ and } \left\|\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c\right\| \boldsymbol{\theta}_{g_d}^T \boldsymbol{\xi}_{g_d}(\hat{\boldsymbol{x}}) \le 0\right) \\ P\left\{\left\|\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c\right\| \gamma_{g_d} \boldsymbol{\xi}_{g_d}(\hat{\boldsymbol{x}})\right\} & \text{if } \left(\|\boldsymbol{\theta}_{g_d}\| = M_{\theta_{g_d}} \text{ and } \left\|\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c\right\| \boldsymbol{\theta}_{g_d}^T \boldsymbol{\xi}_{g_d}(\hat{\boldsymbol{x}})\boldsymbol{u} > 0\right) \end{cases} \tag{34}$$

where $P\left\{\gamma_{g_d} \boldsymbol{\xi}_{g_d}(\hat{\boldsymbol{x}})\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c\right\} = \gamma_{g_d} \boldsymbol{\xi}_{g_d}(\hat{\boldsymbol{x}}) \left\|\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c\right\| - \gamma_{g_d} \dfrac{\boldsymbol{\theta}_{g_d} \boldsymbol{\theta}_{g_d}^T}{\|\boldsymbol{\theta}_{g_d}\|} \boldsymbol{\xi}_{g_d}(\hat{\boldsymbol{x}}) \left\|\tilde{\boldsymbol{e}}_c^T \boldsymbol{P}_1 \boldsymbol{B}_c\right\|$.

Hence, the control law can be chosen as:

$$\boldsymbol{u}(t) = \hat{\boldsymbol{G}}^{-1}(\hat{\boldsymbol{x}}(t)\,|\boldsymbol{\theta}_g)\left[-\hat{\boldsymbol{F}}(\hat{\boldsymbol{x}}(t)\,|\boldsymbol{\theta}_f) + \boldsymbol{y}_d^{(n)} - \boldsymbol{K}_c\hat{\boldsymbol{e}}(t) + \boldsymbol{u}_a(t)\right]. \tag{35}$$

Using Equations (35) and (6) yields:

$$\begin{aligned} \dot{\boldsymbol{e}}(t) &= \boldsymbol{A}\boldsymbol{e}(t) - \boldsymbol{B}\boldsymbol{K}_c\hat{\boldsymbol{e}}(t) + \boldsymbol{B}\Big\{\boldsymbol{F}(\boldsymbol{x}(t) - \hat{\boldsymbol{F}}(\hat{\boldsymbol{x}}(t)\,|\boldsymbol{\theta}_f) \\ &\quad + \left[\boldsymbol{G}\left(\boldsymbol{x}(t) - \hat{\boldsymbol{G}}\left(\hat{\boldsymbol{x}}(t)\,|\boldsymbol{\theta}_g\right)\right)\right]\boldsymbol{u}(t) + \boldsymbol{u}_a(t) + \Delta\boldsymbol{D}(\boldsymbol{x})\Big\}, \\ \boldsymbol{e}_q(t) &= \boldsymbol{C}\boldsymbol{e}(t). \end{aligned} \tag{36}$$

In order to estimate the output tracking error vector, we employ the observer as follows:

$$\begin{aligned} \dot{\hat{\boldsymbol{e}}}(t) &= (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}_c)\,\hat{\boldsymbol{e}}(t) + \boldsymbol{K}_o\left(\boldsymbol{e}_q(t) - \hat{\boldsymbol{e}}_q(t)\right), \\ \hat{\boldsymbol{e}}_q(t) &= \boldsymbol{C}\hat{\boldsymbol{e}}(t). \end{aligned} \tag{37}$$

where $\boldsymbol{K}_o = [K_{o1}, \cdots, K_{op}]$ is the observer gain vector. Define the observation error vector as:

$$\tilde{\boldsymbol{e}}(t) = \boldsymbol{e}(t) - \hat{\boldsymbol{e}}(t). \tag{38}$$

By utilizing Equations (36) and (37), then Equation (38) can be written as:

$$\begin{aligned} \dot{\tilde{\boldsymbol{e}}} &= (\boldsymbol{A} - \boldsymbol{K}_0\boldsymbol{C})\,\tilde{\boldsymbol{e}} + \boldsymbol{B}\Bigg\{\left[\boldsymbol{F}(\boldsymbol{x}(t)) - \hat{\boldsymbol{F}}(\hat{\boldsymbol{x}}(t)\,|\boldsymbol{\theta}_f)\right] \\ &\quad + \left[\boldsymbol{G}(\boldsymbol{x}(t)) - \hat{\boldsymbol{G}}(\hat{\boldsymbol{x}}(t)\,|\boldsymbol{\theta}_g)\right]\boldsymbol{u}(t) + \boldsymbol{u}_a(t) + \Delta\boldsymbol{D}(\boldsymbol{x})\Bigg\}, \\ \tilde{\boldsymbol{e}}_q(t) &= \boldsymbol{C}\tilde{\boldsymbol{e}}(t). \end{aligned} \tag{39}$$

Defining $\boldsymbol{x}$, $\hat{\boldsymbol{x}}$, $\boldsymbol{\theta}_f$, $\boldsymbol{\theta}_g$ and $\boldsymbol{\theta}_{g_d}$ as belonging to the compact sets U_x, $U_{\hat{x}}$, Ω_f, Ω_g and Ω_{g_d}, respectively, can be described as:

$$\begin{cases} U_x = \{x \in R^n : \|x\| \le M_x < \infty\} \\ U_{\hat{x}} = \{\hat{x} \in R^n : \|\hat{x}\| \le M_{\hat{x}} < \infty\} \\ \Omega_f = \left\{\theta_f \in R^N : \|\theta_f\| \le M_f < \infty\right\} \\ \Omega_g = \left\{\theta_g \in R^N : \|\theta_g\| \le M_g < \infty\right\} \\ \Omega_{g_d} = \left\{\theta_{g_d} \in R^N : \|\theta_{g_d}\| \le M_{g_d} < \infty\right\} \end{cases}, \tag{40}$$

where M_x, $M_{\hat{x}}$, M_f, M_g and M_{g_d} are the parameters, and N is the number of fuzzy logic system rules. Define the optimal parameter vectors $\boldsymbol{\theta}_f^*$, $\boldsymbol{\theta}_g^*$ and $\boldsymbol{\theta}_{g_d}^*$ as:

$$\boldsymbol{\theta}_f^* = \arg \min_{\boldsymbol{\theta}_f \in \Omega_f} \left\{ \sup_{x \in U_x, \hat{x} \in U_{\hat{x}}} \left| \boldsymbol{F}(\boldsymbol{x}) - \hat{\boldsymbol{F}}(\hat{\boldsymbol{x}} \,|\boldsymbol{\theta}_f) \right| \right\}, \tag{41}$$

$$\boldsymbol{\theta}_g^* = \arg \min_{\boldsymbol{\theta}_g \in \Omega_g} \left\{ \sup_{x \in U_x, \hat{x} \in U_{\hat{x}}} \left| \boldsymbol{G}(\boldsymbol{x}) - \hat{\boldsymbol{G}}(\hat{\boldsymbol{x}} \,|\boldsymbol{\theta}_g) \right| \right\}, \tag{42}$$

$$\boldsymbol{\theta}_{g_d}^* = \arg \min_{\boldsymbol{\theta}_{g_d} \in \Omega_{g_d}} \left\{ \sup_{x \in U_x, \hat{x} \in U_{\hat{x}}} \left| \boldsymbol{g}_d(\boldsymbol{x}) - \hat{\boldsymbol{g}}_d\left(\hat{\boldsymbol{x}} \,|\boldsymbol{\theta}_{g_d}\right) \right| \right\}, \tag{43}$$

where $\tilde{\boldsymbol{\theta}}_f = \boldsymbol{\theta}_f^* - \boldsymbol{\theta}_f$, $\tilde{\boldsymbol{\theta}}_g = \boldsymbol{\theta}_g^* - \boldsymbol{\theta}_g$ and $\tilde{\boldsymbol{\theta}}_{g_d} = \boldsymbol{\theta}_{g_d}^* - \boldsymbol{\theta}_{g_d}$ are parameter estimation errors. Therefore, the minimum approximation errors can be denoted as:

$$\boldsymbol{w} = \boldsymbol{F}(\boldsymbol{x}(t)) - \hat{\boldsymbol{F}}(\hat{\boldsymbol{x}} \,|\boldsymbol{\theta}_f^*) + \left(\boldsymbol{G}(\boldsymbol{x}(t)) - \hat{\boldsymbol{G}}(\hat{\boldsymbol{x}} \left|\boldsymbol{\theta}_g^*\right.)\right)\boldsymbol{u} + \boldsymbol{g}_d(\boldsymbol{x}(t)) - \hat{\boldsymbol{g}}_d(\hat{\boldsymbol{x}} \left|\boldsymbol{\theta}_{g_d}^*\right.). \tag{44}$$

Substituting Equation (44) into (39) yields:

$$\begin{aligned} \dot{\tilde{\boldsymbol{e}}} &= (\boldsymbol{A} - \boldsymbol{K}_0\boldsymbol{C})\,\tilde{\boldsymbol{e}} + \boldsymbol{B}\Big\{\Big[\boldsymbol{F}(\boldsymbol{x}(t)) - \hat{\boldsymbol{F}}(\hat{\boldsymbol{x}}(t) \,|\boldsymbol{\theta}_f) + \hat{\boldsymbol{F}}(\hat{\boldsymbol{x}}(t) \left|\boldsymbol{\theta}_f^*\right.) - \hat{\boldsymbol{F}}(\hat{\boldsymbol{x}}(t) \left|\boldsymbol{\theta}_f^*\right.) \\ &\quad + \Big[\boldsymbol{G}(\boldsymbol{x}(t)) - \hat{\boldsymbol{G}}\left(\hat{\boldsymbol{x}}(t) \,|\boldsymbol{\theta}_g\right) + \hat{\boldsymbol{G}}(\hat{\boldsymbol{x}}(t) \left|\boldsymbol{\theta}_f^*\right.) - \hat{\boldsymbol{G}}(\hat{\boldsymbol{x}}(t) \left|\boldsymbol{\theta}_f^*\right.)\Big]\boldsymbol{u}(t) \\ &\quad + \boldsymbol{u}_a(t) + \Delta\boldsymbol{D}(\boldsymbol{x})\Big]\Big\}, \qquad\qquad (45) \\ &= (\boldsymbol{A} - \boldsymbol{K}_0\boldsymbol{C})\,\tilde{\boldsymbol{e}} + \boldsymbol{B}\Big\{\Big[\tilde{\boldsymbol{\theta}}_f^T\boldsymbol{\xi}_f(\hat{\boldsymbol{x}}) + \tilde{\boldsymbol{\theta}}_g^T\boldsymbol{\xi}_g(\hat{\boldsymbol{x}})\boldsymbol{u}(t) + \boldsymbol{F}(\boldsymbol{x}(t)) - \hat{\boldsymbol{F}}(\hat{\boldsymbol{x}}(t) \mid \boldsymbol{\theta}_f^*) \\ &\quad + \Big[\boldsymbol{G}(\boldsymbol{x}(t)) - \hat{\boldsymbol{G}}(\hat{\boldsymbol{x}}(t) \left|\boldsymbol{\theta}_f^*\right.)\Big]\boldsymbol{u}(t) + \boldsymbol{u}_a(t) + \Delta\boldsymbol{D}(\boldsymbol{x})\Big]\Big\}, \\ \tilde{\boldsymbol{e}}_q(t) &= \boldsymbol{C}\tilde{\boldsymbol{e}}(t). \qquad\qquad (46) \end{aligned}$$

Then, the output error dynamics of (46) can be described as:

$$\tilde{\boldsymbol{e}}_q(t) = \boldsymbol{H}(\boldsymbol{s})\left[\tilde{\boldsymbol{\theta}}_f^T\boldsymbol{\xi}_f(\hat{\boldsymbol{x}}) + \tilde{\boldsymbol{\theta}}_g^T\boldsymbol{\xi}_g(\hat{\boldsymbol{x}})\boldsymbol{u}(t) + \boldsymbol{w} + \boldsymbol{u}_a(t) + \Delta\boldsymbol{D}(\boldsymbol{x})\right]. \tag{47}$$

The linear transfer function $\boldsymbol{H}(\boldsymbol{s})$ is realized using standard techniques. By using the SPR Lyapunov design method, and the following equation can be obtained:

$$\begin{aligned} \tilde{\boldsymbol{e}}_q(t) = &\boldsymbol{H}(s)\boldsymbol{L}(s)\Big[\boldsymbol{L}(s)^{-1}\tilde{\boldsymbol{\theta}}_f^T\boldsymbol{\xi}_f(\hat{\boldsymbol{x}}) + \boldsymbol{L}(s)^{-1}\tilde{\boldsymbol{\theta}}_g^T\boldsymbol{\xi}_g(\hat{\boldsymbol{x}})\boldsymbol{u}(t) \\ &+\boldsymbol{L}(s)^{-1}\left(\boldsymbol{w} + \boldsymbol{u}_a(t) + \Delta\boldsymbol{D}(\boldsymbol{x})\right)\Big], \end{aligned} \tag{48}$$

where $\boldsymbol{L}(s) = [L_1(s), \cdots, L_p(s)]$ and $L_i(s) = s^{m_i} + b_{i1}s^{m_i-1} + \cdots + b_{im_i}$, $(m_i = n_i - 1)$, $i = 1, 2, \cdots, p$. After some further manipulations, Equation (48) can be rewritten as follows:

$$\begin{aligned}\dot{\tilde{\boldsymbol{e}}}_c &= (\boldsymbol{A}-\boldsymbol{K}_0\boldsymbol{C})\,\tilde{\boldsymbol{e}}_c + \boldsymbol{B}_c\left[\tilde{\boldsymbol{\theta}}_f^T\boldsymbol{\xi}_{cf}(\hat{\boldsymbol{x}}) + \tilde{\boldsymbol{\theta}}_g^T\boldsymbol{\xi}_{cg}(\hat{\boldsymbol{x}})\boldsymbol{u}(t) + \boldsymbol{w}_c + \boldsymbol{u}_{ca}(t) + \Delta\boldsymbol{D}_c(\boldsymbol{x})\right],\\ \tilde{\boldsymbol{e}}_q(t) &= \boldsymbol{C}\tilde{\boldsymbol{e}}_c(t),\end{aligned} \tag{49}$$

where $\boldsymbol{B}_c = \mathrm{diag}\,[\boldsymbol{B}_{c1},\cdots,\boldsymbol{B}_{cp}]$, $\boldsymbol{B}_{ci} = \mathrm{diag}\,[1, b_{i1},\cdots,b_{im_i}]$, $\boldsymbol{\xi}_{cf}(\hat{\boldsymbol{x}}) = \boldsymbol{L}(s)^{-1}\boldsymbol{\xi}_f(\hat{\boldsymbol{x}})$, $w_c = \boldsymbol{L}(s)^{-1}w$, $\Delta\boldsymbol{D}_c(\boldsymbol{x}) = \boldsymbol{L}(s)^{-1}\Delta\boldsymbol{D}(\boldsymbol{x})$ and $\boldsymbol{\xi}_{cg}(\hat{\boldsymbol{x}}) = \boldsymbol{L}(s)^{-1}\boldsymbol{\xi}_g(\hat{\boldsymbol{x}})$. The following lemma will be used to verify our main results.

Lemma 3.1. *[58] Assuming that X and Y are vectors or matrices with corresponding dimensions, the following equation holds:*

$$X^TX + Y^TY \geq 2X^TY. \tag{50}$$

Theorem 3.1. *Consider the MIMO nonlinear system (2) with Assumptions 2.1 and 2.2. The adaptive fuzzy robust controller is defined by Equations (26) and (27) with adaptation laws given by Equations (28)-(31). For the given positive definite matrices $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$, if there exist symmetric positive definite matrixes $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ such that the following Lyapunov equations are satisfied:*

$$(\boldsymbol{A}-\boldsymbol{K}_o\boldsymbol{C})^T\boldsymbol{P}_1 + \boldsymbol{P}_1(\boldsymbol{A}-\boldsymbol{K}_o\boldsymbol{C}) \leq -\boldsymbol{Q}_1, \tag{51a}$$

$$(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K}_c)^T\boldsymbol{P}_2 + \boldsymbol{P}_2(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K}_c) \leq -\boldsymbol{Q}_2, \tag{51b}$$

then the system (2) is asymptotically stable.

Proof: Consider the Lyapunov function as:

$$\begin{aligned}V = {}&\frac{1}{2}\tilde{\boldsymbol{e}}_c^T\boldsymbol{P}_1\tilde{\boldsymbol{e}}_c + \frac{1}{2}\hat{\boldsymbol{e}}^T\boldsymbol{P}_2\hat{\boldsymbol{e}} + \frac{1}{2\gamma_f}tr\left(\tilde{\boldsymbol{\theta}}_f^T\tilde{\boldsymbol{\theta}}_f\right) + \frac{1}{2\gamma_g}tr\left(\tilde{\boldsymbol{\theta}}_g^T\tilde{\boldsymbol{\theta}}_g\right)\\ &+\frac{1}{2\gamma_c}\left(\tilde{\boldsymbol{w}}_c^T\tilde{\boldsymbol{w}}_c\right) + \frac{1}{2\gamma_{g_d}}tr\left(\tilde{\boldsymbol{\theta}}_{g_d}^T\tilde{\boldsymbol{\theta}}_{g_d}\right).\end{aligned} \tag{52}$$

Differentiating Equation (52) with respect to time, yields:

$$\begin{aligned}\dot{V} = {}&\frac{1}{2}\dot{\tilde{\boldsymbol{e}}}_c^T\boldsymbol{P}_1\tilde{\boldsymbol{e}}_c + \frac{1}{2}\tilde{\boldsymbol{e}}_c^T\boldsymbol{P}_1\dot{\tilde{\boldsymbol{e}}}_c + \frac{1}{2}\dot{\hat{\boldsymbol{e}}}^T\boldsymbol{P}_2\hat{\boldsymbol{e}} + \frac{1}{2}\hat{\boldsymbol{e}}^T\boldsymbol{P}_2\dot{\hat{\boldsymbol{e}}} - \frac{1}{\gamma_f}tr\left(\tilde{\boldsymbol{\theta}}_f^T\dot{\hat{\boldsymbol{\theta}}}_f\right) - \frac{1}{\gamma_g}tr\left(\tilde{\boldsymbol{\theta}}_g^T\dot{\hat{\boldsymbol{\theta}}}_g\right)\\ &-\frac{1}{\gamma_{g_d}}tr\left(\tilde{\boldsymbol{\theta}}_{g_d}^T\dot{\hat{\boldsymbol{\theta}}}_{g_d}\right) - \frac{1}{\gamma_c}\left(\tilde{\boldsymbol{w}}_c^T\dot{\hat{\boldsymbol{w}}}_c\right).\end{aligned} \tag{53}$$

Then, Equation (53) can be described as:

$$\begin{aligned}\dot{V} = {}&\frac{1}{2}\tilde{\boldsymbol{e}}_c^T\left[(\boldsymbol{A}-\boldsymbol{K}_o\boldsymbol{C})^T\boldsymbol{P}_1 + \boldsymbol{P}_1(\boldsymbol{A}-\boldsymbol{K}_o\boldsymbol{C})\right]\tilde{\boldsymbol{e}}_c + \frac{1}{2}\hat{\boldsymbol{e}}^T\left[(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K}_c)^T\boldsymbol{P}_2\right.\\ &\left.+\boldsymbol{P}_2(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K}_c)\right]\hat{\boldsymbol{e}} + \tilde{\boldsymbol{e}}_c^T\boldsymbol{P}_1\boldsymbol{B}_c\left[\tilde{\boldsymbol{\theta}}_f^T\boldsymbol{\xi}_f(\hat{\boldsymbol{x}}) + \tilde{\boldsymbol{\theta}}_g^T\boldsymbol{\xi}_g(\hat{\boldsymbol{x}})\boldsymbol{u} + \boldsymbol{w}_c + \boldsymbol{u}_a\right.\\ &\left.+\Delta\boldsymbol{D}(\boldsymbol{x})\right] + \hat{\boldsymbol{e}}^T\boldsymbol{P}_2\boldsymbol{K}_o\tilde{e}_1 - \frac{1}{\gamma_f}tr\left(\tilde{\boldsymbol{\theta}}_f^T\dot{\boldsymbol{\theta}}_f\right) - \frac{1}{\gamma_g}tr\left(\tilde{\boldsymbol{\theta}}_g^T\dot{\boldsymbol{\theta}}_g\right)\\ &-\frac{1}{\gamma_c}\left(\tilde{\boldsymbol{w}}^T\dot{\hat{\boldsymbol{w}}}\right) - \frac{1}{\gamma_{g_d}}tr\left(\tilde{\boldsymbol{\theta}}_{g_d}^T\dot{\boldsymbol{\theta}}_{g_d}\right).\end{aligned} \tag{54}$$

$$\begin{aligned}\dot{V} \leq {}&\frac{1}{2}\tilde{\boldsymbol{e}}_c^T\left[(\boldsymbol{A}-\boldsymbol{K}_o\boldsymbol{C})^T\boldsymbol{P}_1 + \boldsymbol{P}_1(\boldsymbol{A}-\boldsymbol{K}_o\boldsymbol{C})\right]\tilde{\boldsymbol{e}}_c + \frac{1}{2}\hat{\boldsymbol{e}}^T\left[(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K}_c)^T\boldsymbol{P}_2\right.\\ &\left.+\boldsymbol{P}_2(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K}_c)\right]\hat{\boldsymbol{e}} + \tilde{\boldsymbol{e}}_c^T\boldsymbol{P}_1\boldsymbol{B}_c\boldsymbol{u}_a + \tilde{\boldsymbol{e}}_c^T\boldsymbol{P}_1\boldsymbol{B}_c\tilde{\boldsymbol{\theta}}_f^T\boldsymbol{\xi}_f(\hat{\boldsymbol{x}})\\ &+\tilde{\boldsymbol{e}}_c^T\boldsymbol{P}_1\boldsymbol{B}_c\tilde{\boldsymbol{\theta}}_g^T\boldsymbol{\xi}_g(\hat{\boldsymbol{x}})\boldsymbol{u} + \tilde{\boldsymbol{e}}_c^T\boldsymbol{P}_1\boldsymbol{B}_c(\tilde{\boldsymbol{w}}_c + \hat{\boldsymbol{w}}_c) + \left\|\tilde{\boldsymbol{e}}_c^T\boldsymbol{P}_1\boldsymbol{B}_c\right\|\cdot\left\|\Delta\boldsymbol{D}_c\right\|\end{aligned} \tag{55}$$

$$+\hat{\boldsymbol{e}}^T\boldsymbol{P}_2\boldsymbol{K}_o\tilde{\boldsymbol{e}}_1-\frac{1}{\gamma_f}tr\left(\tilde{\boldsymbol{\theta}}_f^T\dot{\boldsymbol{\theta}}_f\right)-\frac{1}{\gamma_g}tr\left(\tilde{\boldsymbol{\theta}}_g^T\dot{\boldsymbol{\theta}}_g\right)$$
$$-\frac{1}{\gamma_c}\left(\tilde{\boldsymbol{w}}_c^T\dot{\hat{\boldsymbol{w}}}_c\right)-\frac{1}{\gamma_{g_d}}tr\left(\tilde{\boldsymbol{\theta}}_{g_d}^T\dot{\boldsymbol{\theta}}_{g_d}\right).$$

According to Lemma 3.1 and Assumption 2.2, Equation (55) can be obtained as:

$$\begin{aligned}\dot{V}\le&\frac{1}{2}\tilde{\boldsymbol{e}}_c^T\left[(\boldsymbol{A}-\boldsymbol{K}_o\boldsymbol{C})^T\boldsymbol{P}_1+\boldsymbol{P}_1(\boldsymbol{A}-\boldsymbol{K}_o\boldsymbol{C})\right]\tilde{\boldsymbol{e}}_c+\frac{1}{2}\hat{\boldsymbol{e}}^T\left[(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K}_c)^T\boldsymbol{P}_2\right.\\&\left.+\boldsymbol{P}_2(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K}_c)\right]\hat{\boldsymbol{e}}+\tilde{\boldsymbol{e}}_c^T\boldsymbol{P}_1\boldsymbol{B}_c\boldsymbol{u}_a+\tilde{\boldsymbol{e}}_c^T\boldsymbol{P}_1\boldsymbol{B}_c\tilde{\boldsymbol{\theta}}_f^T\boldsymbol{\xi}_f(\hat{\boldsymbol{x}})+\tilde{\boldsymbol{e}}_c^T\boldsymbol{P}_1\boldsymbol{B}_c\tilde{\boldsymbol{\theta}}_g^T\boldsymbol{\xi}_g(\hat{\boldsymbol{x}})\boldsymbol{u}\\&+\tilde{\boldsymbol{e}}_c^T\boldsymbol{P}_1\boldsymbol{B}_c(\tilde{\boldsymbol{w}}_c+\hat{\boldsymbol{w}}_c)+\left\|\tilde{\boldsymbol{e}}_c^T\boldsymbol{P}_1\boldsymbol{B}_c\right\|\cdot\hat{\boldsymbol{g}}_d(\hat{\boldsymbol{x}}\,|\boldsymbol{\theta}_{g_d})+\left\|\tilde{\boldsymbol{e}}_c^T\boldsymbol{P}_1\boldsymbol{B}_c\right\|\cdot\tilde{\boldsymbol{\theta}}_{g_d}^T\boldsymbol{\xi}_{g_d}(\hat{\boldsymbol{x}})\\&+\hat{\boldsymbol{e}}^T\boldsymbol{P}_2\boldsymbol{K}_o\tilde{\boldsymbol{e}}_1-\frac{1}{\gamma_f}tr\left(\tilde{\boldsymbol{\theta}}_f^T\dot{\boldsymbol{\theta}}_f\right)-\frac{1}{\gamma_g}tr\left(\tilde{\boldsymbol{\theta}}_g^T\dot{\boldsymbol{\theta}}_g\right)\\&-\frac{1}{\gamma_c}\left(\tilde{\boldsymbol{w}}_c^T\dot{\hat{\boldsymbol{w}}}_c\right)-\frac{1}{\gamma_{g_d}}tr\left(\tilde{\boldsymbol{\theta}}_{g_d}^T\dot{\boldsymbol{\theta}}_{g_d}\right).\end{aligned}\tag{56}$$

After some straightforward manipulations, the following equation can be obtained:

$$\begin{aligned}\dot{V}\le&\frac{1}{2}\tilde{\boldsymbol{e}}_c^T\left[(\boldsymbol{A}-\boldsymbol{K}_o\boldsymbol{C})^T\boldsymbol{P}_1+\boldsymbol{P}_1(\boldsymbol{A}-\boldsymbol{K}_o\boldsymbol{C})\right]\tilde{\boldsymbol{e}}_c\\&+\frac{1}{2}\hat{\boldsymbol{e}}^T\left[(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K}_c)^T\boldsymbol{P}_2+\boldsymbol{P}_2(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K}_c)\right]\hat{\boldsymbol{e}}.\end{aligned}\tag{57}$$

Thus, $\dot{V}<0$, if Equations (51a) and (51b) are satisfied. This completes the proof of the theorem.

4. **Example.** In the following, we apply one example to verify the effectiveness of the proposed approach. The proposed controller is designed for the tracking problem of two well-known chaotic systems. In this example, the control objective is to force the system output to track the given desired trajectory $y_{pd}=\sin t$.

Consider the following interconnected Duffing system given by [59]:

$$\begin{aligned}&\dot{x}_{11}=x_{12},\\&\dot{x}_{12}=-0.1x_{12}-x_{11}^3+12\cos(t)+x_{21}+x_{22}+u_1+\Delta d_1,\\&\dot{x}_{21}=x_{22},\\&\dot{x}_{22}=-0.1x_{22}-x_{21}^3+12\cos(t)+x_{11}+x_{12}+u_2+\Delta d_2.\end{aligned}\tag{58}$$

According to the design procedure, the design proceeds as follows:

Step 1: Specify parameters for PSO-SA: $c_1=1$, $c_2=1$, $\chi=0.8$ and swarm size $N_{swarm}=100$, maximum evolution generation $Gen=200$ (stopping condition).

Step 2: Denote a positive definite $\boldsymbol{Q}_1=\text{diag}[1,1,1,1]$, $\boldsymbol{Q}_2=\text{diag}[1,1,1,1]$, $\gamma_f=\gamma_g=0.1$, $\gamma_{g_d}=0.11$, $r_c=0.01$ and $\Delta\boldsymbol{d}=\begin{bmatrix}0.5\sin(t)\\0.5\sin(t)\end{bmatrix}$. Then solving the Lyapunov Equations (51a) and (51b), we can get $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$.

Step 3: Specify the design parameters

$$K_c=\begin{bmatrix}145&45&0&0\\0&0&140&35\end{bmatrix},\quad K_o=\begin{bmatrix}45&100&0&0\\0&0&70&85\end{bmatrix}^T.$$

Step 4: The membership functions of the fuzzy sets are described as:

$$\mu_{F_i^1}(x_i)=1/1+\exp\left(5\times(x_i+2)\right),\quad\mu_{F_i^2}(x_i)=\exp\left[-(x_i+1)^2\right],$$
$$\mu_{F_i^3}(x_i)=\exp\left[-x_i^2\right],\quad\mu_{F_i^4}(x_i)=\exp\left[-(x_i-1)^2\right],$$
$$\mu_{F_i^5}(x_i)=1/1+\exp\left(5\times(x_i-2)\right).$$

Step 5: Obtain the control law and apply it to the plant. Then, compute the adaptive law to adjust the parameter.

The initial conditions are $\boldsymbol{x}(0) = [2, 2, 2, 2]^T$, $\hat{\boldsymbol{e}}(0) = [0.1, 0.1, 0.1, 0.1]^T$, $\boldsymbol{\theta}_f(0) = 0$, $\boldsymbol{\theta}_g(0) = 0.2$ and $\boldsymbol{\theta}_{g_d}(0) = 0$. Figures 1-3 show the initial membership functions for the control inputs u, e and $\dot{e}$, where e and $\dot{e}$ are mapped to the linguistic variable error and its differentiation by the fuzzification operator. In this, the two subsystems have the same initial membership functions. The fuzzy logic system inputs are composed of the five linguistic terms NB (Negative Big), NM (Negative Medium), Z (Zero), PM (Positive Medium) and PB (Positive Big). In order to compare our stabilization performance of the proposed controller with that of a conventional observer-based fuzzy adaptive robust controller (OFARC), we consider the example used in [59]. Figures 4-6 show the membership functions of u, e and $\dot{e}$ after adjustment with the PSO-SA algorithm for u_1, while Figures 7-9 show membership functions of u, e and $\dot{e}$ after adjustment with PSO-SA algorithm for u_2. Figures 10-13 show the response of the nonlinear system under the control law of OFARC and the proposed method with the initial conditions are $\boldsymbol{x}(0) = [2, 2, 2, 2]^T$, $\hat{\boldsymbol{e}}(0) = [0.1, 0.1, 0.1, 0.1]^T$, $\boldsymbol{\theta}_f(0) = 0$, $\boldsymbol{\theta}_g(0) = 0.2$ and $\boldsymbol{\theta}_{g_d}(0) = 0$. The selection of these parameters affects the convergence rate of the adaptive parameters, settling time and overshoot. It is clear that the proposed controller can stabilize the nonlinear system and show better ability of the tracking reference signals than the OFARC. From the experiment, it is shown that the convergence times of OFARC and proposed scheme are $5.5 \sim 6.0$ seconds and $2.5 \sim 3.0$ seconds, respectively. However, the fast convergence time of proposed scheme renders its performance increased about 50% over that of OFARC. The proposed control scheme can be applied to MIMO chaotic system and can achieve excellent performance for all state variables. In addition, the tracking control problems of the uncertain MIMO system have three different initial conditions $[3, 3, 3, 3]$, $[-3, -3, -3, -3]$ and $[5, -5, 5, -5]$ are considered. The simulation results are depicted in Figures 14-17, and the results show the stabilization performance of the proposed controller. One can find that the proposed scheme can achieve much better tracking performance for all state variables and is robust to external disturbance and internal uncertainty.

In order to evaluate the output control performance, we will also refer to the Integral Square Error (ISE) [60] and the Integral Absolute Error (IAE) [61] to compare with the performance of OFARC and proposed scheme, respectively.

$$J = ISE = \int_0^\infty e^2(t)dt. \tag{59}$$

$$J = IAE = \int_0^\infty |e(t)|\, dt. \tag{60}$$

Finally, the performance indices (tracking time, adjustment parameter of membership functions, exploit, IAE and ISE) of the simulation results are tabulated in Table 1.

Table 1. Performance comparison of various existing methods

Method	Time (sec)	Membership functions	Exploit	ISE	IAE
OFARC	$5.5 \sim 6.0$	Fixed	Affected by noise	2.6	6.5
Proposed method	$2.5 \sim 3.0$	Flexible	Eliminate noise	1.1	2.8

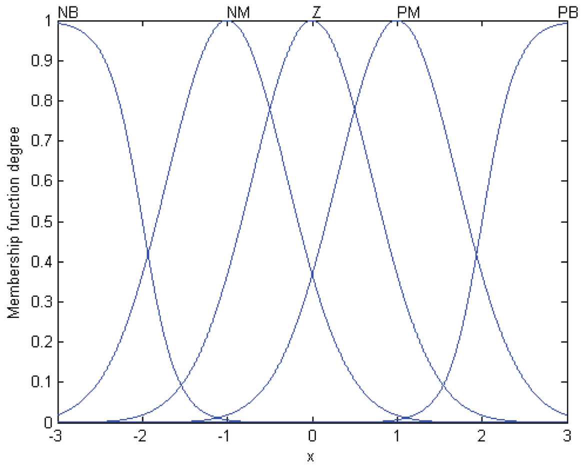

FIGURE 1. Membership functions of input variables for u_1 and u_2

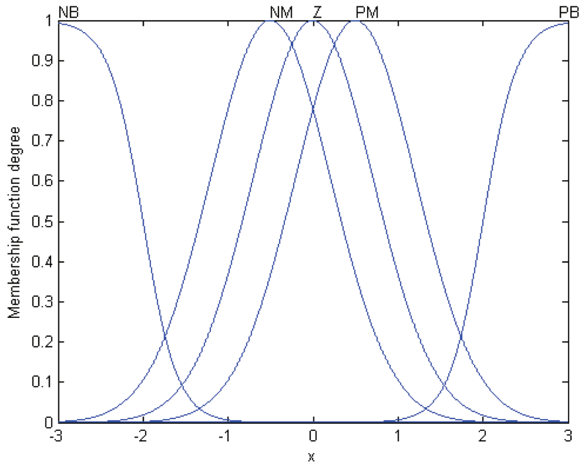

FIGURE 2. Membership functions of e for u_1 and u_2

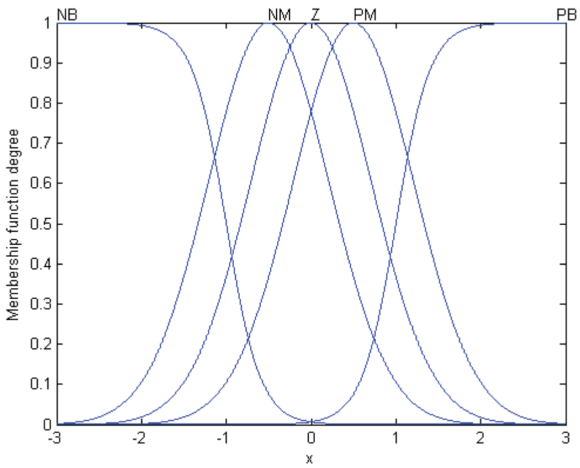

FIGURE 3. Membership functions of $\dot{e}$ for u_1 and u_2

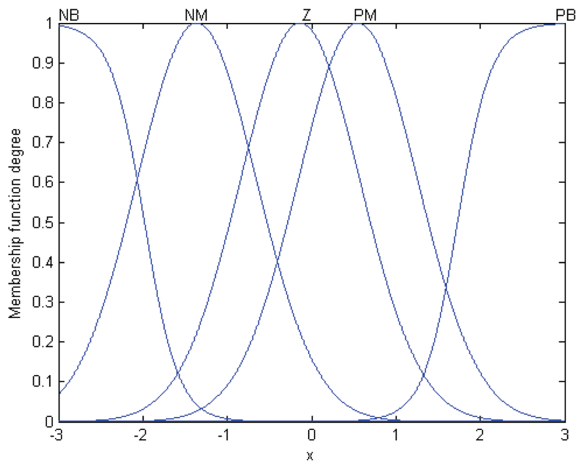

FIGURE 4. Membership functions after adjustment of input variables for u_1

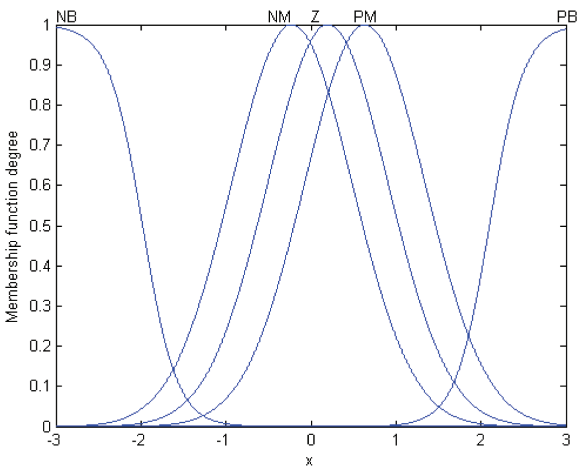

FIGURE 5. Membership functions after adjustment of input e variables for u_1

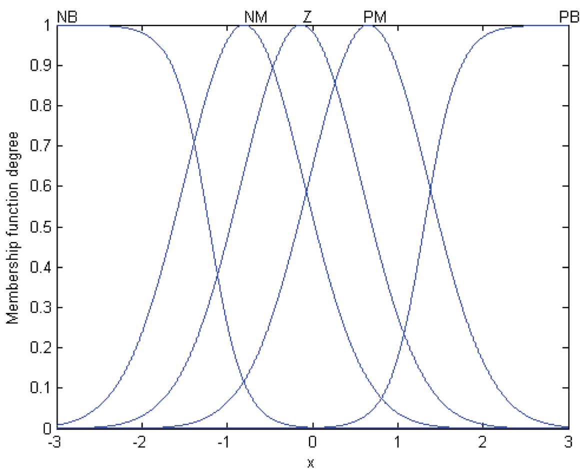

FIGURE 6. Membership functions after adjustment of input $\dot{e}$ variables for u_1

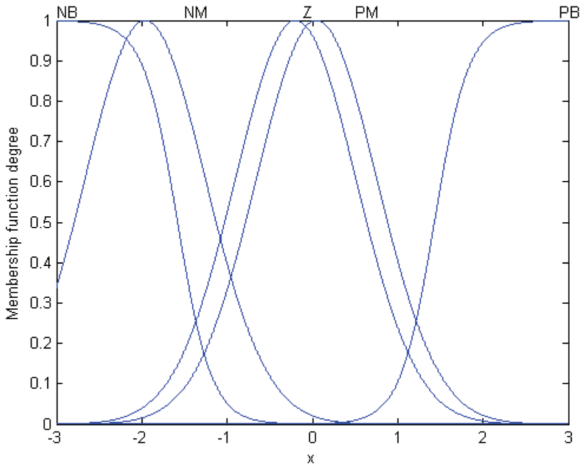

FIGURE 7. Membership functions after adjustment of input variables for u_2

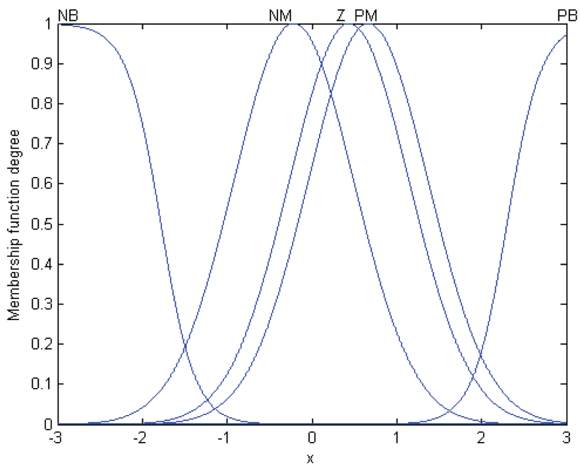

FIGURE 8. Membership functions after adjustment of e for u_2

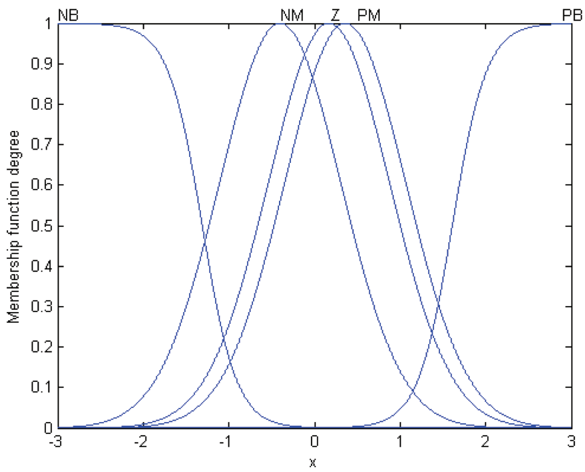

FIGURE 9. Membership functions after adjustment of $\dot{e}$ for u_2

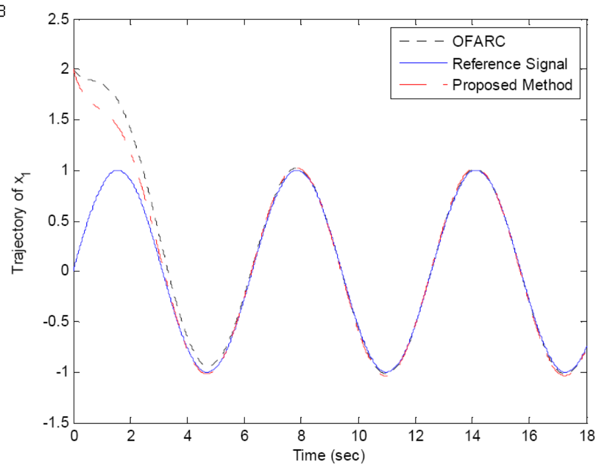

FIGURE 10. Trajectories of x_{11}

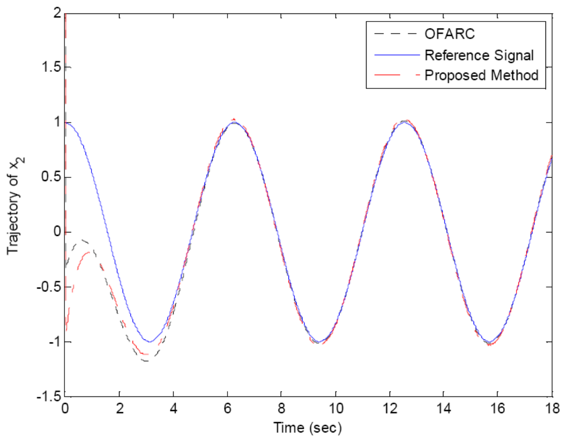

FIGURE 11. Trajectories of x_{12}

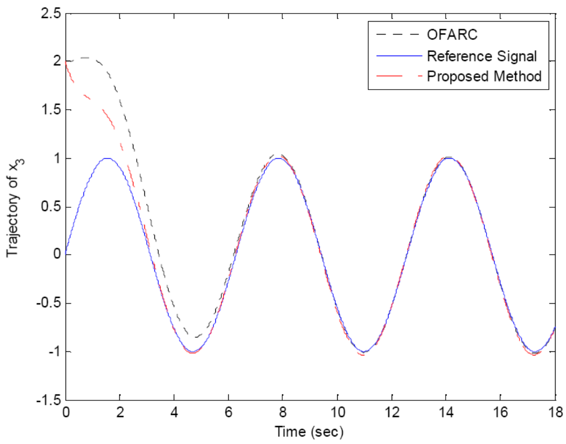

FIGURE 12. Trajectories of x_{21}

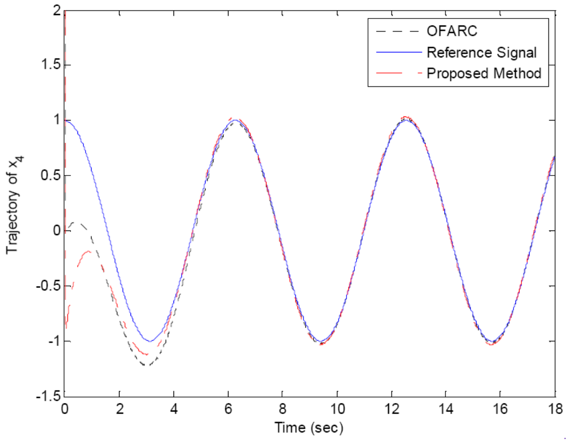

FIGURE 13. Trajectories of x_{22}

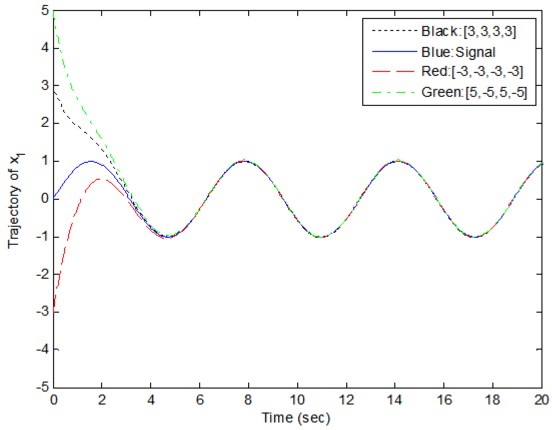

FIGURE 14. Trajectories of x_{11} (various initial conditions)

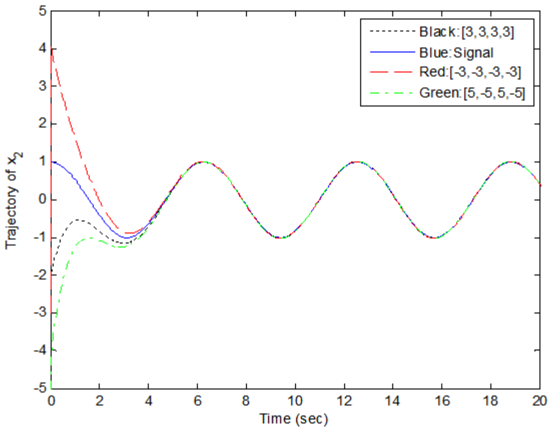

FIGURE 15. Trajectories of x_{12} (various initial conditions)

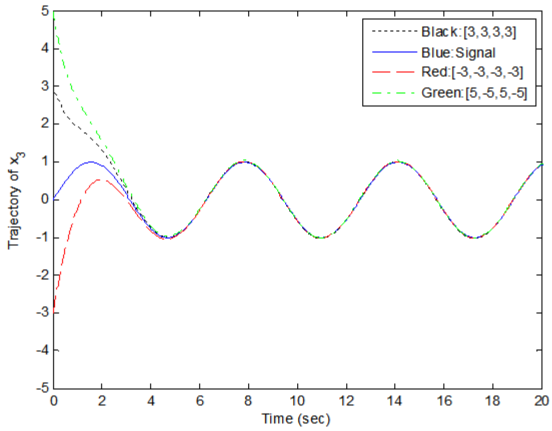

FIGURE 16. Trajectories of x_{21} (various initial conditions)

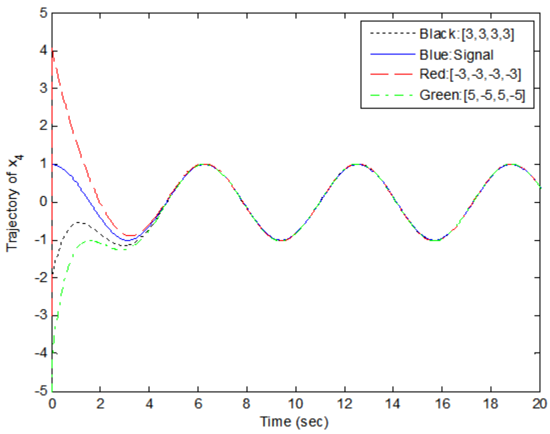

FIGURE 17. Trajectories of x_{22} (various initial conditions)

5. Conclusions and Future Work.

5.1. Conclusions.
In this paper, an observer-based adaptive fuzzy robust controller design was proposed for a class of MIMO nonlinear systems. Compared with existing designs, the four main advantages of the proposed approach are: (1) an adaptive fuzzy robust tracking control method for a class of MIMO systems is designed; (2) the controller does not require a priori knowledge of the sign of the control coefficient; (3) a PSO–SA algorithm is used to self-adjust the controller's coefficient for the optimal solution; and (4) a novel auxiliary compensation is introduced to eliminate external disturbance and fuzzy approximation error. The PSO-SA based adaptive fuzzy controller can generate fuzzy rules by transforming the universe of discourse and reducing the dependence on expert knowledge. Furthermore, the stabilization condition of the closed-loop is proposed using the Lyapunov theorem. Simulation results show that the proposed PSO-SA based adaptive fuzzy robust control scheme provides better tracking performance than an existing conventional method.

5.2. Future work.
Since the novel auxiliary compensation in this paper can eliminate external disturbance and internal uncertainty, the proposed approach can be extended to uncertain MIMO nonlinear systems with time delays. However, there are some deficiencies when solving this kind of problem in traditional PSO-SA ways. In order to deal with the issues, we will improve the PSO-SA algorithm for these existing frameworks. In addition, the proposed method can be extended the application of the PSO-SA based adaptive fuzzy controller performance in real-world problems. Furthermore, based on the proposed PSO-SA, the algorithm can be improved to deal with more complex external disturbance and internal uncertainty of nonlinear system.

Acknowledgment. This work was supported by the National Science Council of Taiwan, under Grant NSC-100-2221-E-027-017.

REFERENCES

[1] L. X. Wang, Stable adaptive fuzzy control of nonlinear systems, *IEEE Transactions on Fuzzy Systems*, vol.1, no.2, pp.146-155, 1993.
[2] A. Boulkroune, M. Tadjine, M. M'Saad and M. Farza, Fuzzy adaptive controller for MIMO nonlinear systems with known and unknown control direction, *Fuzzy Sets and Systems*, vol.161, no.6, pp.797-820, 2010.
[3] Y. C. Chang, Robust tracking control for nonlinear MIMO systems via fuzzy approaches, *Automatica*, vol.36, no.10, pp.1535-1545, 2000.
[4] B. Chen, S. C. Tong and X. P. Liu, Fuzzy approximate disturbance decoupling of MIMO nonlinear systems by backstepping approach, *Fuzzy Sets and Systems*, vol.158, no.10, pp.1097-1125, 2007.
[5] Y. J. Liu and W. Wang, Adaptive fuzzy control for a class of uncertain nonaffine nonlinear systems, *Information Sciences*, vol.177, no.18, pp.3901-3917, 2007.
[6] Y. J. Liu, S. C. Tong and W. Wang, Adaptive fuzzy output tracking control for a class of uncertain nonlinear systems, *Fuzzy Set and Systems*, vol.160, no.19, pp.2727-2754, 2009.
[7] Y. J. Liu, W. Wang, S. C. Tong and Y. S. Liu, Robust adaptive tracking control for nonlinear systems based on bounds of fuzzy approximation parameters, *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, vol.40, no.1, pp.170-184, 2010.
[8] M. Wang, B. Chen and S. L. Dai, Direct adaptive fuzzy tracking control for a class of perturbed strict-feedback nonlinear systems, *Fuzzy Sets and Systems*, vol.158, no.1, pp.2655-2670, 2007.
[9] H. G. Zhang and Z. Bien, Adaptive fuzzy control of MIMO nonlinear systems, *Fuzzy Sets and Systems*, vol.115, no.1, pp.191-204, 2000.
[10] R. Luo and B. Hong, An adaptive algorithm for localization in highly symmetric environments, *International Journal of Innovative Computing, Information and Control*, vol.1, no.2, pp.167-179, 2005.

[11] N. Iwasaki and K. Yasuda, Adaptive particle swarm optimization using velocity feedback, *International Journal of Innovative Computing, Information and Control*, vol.1, no.3, pp.369-380, 2005.

[12] H. Han, Adaptive fuzzy controller for a class of nonlinear systems, *International Journal of Innovative Computing, Information and Control*, vol.1, no.4, pp.727-742, 2005.

[13] T.-S. Shih, J.-S. Su, C.-F. Fuh and H.-M. Lee, Applying fuzzy genetic algorithms to solve a fuzzy inventory with backorder problem, *International Journal of Innovative Computing, Information and Control*, vol.6, no.1, pp.229-237, 2010.

[14] S. Bououden, S. Filali and K. Kemih, Adaptive fuzzy tracking control for unknown nonlinear systems, *International Journal of Innovative Computing, Information and Control*, vol.6, no.2, pp.541-549, 2010.

[15] S. N. Huang, K. K. Tan and T. H. Lee, Nonlinear adaptive control of interconnected systems using neural networks, *IEEE Transactions on Neural Networks*, vol.17, no.1, pp.243-246, 2006.

[16] T. P. Zhang and S. S. Ge, Adaptive neural control of MIMO nonlinear state time-varying delay systems with unknown dead-zones and gain signs, *Automatica*, vol.43, no.6, pp.1021-1033, 2007.

[17] R. Mei, Q.-X. Wu and C.-S. Jiang, Neural network robust adaptive control for a class of time delay uncertain nonlinear systems, *International Journal of Innovative Computing, Information and Control*, vol.6, no.3(A), pp.931-940, 2010.

[18] W.-Y. Wang, I.-H. Li, M.-C. Chen, S.-F. Su and Y.-G. Leu, New time-efficient structure for observer-based adaptive fuzzy-neural controllers for nonaffine nonlinear systems, *International Journal of Innovative Computing, Information and Control*, vol.6, no.3(A), pp.963-978, 2010.

[19] Z. He and Y. Lu, Decentralized robust control design for spatially interconnected systems with uncertainties, *International Journal of Innovative Computing, Information and Control*, vol.6, no.4, pp.1763-1771, 2010.

[20] C.-W. Chen and P.-C. Chen, GA-based adaptive neural network controllers for nonlinear systems, *International Journal of Innovative Computing, Information and Control*, vol.6, no.4, pp.1793-1803, 2010.

[21] X. Liu, Y. Xia, M. S. Mahmoud and Z. Deng, Modeling and stabilization of MIMO networked control systems with network constraints, *International Journal of Innovative Computing, Information and Control*, vol.6, no.10, pp.4409-4419, 2010.

[22] Y. H. Kim, F. L. Lewis and C. T. Abdallah, A dynamic recurrent neural-network-based adaptive observer for a class of nonlinear systems, *Automatica*, vol.33, no.8, pp.1539-1543, 1997.

[23] S. N. Huang, K. K. Tan and T. H. Lee, Further result on a dynamic recurrent neural-network-based adaptive observer for a class of nonlinear systems, *Automatica*, vol.41, no.12, pp.2161-2162, 2005.

[24] Y.-G. Leu, T.-T. Lee and W.-Y. Wang, Observer-based adaptive fuzzy-neural control for unknown nonlinear dynamical systems, *IEEE Transactions on Systems, and Cybernetics – Part B: Cybernetics*, vol.29, no.1, pp.583-591, 1999.

[25] S. He and F. Liu, Robust finite-time stabilization of uncertain fuzzy jump systems, *International Journal of Innovative Computing, Information and Control*, vol.6, no.9, pp.3853-3862, 2010.

[26] J. H. Park, G. T. Park, S. H. Kim and C. J. Moon, Direct adaptive self-structuring fuzzy controller for nonaffine nonlinear system, *Fuzzy Sets and Systems*, vol.153, no.1, pp.429-445, 2005.

[27] J.-D. Hwang and Z.-R. Tsai, Robust self-tuning fuzzy tracker design of time-varying nonlinear systems via conditional-LMI approach, *International Journal of Innovative Computing, Information and Control*, vol.6, no.6, pp.2621-2636, 2010.

[28] J. H. Park, G. T. Park, S. H. Kim and C. J. Moon, Output feedback control of uncertain nonlinear systems using a self-structuring adaptive fuzzy observer, *Fuzzy Sets and Systems*, vol.151, no.1, pp.21-42, 2005.

[29] C. Karakuzu, Parameter tuning of fuzzy sliding mode controller using particle swarm optimization, *International Journal of Innovative Computing, Information and Control*, vol.6, no.10, pp.4755-4770, 2010.

[30] S. C. Tong, H. X. Li and W. Wang, Observer-based adaptive fuzzy control for SISO nonlinear systems, *Fuzzy Sets and Systems*, vol.148, no.2, pp.355-376, 2004.

[31] M. Deng, N. Bu and A. Inoue, Output tracking of nonlinear feedback systems with perturbation based on robust right coprime factorization, *International Journal of Innovative Computing, Information and Control*, vol.5, no.10(B), pp.3359-3366, 2009.

[32] P. Mei, C. Cai and Y. Zou, Robust fuzzy control of nonlinear singularly perturbed systems with parametric uncertainties, *International Journal of Innovative Computing, Information and Control*, vol.4, no.8, pp.2079-2086, 2008.

[33] N. Golea, A. Golea, K. Barra and T. Bouktir, Observer-based adaptive control of robot manipulators: fuzzy systems approach, *Applied Soft Computing*, vol.8, no.1, pp.778-787, 2008.
[34] S. C. Tong, C. Y. Li and Y. M. Li, Fuzzy adaptive observer backstepping control for MIMO nonlinear systems, *Fuzzy Sets and Systems*, vol.160, no.1, pp.2755-2775, 2009.
[35] R. Lozano and B. Brogliato, Adaptive control of a simple nonlinear system without a priori information on the plant parameters, *IEEE Transactions on Automatic Control*, vol.37, no.1, pp.30-37, 1992.
[36] S. S. Ge and J. Wang, Robust adaptive tracking for time-varying uncertain nonlinear systems with unknown control coefficients, *IEEE Transactions on Automatic Control*, vol.48, no.8, pp.1463-1469, 2003.
[37] J. Sun, K. Fujimoto, C. Dou, J. Zhong and S. Yuan, H-infinity fuzzy tracking control for multimachine interconnected power system with model uncertainties, *International Journal of Innovative Computing, Information and Control*, vol.2, no.1, pp.61-68, 2006.
[38] W.-J. Chang, C.-C. Ku, W. Chang and P.-H. Huang, Passive fuzzy control with relaxed conditions for discrete affine T-S fuzzy systems, *International Journal of Innovative Computing, Information and Control*, vol.3, no.4, pp.853-871, 2007.
[39] S. Tong and Y. Li, Direct adaptive fuzzy backstepping control for a class of nonlinear systems, *International Journal of Innovative Computing, Information and Control*, vol.3, no.4, pp.887-896, 2007.
[40] Q. Kang, W. Wang and Y. Liu, Robust adaptive fuzzy control for a class of nonlinear interconnected systems, *International Journal of Innovative Computing, Information and Control*, vol.5, no.2, pp.323-335, 2009.
[41] D. V. Diaz and Y. Tang, Adaptive robust fuzzy control of nonlinear systems, *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics*, vol.34, no.3, pp.1596-1601, 2004.
[42] M. Wang, B. Chen, X. P. Liu and P. Shi, Adaptive fuzzy tracking control for a class of perturbed strict-feedback nonlinear time-delay systems, *Fuzzy Sets and Systems*, vol.159, no.1, pp.949-967, 2008.
[43] H. Xu, Y. Zou and S. Xu, Robust H-infinity control for a class of uncertain nonlinear two-dimensional systems, *International Journal of Innovative Computing, Information and Control*, vol.1, no.2, pp.181-191, 2005.
[44] M. Wang, B. Chen and P. Shi, Adaptive neural control for a class of perturbed strict-feedback nonlinear time-delay systems, *IEEE Transactions on Systems, Man and Cybernetics – Part B: Cybernetics*, vol.38, no.3, pp.721-730, 2008.
[45] J. Kennedy and R. Eberhart, Particle swarm optimization, *Proc. of IEEE Internat. Conf. on Neural Networks*, vol.4, pp.1942-1948, 1995.
[46] F. Herrera, M. Lozano and J. L. Verdegay, A learning process for fuzzy control rules using genetic algorithms, *Fuzzy Sets and Systems*, vol.100, no.1-3, pp.143-158, 1998.
[47] A. Chatterjee and K. Watanabe, An optimized Takagi-Sugeno type neuro-fuzzy system for modeling robot manipulators, *Neural Computing & Applications*, vol.15, no.1, pp.55-61, 2006.
[48] C. F. Juang and C. Lo, Fuzzy systems design by clustering-aided ant colony optimization for plant control, *International Journal of General Systems*, vol.36, pp.623-641, 2007.
[49] C. C. Wong, H. Y. Wang and S. A. Li, PSO-based motion fuzzy controller design for mobile robots, *International Journal of Fuzzy Systems*, vol.10, no.1, pp.24-32, 2008.
[50] L.-X. Wang, *Adaptive Fuzzy Systems and Control: Design and Stability Analysis*, Prentice Hall, NJ, USA, 1994.
[51] S. Naka, T. Genji, T. Yura and Y. Fukuyama, A hybrid particle swarm optimization for distribution state estimation, *IEEE Trans. Power Systems*, vol.18, no.1, pp.60-68, 2003.
[52] I. C. Trelea, The particle swarm optimization algorithm: Convergence analysis and parameter selection, *Information Processing Letters*, vol.85, no.6, pp.317-325, 2003.
[53] J. Olamaei, T. Niknam and G. Gharehpetian, Application of particle swarm optimization for distribution feeder reconfiguration considering distributed generators, *Applied Mathematics and Computation of Journal*, vol.200, no.1, pp.575-586, 2008.
[54] H. Izakian, B. T. Ladani, A. Abraham and V. Snasel, A discrete particle swarm optimization approach for grid job scheduling, *International Journal of Innovative Computing, Information and Control*, vol.6, no.9, pp.4219-4233, 2010.
[55] R.-J. Kuo and F.-J. Lin, Application of particle swarm optimization-based clustering method to reduce SMT setup time for industrial PC manufacturer in Taiwan, *International Journal of Innovative Computing, Information and Control*, vol.6, no.8, pp.3381-3400, 2010.

[56] M. Rashid and A. R. Baig, PSOGP: A genetic programming based adaptable evolutionary hybrid particle swarm optimization, *International Journal of Innovative Computing, Information and Control*, vol.6, no.1, pp.287-296, 2010.
[57] S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi, Optimization by simulated annealing, *Science*, vol.220, pp.671-680, 1983.
[58] P. A. Ioannou and J. Sun, *Robust Adaptive Control*, Prentice Hall, NJ, USA, 1996.
[59] D. Lin and X. Wang, Observer-based decentralized fuzzy neural sliding mode control for interconnected unknown chaotic systems via network structure adaptation, *Fuzzy Sets and Systems*, vol.161, no.1, pp.2066-2080, 2010.
[60] C. A. Smith and A. B. Corripio, *Principles and Practice of Automatic Process Control*, Wiley, NY, USA, 1985.
[61] F. G. Shinskey, *Process Control System: Application, Design and Tuning*, 3rd Edition, McGraw-Hill, NY, USA, 1988.