Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2008, Article ID 325859, 16 pages
doi:10.1155/2008/325859

Research Article

GA-Based Fuzzy Sliding Mode Controller for Nonlinear Systems

P. C. Chen,[1] C. W. Chen,[2] and W. L. Chiang[1]

[1] *Department of Civil Engineering, National Central University, Chung-li 32011, Taiwan*
[2] *Department of Logistics Management, College of Management, Shu-Te University, Kaohsiung 82445, Taiwan*

Correspondence should be addressed to C. W. Chen, cwchen@mail.stu.edu.tw

Received 20 February 2008; Revised 4 June 2008; Accepted 8 August 2008

Recommended by Paulo Gonçalves

Generally, the greatest difficulty encountered when designing a fuzzy sliding mode controller (FSMC) or an adaptive fuzzy sliding mode controller (AFSMC) capable of rapidly and efficiently controlling complex and nonlinear systems is how to select the most appropriate initial values for the parameter vector. In this paper, we describe a method of stability analysis for a GA-based reference adaptive fuzzy sliding model controller capable of handling these types of problems for a nonlinear system. First, we approximate and describe an uncertain and nonlinear plant for the tracking of a reference trajectory via a fuzzy model incorporating fuzzy logic control rules. Next, the initial values of the consequent parameter vector are decided via a genetic algorithm. After this, an adaptive fuzzy sliding model controller, designed to simultaneously stabilize and control the system, is derived. The stability of the nonlinear system is ensured by the derivation of the stability criterion based upon *Lyapunov*'s direct method. Finally, an example, a numerical simulation, is provided to demonstrate the control methodology.

1. Introduction

Over the past few years, fuzzy control (FC) can be designed without needing an exact mathematical model of the system to be controlled, and can efficiently control complex continuous unmodeled or partially modeled processes [1, 2]. There have been significant research efforts devoted to the analysis and control designs for fuzzy systems (see [3, 4] and the references therein). The main motivation for this development has been applied to practical nonlinear systems and engineering problems (see [5–7] and the references therein). Undoubtedly, Lyapunov's theory is one of the most common approaches for dealing with the stability analysis of systems. However, to overcome the conservatism that arises from the use of Lyapunov's methods, it has been necessary to develop a number of more effective methods, for example, fuzzy Lyapunov functions [8, 9]. There are also many important issues that have advanced results for T-S fuzzy control systems, such as time delays [10–13], H^{∞} performance [3–15], robustness [16, 17], neural networks (NNs), and genetic algorithms (GAs) [18–21]. Furthermore, much work has been published on the design of fuzzy sliding mode controllers (FSMCs) [22, 23]. An FSMC is composed of an FC and a sliding mode controller (SMC) [24–26]. An FSMC is a powerful and robust control strategy for the treatment of modeling uncertainties and external disturbances. Although control performance is good, one still has to decide on the parameters. This is one of the most important issues in their design.

In the so-called *adaptive FSMC* (AFSMC), [27–29], an adaptive algorithm is utilized to find the best high-performance parameters for the FSMC [30, 31]. In recent years, adaptive fuzzy control system designs have attracted a good deal of attention as a promising way to approach nonlinear control problems [30, 31]. For adaptive fuzzy control, one initially constructs a fuzzy model to describe the dynamic characteristics of the controlled system; then, an FSMC is designed based on the fuzzy model to achieve the control objectives. After this, adaptive laws are designed (with Lyapunov's synthesis approach) for tuning the adjustable parameters of the fuzzy models, and analyzing the stability of the overall system.

Deciding on the fuzzy rules and the initial parameter vector values for the AFSMC is very important. A genetic algorithm [32–34] is usually used as an optimization technique in the self-learning or training strategy for deciding on the fuzzy control rules and the initial values of the parameter vector. This GA-based AFSMC should improve the immediate response, the stability, and the robustness of the control system.

Another common problem encountered when switching the control input of the FSMC system is the so-called "chattering" phenomenon. Chattering is eliminated by smoothing the control discontinuity inside a thin boundary layer, which essentially acts as a low-pass filter structure for the local dynamics [25]. The boundary-layer function is introduced into these updated laws to cover parameter and modeling errors, and to guarantee that the state errors converge within a specified error bound.

In this study, we focus on the design of robust tracking control for a class of nonlinear uncertain system involving plant uncertainties and external disturbances. First, the nonlinear system for the tracking of a reference trajectory for the plant [35] is described via fuzzy models with fuzzy rules. A genetic algorithm is used to find the initial values of the parameter vector. Then the designed adaptive control laws of the reference adaptive fuzzy sliding mode controller (RAFSMC) are updated. This GA-based RAFSMC would improve the immediate response, the stability, and the robustness of the control system. Finally, both the tracking error and the modeling error approach zero.

2. Reference modeling of a nonlinear dynamic system

The plant is a single-input/single-out nth-order system with $n \geq 1$:

$$\begin{gathered}
\dot{x}_1 = x_2, \\
\vdots \\
\dot{x}_{n-1} = x_n, \\
\dot{x}_n = f(x) + g(x) \cdot u + d, \\
y = x_1,
\end{gathered} \tag{2.1}$$

where $x = [x_1, x_2, \ldots, x_{n-1}, x_n]^T \in R^n$ is the state vector of the system; $u \in R$ is the control signal; f, g are smooth nonlinear functions; d denotes the external disturbance $d(t)$ which is unknown but usually bounded.

The states $x = [x_1, x_2, \ldots, x_{n-1}, x_n]^T$ are assumed to be available. For example, a single robot can be represented in the form of (2.1), with $n = 2$ and $x(x_1 = \theta, x_2 = \dot{\theta})$ being measurable. Differentiating the output with respect to time for n times (till the control input u appears), one obtains the input/output form of (2.1):

$$\overset{(n)}{y} = f(x) + g(x) \cdot u + d(t). \tag{2.2}$$

The system is said to have a relative degree n, if $g(x)$ is bounded away from zero.

Assumption 2.1. g(x) is bounded away from zero over a compact set $\zeta \subset R^n$,

$$\left|g(x)\right| \geq b > 0, \quad \forall x \in \zeta. \tag{2.3}$$

If the control goal is for the plant output y to track a reference trajectory y_r, the reference control input r can be defined by the following reference model:

$$r = \overset{(n)}{y_r} + \alpha_{n-1} \overset{(n-1)}{y_r} + \alpha_{n-2} \overset{(n-2)}{y_r} + \cdots + \alpha_1 \dot{y}_r + \alpha_0 y_r, \tag{2.4}$$

where $\alpha_{n-1}, \alpha_{n-2}, \ldots, \alpha_1, \alpha_0$ are chosen such that the polynomial $\ell^n + \alpha_{n-1}\ell^{n-1} + \alpha_{n-2}\ell^{n-2} + \cdots + \alpha_1 \ell + \alpha_0$ is Hurwitz, and ℓ here denotes the complex *Laplace* variable.

If $f(x)$, $g(x)$ are known, and assumption 2.1 is satisfied, the control law can defined by

$$u = \frac{-f(x) - d(x) - \left(\alpha_{n-1} \overset{(n-1)}{y} + \cdots + \alpha_1 \dot{y} + \alpha_0 y\right) + r}{g(x)}, \quad \forall x \in S. \tag{2.5}$$

Substituting (2.5) into (2.1), the linearized system becomes

$$\left(\overset{(n)}{y_r} - \overset{(n)}{y}\right) + \alpha_{n-1}\left(\overset{(n-1)}{y_r} - \overset{(n-1)}{y}\right) + \cdots + \alpha_1 (\dot{y}_r - \dot{y}) + \alpha_0 (y_r - y) = 0. \tag{2.6}$$

If we define $e = y_r - y$ as the tracking error, then the reference control input (2.4) results in the following error equation:

$$\overset{(n)}{e} + \alpha_{n-1} \overset{(n-1)}{e} + \cdots + \alpha_1 \dot{e} + \alpha_0 e = 0. \tag{2.7}$$

It is clear that e will approach zero if $\alpha_{n-1}, \alpha_{n-2}, \ldots, \alpha_1, \alpha_0$ are chosen, such that the polynomial $\ell^n + \alpha_{n-1}\ell^{n-1} + \alpha_{n-2}\ell^{n-2} + \cdots + \alpha_1 \ell + \alpha_0$ is Hurwitz.

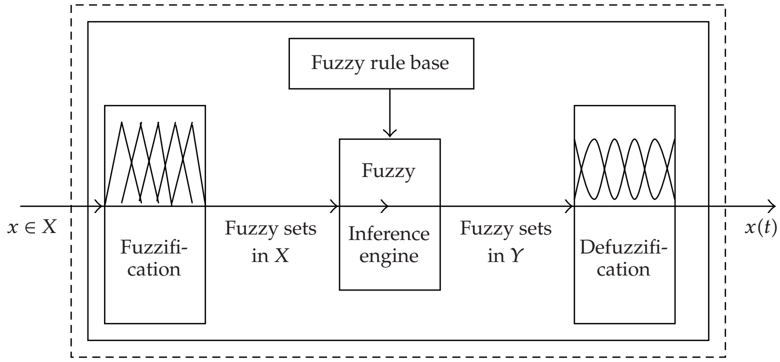

Figure 1: The fuzzy logic controller system.

3. Development of a GA-based FSMC

In general, people describe the decision-making process using linguistic statements, such as "IF something happens, THEN do a certain action." For example, let us look at a rule: "IF the temperature is high, THEN the power of the heater is low." In this statement both "high" and "low" are linguistic terms. Although this kind of linguistic rule is not precise, humans can use them to make correct decisions. To utilize such fuzzy information in a scientific way, mathematical representation of the fuzzy information is needed. Fuzzy set theory and approximate reasoning are two ways that such linguistic information can be dealt with mathematically. A review of the literature provides the theoretical foundation for the developed fuzzy logic controller. The configuration of the fuzzy logic controller is shown in Figure 1.

The basic concepts for fuzzy sets and fuzzy logic are briefly described below.

(1) *Fuzzy set, fuzzifier, and membership function*. Let X denote the universe of discourse. A fuzzy set A in X is characterized by a membership function $\mu_A : X \to [0,1]$, with $\mu_A(x)$ representing the grade of membership of $x \in X$ in fuzzy set A. For example, the Gaussian-shaped membership function is represented as $\mu_A(x) = \exp(-((x-m)/\sigma)^2)$, where m is the center and σ denotes the spread of the membership function.

(2) *Fuzzy rule base and fuzzy inference engine*. Each rule R_j in the fuzzy rule base can be expressed as

$$R_j : \text{IF } x_1 \text{ is } A_{1j} \text{ and } \cdots \ x_n \text{ is } A_{nj}\text{, THEN } y \text{ is } B_j\text{; and } \ \mu_{R_j}(\chi) = \bigcap_{i=1}^{n} \mu_{A_{ij}}(x_i). \tag{3.1}$$

(3) *Deffuzzifier*. The defuzzifier maps a fuzzy set A in X to a crisp point $x \in X$. There are several defuzzification methods described in the literature. The most popular is the weighted average defuzzification method defined as $y = \sum_{j=1}^{N} \theta_j \cdot \mu_{R_j}(\chi) / \sum_{j=1}^{N} \mu_{R_j}(\chi)$.

The FSMC is composed of a sliding mode controller and an FLC. This makes it a powerful and robust control strategy for the treatment of modeling uncertainties and external disturbances. The sliding mode plant combined with the FLC is shown in Figure 2.

Genetic algorithms (GAs) are parallel, global search techniques derived from the concepts of evolutionary theory and natural genetics. They emulate biological evolution by means of genetic operations such as reproduction, crossover, and mutation. GAs are usually

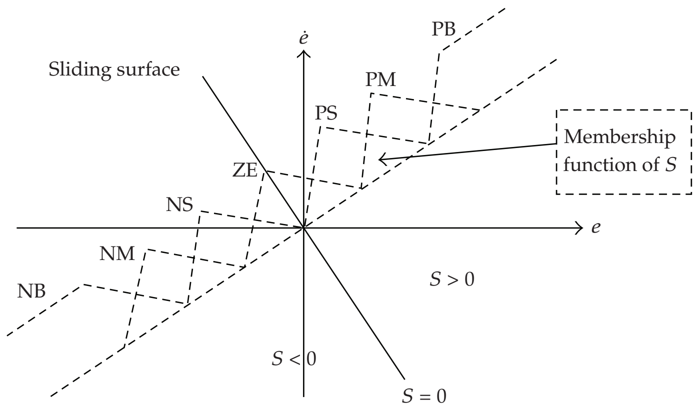

Figure 2: The sliding mode plant combined with the FLC.

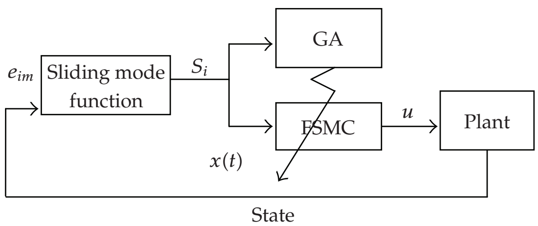

Figure 3: GA-based FSMC.

used as optimization techniques and it has been shown that they also perform well with multimodal functions (i.e., functions which have multiple local optima).

Genetic algorithms work with a set of artificial elements (binary strings, e.g., 0101010101) called a population. An individual (string) is referred to as a chromosome, and a single bit in the string is called a gene. A new population (called offspring) is generated by the application of genetic operators to the chromosomes in the old population (called parents). Each iteration of the genetic operation is referred to as a generation.

A fitness function, specifically the function to be maximized, is used to evaluate the fitness of an individual. The offspring may have better fitness than their parents. Consequently, the value of the fitness function increases from generation to generation. In most genetic algorithms, mutation is a random-work mechanism to avoid the problem of being trapped in a local optimum. Theoretically, a global optimal solution can be found.

Offspring are generated from the parents until the size of the new population is equal to that of the old population. This evolutionary procedure continues until the fitness reaches the desired specifications. However, in a specific application, the fitness specification might be used to stop the evolutionary process. In most applications, the optimal fitness value is totally unknown. In this case, the evolutionary process is interrupted either by stabilization of the fitness value (the variation is below a specific value) or by reaching the maximum number of generations.

Knowledge acquisition is the most important task in the fuzzy sliding mode controller design. The initial values of the entries in the consequent parameter vector are decided by the self-organizing of FSMC system which developed based on GA. The configuration of this system is shown in Figure 3.

The learning procedure for the GA-based FSMC is summarized as follows.

(1) The fuzzy rule base of FSMC (with fixed premise parts and random consequence parts) is constructed. For example, FSMC for system (2.1):

$$\text{FSMC}: \begin{cases} R_1^{(i)}: \text{ IF } S \text{ is } PB(4, 0.424) & \text{THEN } u \text{ is } \hat{u}_1^{(i)}(\hat{\theta}_1^{(i)}), \\ R_2^{(i)}: \text{ IF } S \text{ is } PM(3.2, 0.424) & \text{THEN } u \text{ is } \hat{u}_2^{(i)}(\hat{\theta}_2^{(i)}), \\ \vdots & \\ R_N^{(i)}: \text{ IF } S \text{ is } PB(-4, 0.424) & \text{THEN } u \text{ is } \hat{u}_N^{(i)}(\hat{\theta}_N^{(i)}), \end{cases} \tag{3.2}$$

where $\hat{u}_j^{(i)}$ is an unknown linguistic label for the control u; $\hat{\theta}_j^{(i)}$ is the adjustable parameter, which have to be encoded as binary strings for genetic operations.

(2) Encode each parameter, $\hat{\theta}_j^{(i)}$ $(i = 1, 2, \ldots, M;\ j = 1, 2, \ldots, N)$, to a d-bit binary code, $P_j^{(i)}(h) = (b_j^1 b_j^2 \cdots b_j^d)(h) = \text{enc}(\hat{\theta}_j^{(i)}(h))$, where $b_j^1, b_j^2, \ldots, b_j^d \in \{0, 1\}$ and $\text{enc}(*)$ denote the encoding operator which encodes the real values to the corresponding binary codes and synthesizes the chromosome of the ith individual.

(3) Establish the population for generation h, $P_j(h) = \{P_j^{(1)}(h), P_j^{(2)}(h), \ldots, P_j^{(M)}(h)\}$, where M is the population size, and every individual $P_j^{(i)}(h)$ corresponds to a binary-code parameter of an FSMC candidate.

(4) Evaluate the fitness value of each individual. The fitness function F is defined as $F = 1/(w\|s(k)\| + v\|u(k)\| + \varepsilon_0)$, where $k = \text{int}(t/\Delta t)$ denotes the iteration instance; Δt is the sampling period; $\text{int}(*)$ is the rounding off operator; w and v are positive weights; ε_0 is a very small positive constant used to avoid the numerical error of dividing by zero.

(5) Based on the fitness value of the individual, keep the best and apply the genetic operators. Assuming that the population size M is 12, pick the top ten-fitted individuals in $P_j(h)$ to apply as genetic operators, that is, reproduction, crossover, mutation (assuming the mutation rate is 0.03125), and keep the top two fitted individuals to generate a new population $P_j(h+1)$, as the offspring of $P_j(h)$.

(6) Decode each binary code to its real value and use this to calculate the control u, then apply u to the system (2.1).

(7) Set $h = h + 1$; go to Step 2, and repeat the aforementioned procedure until $F \geq F_M$ or $h \geq H$, where F_M and H denote an acceptable specific fitness value and the top generation number, respectively, as specified by the designer.

In general, there are at least four methods for the construction of a fuzzy rule base: (1) from expert knowledge or operator experience; (2) modeling an operator's control action; (3) modeling a process; (4) generating fuzzy rules by training, self-organizing, and self-learning algorithms. In Figure 3, GA is used as the learning and training mechanism. The use of the GA means that the second, third, and fourth approaches also provide an efficient way to obtain a fuzzy rule base. Although there are several methods that can provide excellent results in this kind of modeling [36–38], we are convinced that GAs are the most advantageous way to extract an optimal, or at least suboptimal fuzzy rule base for the initial values of the consequent parameter vector of the FSMC or AFSMC.

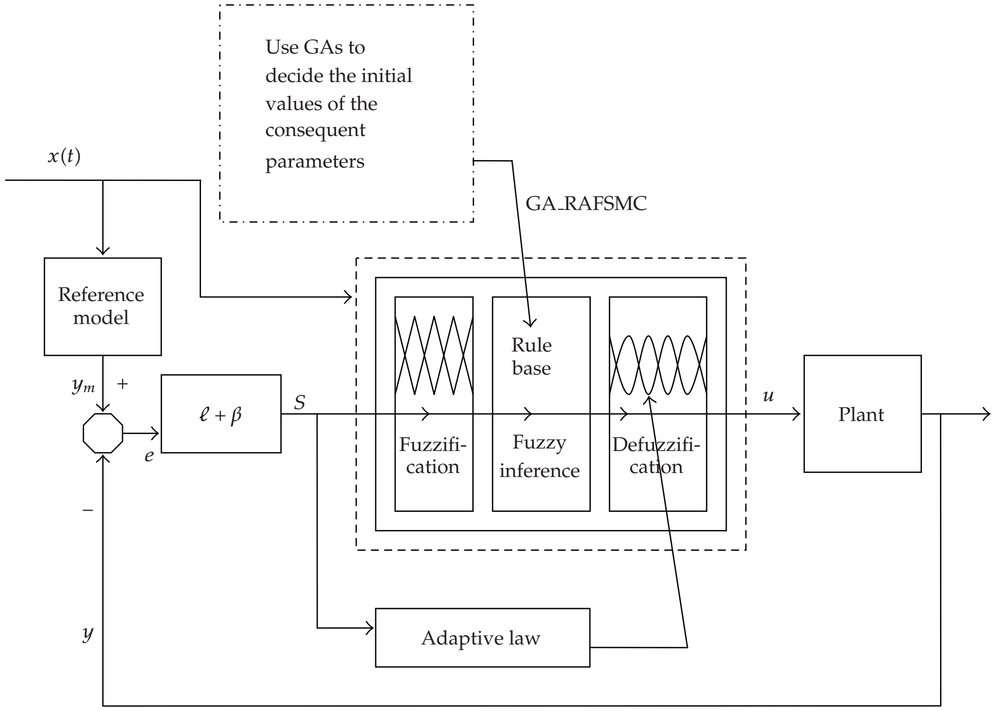

Figure 4: GA_RAFSMC system.

4. GA-based RAFSMC for nonlinear systems

A schematic representation of the GA_RAFSMC system is shown in Figure 4. If $f(x)$, $g(x)$ are known, we can design the FLC (4.1) to approximate u:

$$\overline{u}(\overline{\theta}) = \sum_{k=1}^{m} R_k\left(-\left(\frac{\|S_i - C_{ki}\|}{\beta}\right)^2\right)\cdot\overline{\theta}_k, \tag{4.1}$$

where m is the sum of the fuzzy rules, $\overline{\theta}_k$, that is, $|\overline{\theta}_k| \le \theta_{\max}$ indicate the adjustable consequent parameters of the FLC, and $R(S) = [R_1(S), R_2(S), \ldots, R_m(S)]^T$ is the vector of fuzzy basis function [23] which is defined as

$$R_k(S) = R_k(\|S_i - C_{ki}\|) = \frac{\prod_{i=1}^{n}\mu_k(\|S_i - C_{ki}\|)}{\sum_{k=1}^{m}\left[\prod_{i=1}^{n}\mu_k(\|S_i - C_{ki}\|)\right]}, \tag{4.2}$$

where $k = 1, \ldots, m$ and $i = 1, \ldots, n$ with μ_k represent the degree of membership. The S_i in μ_k can be chosen by

$$\mu_k(\|S_i - C_{ki}\|) = \exp\left(-\left(\frac{\|S_i - C_{ki}\|}{\beta}\right)^2\right). \tag{4.3}$$

Since here n, the sum of input variables, is only one, we know that

$$R_k(S) = \frac{\mu_k(S - C_k)}{\sum_{k=1}^{m} \mu_k(S - C_k)}, \tag{4.4}$$

where $k = 1, \ldots, m$ with μ_k represent the degree of membership. The S in μ_k can be chosen by $\mu_k(\|S - C_k\|) = \exp(-(\|S - C_k\|/\beta)^2)$.

From the approximation property of the fuzzy system, an uncertain and nonlinear plant can be well approximated and described via a fuzzy model with FLC rules to achieve the control object [14, 39, 40].

Assumption 4.1. For $x \in \zeta \subset R^n$, there exists an adjustable parameter vector $\overline{\theta} = [\overline{\theta}_1, \overline{\theta}_2, \ldots, \overline{\theta}_m]^T$ such that the fuzzy system $\overline{u}(S, \overline{\theta}) = \overline{\theta}^T R(S)$ can approximate a continuous function u with accuracy $\varepsilon_{\max}$ over the set ζ, that is, $\exists\, \overline{\theta}$, such that

$$\sup |\, \overline{u}(S, \overline{\theta}) - u(S) \,| \le \varepsilon_{\max}, \quad \forall S \in \zeta. \tag{4.5}$$

Let $\widehat{\theta}$ denote the estimate of $\overline{\theta}$ at time t. Now, we can define the estimated control output $\widehat{u}(S, \widehat{\theta})$ by

$$\widehat{u}(S, \widehat{\theta}) = \sum_{k=1}^{m} \widehat{\theta}_k \cdot R_k(S) = \widehat{\theta}^T R(S), \tag{4.6}$$

and decide on the initial values of the consequent parameter vector $\widehat{\theta} = [\widehat{\theta}_1, \widehat{\theta}_2, \ldots, \widehat{\theta}_m]^T$ based on the genetic algorithm.

First, define the parameter error vector at time t by $\widetilde{\theta} = \overline{\theta} - \widehat{\theta}$, and then

$$\widetilde{\theta}^T R(S) = \overline{u}(S, \overline{\theta}) - \widehat{u}(S, \widehat{\theta}). \tag{4.7}$$

According to assumption 4.1, we can define the modeling error

$$\varepsilon = u - \overline{u}(S, \overline{\theta}), \tag{4.8}$$

where $|\varepsilon| \le \varepsilon_{\max}$.

We can say that

$$u = \widehat{u}(S, \widehat{\theta}) + \widetilde{\theta}^T R(S) + \varepsilon. \tag{4.9}$$

Now, by substituting (4.9) into (2.5), we obtain the error dynamic equation:

$$\overset{(n)}{e} + \alpha_{n-1} \overset{(n-1)}{e} + \cdots + \alpha_1 \dot{e} + \alpha_0 e = g(x) \cdot \left(\widetilde{\theta}^T R(S) + \varepsilon\right). \tag{4.10}$$

We now define the augmented error as

$$S = \beta_{n-1} \overset{(n-1)}{e} + \cdots + \beta_1 \dot{e} + \beta_0 e, \tag{4.11}$$

where $\beta_{n-1}, \ldots, \beta_1, \beta_0$ in (4.11), and $\alpha_{n-1}, \ldots, \alpha_1, \alpha_0$ in (4.10) are chosen such that

$$\widehat{M}(\ell) = \frac{\beta_{n-1}\ell^{n-1} + \cdots + \beta_1 \ell + \beta_0}{\ell^n + \alpha_{n-1}\ell^{n-1} + \cdots + \alpha_1 \ell + \alpha_0} = \frac{N(\ell)}{D(\ell)} \tag{4.12}$$

is strictly positive real (SPR) transfer function, and $N(\ell)$ and $D(\ell)$ are coprime. Now, S and $g(x)\cdot(\tilde{\theta}^T R(S) + \varepsilon)$ can be related by

$$L\{S(t)\} = \widehat{M}(\ell) \cdot L\{g(x) \cdot (\tilde{\theta}^T R(S) + \varepsilon)\}, \tag{4.13}$$

where $L\{\cdot\}$ is the *Laplace* transform of the function, and ℓ denotes the complex *Laplace* transform variable.

If we define $e_m = [e, \ldots, \overset{(n-1)}{e}]^T$ as the states of (4.10), then (4.10) can be realized as

$$\dot{e}_m(t) = \Lambda \cdot e_m(t) + b \cdot [g(x) \cdot (\tilde{\theta}^T R(S) + \varepsilon)], \tag{4.14}$$

$$S(t) = c^T e_m(t), \tag{4.15}$$

where

$$\Lambda = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ -\alpha_0 & -\alpha_1 & -\alpha_2 & \cdots & -\alpha_{n-2} & -\alpha_{n-1} \end{bmatrix}_{n\times n}, \quad b = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 1 \end{bmatrix}_{n\times 1}, \tag{4.16}$$

$$c = [\beta_0 \; \beta_1 \; \cdots \; \beta_{n-1}]^T, \quad \text{let } \beta_{n-1} = 1.$$

According to the *Kalman-Yakubovich* lemma, when $\widehat{M}(\ell)$ is SPR, there exist symmetric and positive definite matrices P and Q such that

$$\begin{aligned} P\Lambda + \Lambda^T P &= -Q, \\ Pb &= c, \quad \text{for } i = 1, \ldots, p. \end{aligned} \tag{4.17}$$

Next, we investigate the asymptotic stability of the origin using *Lyapunov's* function candidates. First, define a *Lyapunov* candidate function as

$$V(e_m, \tilde{\theta}) = \eta \cdot e_m^T P e_m + \tilde{\theta}^T H_{11} \tilde{\theta}, \tag{4.18}$$

where η is a positive constant representing the learning rate

$$\tilde{\theta} = [\,\tilde{\theta}_1\ \tilde{\theta}_2\ \cdots \tilde{\theta}_m]^T, \qquad H_{11} = g(x)\cdot I_{m\times m},$$
$$\tilde{\theta}^T H_{11} = \begin{bmatrix} g(x)\cdot\tilde{\theta}_1 & 0 & \cdots & 0 \\ 0 & g(x)\cdot\tilde{\theta}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & g(x)\cdot\tilde{\theta} \end{bmatrix}_{m\times m}, \quad m: \text{ the sum of the fuzzy rules.} \tag{4.19}$$

If $e_m^T P e_m > \phi^2$, the derivate of $V(e_m, \tilde{\theta})$ along the trajectories of the system should be negative definite for all nonlinearities that satisfy a given sector condition (*Lyapunov's* stability):

$$\dot{V}(e_m, \tilde{\theta}) = \eta\cdot(\dot{e}_m^T P e_m + e_m^T P \dot{e}_m) + 2\tilde{\theta}^T H_{11}\dot{\tilde{\theta}}. \tag{4.20}$$

As mentioned above $\tilde{\theta} = \overline{\theta} - \hat{\theta}$, and we can infer that $\dot{\tilde{\theta}} = -\dot{\hat{\theta}}$, and

$$\begin{aligned}\dot{V} &= \eta\cdot(e_m^T\Lambda^T P e_m + e_m^T P\Lambda e_m) + 2\eta\cdot e_m^T Pb\cdot\left[g(x)\cdot(\tilde{\theta}^T R(S)+\varepsilon)\right] + 2\cdot\tilde{\theta}^T H_{11}(-\dot{\hat{\theta}}) \\ &= \eta\cdot e_m^T(-Q)e_m + 2\eta\cdot S\cdot\left[g(x)\cdot(\tilde{\theta}^T R(S)+\varepsilon)\right] + 2\cdot\tilde{\theta}^T H_{11}(-\dot{\hat{\theta}}).\end{aligned} \tag{4.21}$$

In general, chattering must be eliminated for the controller to perform properly. This can be achieved by smoothing out control discontinuity in a thin boundary layer neighboring the switching surface. To amend the modeling error ε and the chattering phenomenon, we propose a modified adaptive law (4.22) with which to tune the adjustable consequent parameters of the RAFSMC:

$$\dot{\hat{\theta}} = \eta\cdot|S|\cdot R(S)\cdot \operatorname{sat}\left(\frac{S}{\Phi}\right). \tag{4.22}$$

The thin boundary layer function $\operatorname{sat}(S/\Phi)$ is defined as

$$\operatorname{sat}\left(\frac{S}{\Phi}\right) = \begin{cases} 1, & \text{if } \left(\dfrac{S}{\Phi}\right) > 1, \\ \left(\dfrac{S}{\Phi}\right), & \text{if } -1 \le \left(\dfrac{S}{\Phi}\right) \le 1, \\ -1, & \text{if } \left(\dfrac{S}{\Phi}\right) < -1, \end{cases} \tag{4.23}$$

where $\Phi > 0$ is the thickness of the boundary layer.

If we substitute (4.22) into (4.21), then (4.21) becomes

$$\dot{V} = -\eta\cdot e_m^T Q e_m + 2\eta\cdot S\cdot\left[g(x)\cdot(\tilde{\theta}^T R(S)+\varepsilon)\right] - 2\eta\cdot|S|\cdot\left[g(x)\cdot\tilde{\theta}^T R(S)\right]\cdot\operatorname{sat}\left(\frac{S}{\Phi}\right). \tag{4.24}$$

When $|S| > \Phi$, then

$$\begin{aligned}\dot{V} &= -\eta \cdot e_m^T Q e_m + 2\eta \cdot e_m^T c \cdot (g(x) \cdot \varepsilon) \\ &\le -\eta \cdot \|e_m\|^2 \cdot Q + 2\eta \cdot \|e_m\| \cdot \|c\| \cdot \|g(x) \cdot \varepsilon\| \\ &\le -\eta \cdot \|e_m\| \cdot [\|e_m\| \cdot Q - 2\|c\| \cdot \|g(x) \cdot \varepsilon\|].\end{aligned} \tag{4.25}$$

If μ is positive and small enough, then $\phi > 0$ and $\sigma > 0$, such that

$$\left\{ \frac{\phi Q}{\sqrt{P}} - 2\|c\| \cdot \|g(x) \cdot \varepsilon\| \right\} > \sigma, \tag{4.26}$$

where $e_m^T P e_m > \phi^2$.

It is real that $\dot{V} \le -\eta \cdot \|e_m\| \cdot \sigma$ if $e_m^T P e_m > \phi^2$ and $|S| > \Phi$, and hence $\dot{V} < 0$. Thus V will gradually converge to zero as all the ς.

Based on the above inference and *Lyapunov's* stability theory, e_m will gradually converge inside the bounded zone $|e_m| \le (\phi/\sqrt{P}, \Phi/\beta_0)$. The tracking error and the modeling error will then both approach zero.

Theorem 4.2. *Consider a nonlinear uncertain system* $\overset{(n)}{y} = f(x) + g(x) \cdot u + d$ *that satisfies the assumptions* $(\overline{\theta}, \widehat{\theta})$. *Suppose that the unknown control input* u *can be approximated by* $\widehat{u}(S, \widehat{\theta})$ *as in* (4.6). *Now,* S *is given by* (4.15), *and* Q *is a symmetric positive definite weighting matrix.*

5. Numerical simulation

In this section, the proposed GA-based RAFSMC is demonstrated with an example of the control methodology.

Consider the problem of balancing an inverted pendulum on a cart as shown in Figure 5. The dynamic equations of motion of the pendulum are given below [27]:

$$\begin{aligned}\dot{x}_1 &= x_2, \\ \dot{x}_2 &= \frac{g \cdot \sin(x_1) - aml\, x_2^2 \sin(2x_1)/2 - a\cos(x_1) \cdot u}{4l/3 - aml\, \cos^2(x_1)},\end{aligned} \tag{5.1}$$

where x_1 denotes the angle (in radian) of the pendulum from the vertical; and x_2 is the angular vector. Thus the gravity constant $g = 9.8\ \mathrm{m/s^2}$, where m is the mass of the pendulum, M is the mass of the cart, l is the length of F (input force), s is the force applied to the cart (in Newtons), and $a = 1/(m + M)$. The parameters chosen for the pendulum in this simulation are m =0.1 kg, M =1 kg, and l =0.5 m.

The control objective in this example is to balance the inverted pendulum in the approximate range $x \in (-\pi/2, \pi/2)$. The GA-based RAFSMC designed based on the procedure discussed above will have the following steps.

Step 1. Specify the response of the control system by defining a suitable sliding surface

$$S = c^T e_m = 5e + \dot{e}\ [27]. \tag{5.2}$$

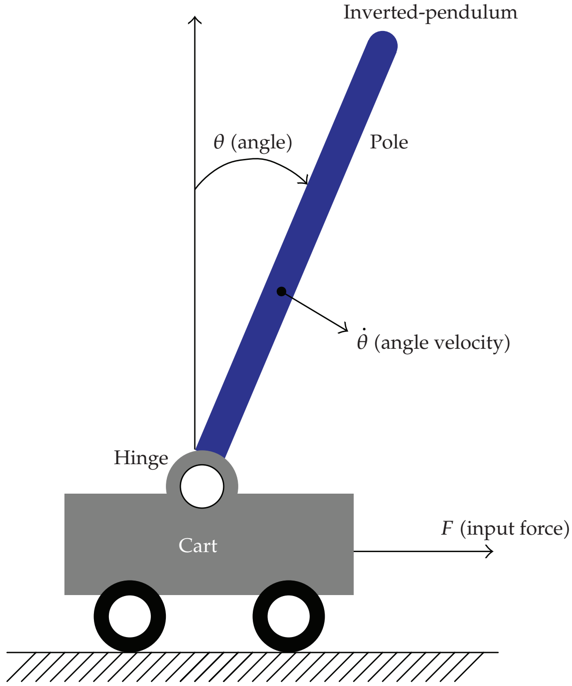

Figure 5: Inverted pendulum system.

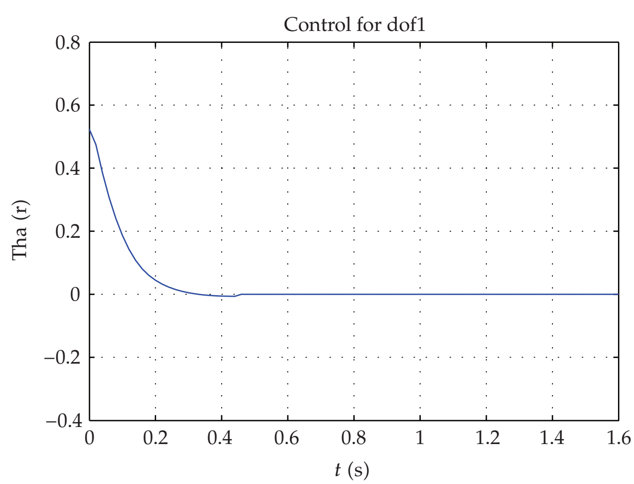

Figure 6: Angle response of the pendulum with the initial condition $x_1(0) = 30^\circ$.

Step 2. Construct the fuzzy rule base (3.2) and the fuzzy models (4.6) based on the genetic algorithm. After carrying out the abovementioned genetic-based learning procedure, the number of individual strings is 10, the size of population M is 12, the crossover rate is 0.8333, the mutation rate is 0.03125, and the maximum number of the generations H is 15. Now, the

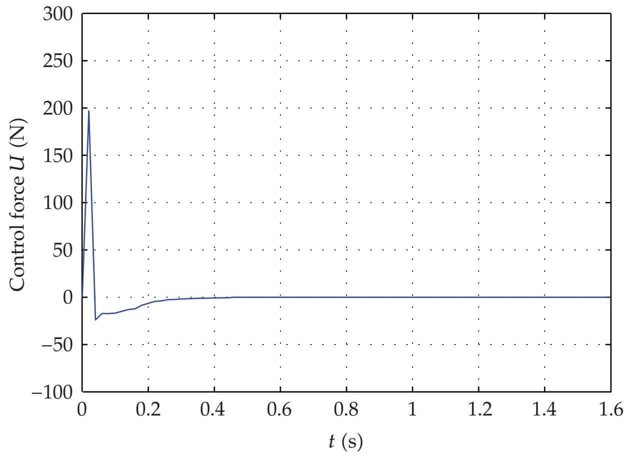

Figure 7: Control force in the pendulum system with the initial $x_1(0) = 30°$.

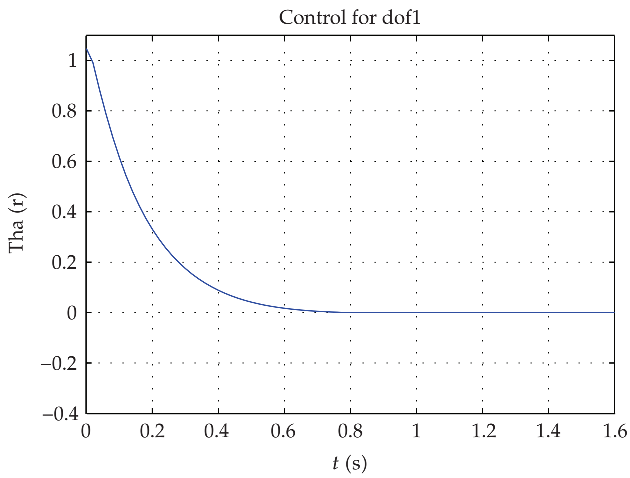

Figure 8: Angle response of the pendulum with the initial condition $x_1(0) = 60°$.

initial values of the consequent parameter vector $\widehat{\theta}$ for the GA-based RAFSMC can be chosen as follows:

$$[1,\ 0.6263,\ 0.4113,\ 0.2100,\ 0.0850,\ 0\,, -0.0850, -0.2100, -0.4113, -0.6263,\ -1]^T. \tag{5.3}$$

Step 3. Apply the controller as given by (4.6) to control the nonlinear system (2.1). Now, let $\eta = 10$, $\Phi = 0.3$, and adjust $\widehat{\theta}$ by the adaptive law as given by (4.22).

Therefore, based on Theorem 4.2, the proposed GA-based RAFSMC can asymptotically stabilize the inverted pendulum. The simulation results are illustrated in Figures 6–9. The initial conditions are $x_1(0) = 30°,\ 60°$, and $x_2(0) = 0$.

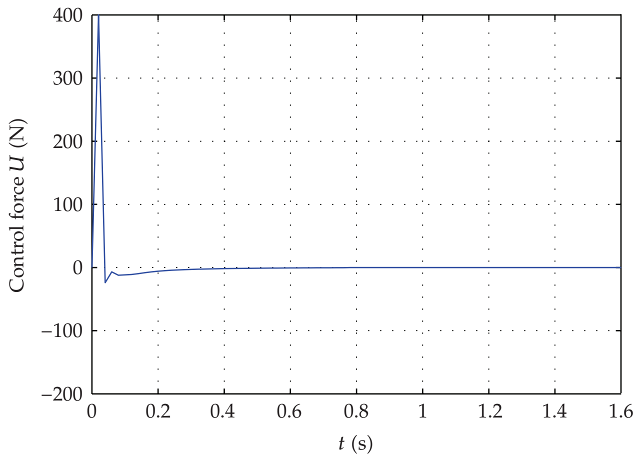

Figure 9: Control force in the pendulum system with the initial condition $x_1(0) = 60°$.

Figures 6–9 show that the inverted pendulum system (compare with Yoo and Ham [27]) is rapidly, asymptotically stable because the system trajectory starts from any nonzero initial state, to rapidly and asymptotically approach the origin.

6. Conclusion

The stability analysis of a GA-based reference adaptive fuzzy sliding model controller for a nonlinear system is discussed. First, we track the reference trajectory for an uncertain and nonlinear plant. We make sure that it is well approximated and described via the fuzzy model involving FLC rules. Then we decide on the initial values of the consequent parameter vector $\widehat{\theta}$ via a GA. Next, an adaptive fuzzy sliding model controller is proposed to simultaneously stabilize and control the system. A stability criterion is also derived from *Lyapunov*'s direct method to ensure stability of the nonlinear system. Finally, we discuss an example and provide a numerical simulation. From this example, we see that the stability of the inverted pendulum system is ensured because the trajectories from nonzero initial states approach to zero by proposed controller design, and the results demonstrate that with this control methodology we can rapidly and efficiently control a complex and nonlinear system.

Acknowledgments

The authors would like to thank the National Science Council of the Republic of China, Taiwan for financial support of this research under Contract no. NSC 96-2628-E-366-004-MY2. The authors are also most grateful for the kind assistance of Professor Balthazar, Editor of special issue, and the constructive suggestions from anonymous reviewers all of which has led to the making of several corrections and suggestions that have greatly aided us in the presentation of this paper.

References

[1] G. J. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic: Theory and Applications*, Prentice Hall, Upper Saddle River, NJ, USA, 1995.

[2] W.-J. Wang and H.-R. Lin, "Fuzzy control design for the trajectory tracking on uncertain nonlinear systems," *IEEE Transactions on Fuzzy Systems*, vol. 7, no. 1, pp. 53–62, 1999.

[3] X.-J. Ma, Z.-Q. Sun, and Y.-Y. He, "Analysis and design of fuzzy controller and fuzzy observer," *IEEE Transactions on Fuzzy Systems*, vol. 6, no. 1, pp. 41–51, 1998.

[4] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 15, no. 1, pp. 116–132, 1985.

[5] F.-H. Hsiao, C. W. Chen, Y.-H. Wu, and W.-L. Chiang, "Fuzzy controllers for nonlinear interconnected TMD systems with external force," *Journal of the Chinese Institute of Engineers*, vol. 28, no. 1, pp. 175–181, 2005.

[6] T.-Y. Hsieh, M. H. L. Wang, C. W. Chen, et al., "A new viewpoint of s-curve regression model and its application to construction management," *International Journal on Artificial Intelligence Tools*, vol. 15, no. 2, pp. 131–142, 2006.

[7] C.-H. Tsai, C. W. Chen, W.-L. Chiang, and M.-L. Lin, "Application of geographic information system to the allocation of disaster shelters via fuzzy models," *Engineering Computations*, vol. 25, no. 1, pp. 86–100, 2008.

[8] C. W. Chen, W.-L. Chiang, C.-H. Tsai, C.-Y. Chen, and M. H. L. Wang, "Fuzzy Lyapunov method for stability conditions of nonlinear systems," *International Journal on Artificial Intelligence Tools*, vol. 15, no. 2, pp. 163–171, 2006.

[9] K. Tanaka, T. Hori, and H. O. Wang, "A multiple Lyapunov function approach to stabilization of fuzzy control systems," *IEEE Transactions on Fuzzy Systems*, vol. 11, no. 4, pp. 582–589, 2003.

[10] B. Chen, X. Liu, and S. Tong, "New delay-dependent stabilization conditions of T-S fuzzy systems with constant delay," *Fuzzy Sets and Systems*, vol. 158, no. 20, pp. 2209–2224, 2007.

[11] C. W. Chen, C.-L. Lin, C.-H. Tsai, C.-Y. Chen, and K. Yeh, "A novel delay-dependent criterion for time-delay T-S fuzzy systems using fuzzy Lyapunov method," *International Journal on Artificial Intelligence Tools*, vol. 16, no. 3, pp. 545–552, 2007.

[12] F.-H. Hsiao, J.-D. Hwang, C. W. Chen, and Z.-R. Tsai, "Robust stabilization of nonlinear multiple time-delay large-scale systems via decentralized fuzzy control," *IEEE Transactions on Fuzzy Systems*, vol. 13, no. 1, pp. 152–163, 2005.

[13] K. Yeh, C.-Y. Chen, and C. W. Chen, "Robustness design of time-delay fuzzy systems using fuzzy Lyapunov method," *Applied Mathematics and Computation*, vol. 205, no. 2, pp. 568–577, 2008.

[14] C. W. Chen, K. Yeh, W.-L. Chiang, C.-Y. Chen, and D.-J. Wu, "Modeling, H^{∞} control and stability analysis for structural systems using Takagi-Sugeno fuzzy model," *Journal of Vibration and Control*, vol. 13, no. 11, pp. 1519–1534, 2007.

[15] G. Feng, C.-L. Chen, D. Sun, and Y. Zhu, "H^{∞} controller synthesis of fuzzy dynamic systems based on piecewise Lyapunov functions and bilinear matrix inequalities," *IEEE Transactions on Fuzzy Systems*, vol. 13, no. 1, pp. 94–103, 2005.

[16] F.-H. Hsiao, C. W. Chen, Y.-W. Liang, S.-D. Xu, and W.-L. Chiang, "T-S fuzzy controllers for nonlinear interconnected systems with multiple time delays," *IEEE Transactions on Circuits and Systems I*, vol. 52, no. 9, pp. 1883–1893, 2005.

[17] S. Xu and J. Lam, "Robust H^{∞} control for uncertain discrete-time-delay fuzzy systems via output feedback controllers," *IEEE Transactions on Fuzzy Systems*, vol. 13, no. 1, pp. 82–93, 2005.

[18] C. W. Chen, "Modeling and control for nonlinear structural systems via a NN-based approach," *Expert Systems with Applications*. In press.

[19] C.-Y. Chen, J. R.-C. Hsu, and C. W. Chen, "Fuzzy logic derivation of neural network models with time delays in subsystems," *International Journal on Artificial Intelligence Tools*, vol. 14, no. 6, pp. 967–974, 2005.

[20] P. C. Chen, C. W. Chen, and W. L. Chiang, "GA-based modified adaptive fuzzy sliding mode controller for nonlinear systems," *Expert Systems with Applications*. In press.

[21] S. Limanond and J. Si, "Neural-network-based control design: an LMI approach," *IEEE Transactions on Neural Networks*, vol. 9, no. 6, pp. 1422–1429, 1998.

[22] R. Palm, "Robust control by fuzzy sliding mode," *Automatica*, vol. 30, no. 9, pp. 1429–1437, 1994.

[23] L. X. Wang, *A Course in Fuzzy Systems and Control*, Prentice Hall, Englewood Cliffs, NJ, USA, 1997.

[24] V. I. Utkin, *Sliding Modes and Their Application in Variable Structure Systems*, MIR Publishers, Moscow, Russia, 1978.

[25] J. J. E. Slotine and W. Li, *Applied Nonlinear Control*, Prentice Hall, Englewood Cliffs, NJ, USA, 1991.

[26] Y. Xia and Y. Jia, "Robust sliding-mode control for uncertain time-delay systems: an LMI approach," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 1086–1092, 2003.

[27] B. Yoo and W. Ham, "Adaptive fuzzy sliding mode control of nonlinear system," *IEEE Transactions on Fuzzy Systems*, vol. 6, no. 2, pp. 315–321, 1998.

[28] S. Tong and H.-X. Li, "Fuzzy adaptive sliding-mode control for MIMO nonlinear systems," *IEEE Transactions on Fuzzy Systems*, vol. 11, no. 3, pp. 354–360, 2003.

[29] S. Labiod, M. S. Boucherit, and T. M. Guerra, "Adaptive fuzzy control of a class of MIMO nonlinear systems," *Fuzzy Sets and Systems*, vol. 151, no. 1, pp. 59–77, 2005.

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

[31] G. Feng, S. G. Cao, and N. W. Rees, "Stable adaptive control for fuzzy dynamic systems," *Fuzzy Sets and Systems*, vol. 131, no. 2, pp. 217–224, 2002.

[32] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, Mass, USA, 1989.

[33] S. C. Lin, *Stable self-learning optimal fuzzy control system design and application*, Ph.D. dissertation, Department of Electrical Engineering, National Taiwan University, Chung-li, Taiwan, 1997.

[34] P. C. Chen, *Genetic algorithm for control of structure system*, M.S. thesis, Department of Civil Engineering, Chung Yuan University, Taipei, Taiwan, 1998.

[35] C. C. Liu and F. C. Chen, "Adaptive control of nonlinear continuous-time systems using neural networks—general relative degree and MIMO cases," *International Journal of Control*, vol. 58, no. 2, pp. 317–335, 1993.

[36] J.-S. R. Jang, C.-T. Sun, and E. Mizutani, *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*, Prentice-Hall, Upper Saddle River, NJ, USA, 1997.

[37] J.-S. R. Jang, "ANFIS: adaptive-network-based fuzzy inference system," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 23, no. 3, pp. 665–685, 1993.

[38] F. J. de Souza, M. M. R. Vellasco, and M. A. C. Pacheco, "Hierarchical neuro-fuzzy quadtree models," *Fuzzy Sets and Systems*, vol. 130, no. 2, pp. 189–205, 2002.

[39] C. W. Chen, W. L. Chiang, and F. H. Hsiao, "Stability analysis of T-S fuzzy models for nonlinear multiple time-delay interconnected systems," *Mathematics and Computers in Simulation*, vol. 66, no. 6, pp. 523–537, 2004.

[40] C. W. Chen, "Stability conditions of fuzzy systems and its application to structural and mechanical systems," *Advances in Engineering Software*, vol. 37, no. 9, pp. 624–629, 2006.