ELSEVIER

Fuzzy Sets and Systems 125 (2002) 23–38

FUZZY sets and systems

www.elsevier.com/locate/fss

Robust fuzzy control of nonlinear systems using shape-adaptive radial basis functions

Hugang Han[a,*], Chun-Yi Su[b]

[a]School of Business, Hiroshima Prefectural University, 562 Nanatuska-cho, Shobara-shi, Hiroshima 727-0023, Japan
[b]Department of Mechanical Engineering, Concordia University, 1455 de Maisonneuve Blvd. W., Montreal, Que., Canada H3G 1M8

Received 13 August 1999; received in revised form 21 March 2000; accepted 21 November 2000

Abstract

Recently, various adaptive fuzzy control schemes have been proposed to deal with nonlinear systems whose dynamics are poorly understood by using the parameterized fuzzy approximators. However, most of the schemes presented so far can only tune the parameters, which appear linearly in the parameterized fuzzy approximators. The major disadvantage is that the desired approximation error may not be warranted and consequently, the control performance could be influenced. In this paper, we tune not only the linear parameters but also the parameters, which appear nonlinearly in the fuzzy approximators, to reduce the approximation error and to improve control performance. The proposed adaptive controller ensures global stability of the overall adaptive system and achieves desired tracking.

Keywords: Fuzzy control; Fuzzy approximator; Adaptive control; Nonlinear parameterization; Global stability; Nonlinear systems

1. Introduction

The application of fuzzy set theory to control problems has been the focus of numerous studies [1]. The motivation is often that fuzzy set theory provides an alternative to the traditional modeling and design of control systems where system knowledge and dynamic models in the traditional sense are uncertain and time varying. In spite of many practical successes, fuzzy control has not been viewed as a rigorous approach due to the lack of formal synthesis techniques, which can guarantee the basic requirements for control system such as global stability. Recently, Lyapunov synthesis approach has successfully been used to construct stable fuzzy controllers [2]. Therein, much attention has been paid to the systems with poorly understood dynamics and several adaptive fuzzy control schemes have been proposed by using the parameterized fuzzy approximators [2–6]. However, most of those schemes can only tune the parameters, which appear linearly in the parameterized fuzzy approximators [2,3,5,6]. The major disadvantage is that the desired approximation error may not

* Corresponding author. Tel.: +81-8247-41706; fax: +81-8247-40191.
E-mail address: kan@bus.hiroshima-pu.ac.jp (H. Han).

0165-0114/02/$ - see front matter
PII: S0165-0114(01)00026-4

be warranted and consequently, the control performance could be influenced. Until now, very few results are available for the adjustment of nonlinearly parameterized systems [23]. Though the gradient approaches were used [11–13], the way of fusing into the fuzzy control schemes to generate global stability is still an open problem.

In this paper, our discussion is focused on a well-used parameterized fuzzy approximator, which is expressed as a series of radial basis function (RBF) expansion [7]. The reason is its best approximation property. In the RBF expansion, three parameter vectors are used: connection weights, variances, and centers. It is obvious that as changes of these parameters, the bell-shaped radial functions will vary accordingly, and so will exhibit various forms of shapes. This property could be utilized to capture the fast changing system dynamics, reduce approximation error, and improve control performance. When centers are defined on a regular mesh, as shown in [10], the approximation error of the RBF expansion relies crucially on the choices of the weights and variances. Since the variances appear nonlinearly in RBF, the determination of adaptive law for such systems is a nontrivial task. In this paper, with the help of concave/convex optimization technique in [15], a new adaptive law is introduced, which not only tunes the linear parameters but also the nonlinear parameters. The proposed adaptive controller ensures global stability of the overall adaptive system and achieves the desired tracking. Simulations performed on different nonlinear systems illustrate and clarify the approach.

The arrangement of this article is as follows: Section 2 addresses the problem that will be discussed in this paper. Section 3 is devoted to the general problem of function approximation using fuzzy IF–THEN rules from human experts. In Section 4 an adaptive controller is developed to tune the parameters that appear nonlinearly, while at the same time guaranteeing the global stability of the closed loop systems. In Section 5, the performance of the proposed adaptive controller is verified through simulations and conclusion is given in Section 6. A min–max optimization technique [15] used extensively in the paper is briefly reviewed in the appendix.

2. Problem statement

This paper focuses on the design of adaptive control algorithms for a class of dynamic systems whose equation of motion can be expressed in the canonical form as follows:

$$x^{(n)}(t) + f(x(t),\dot{x}(t),\ldots,x^{(n-1)}(t)) = b(x(t),\dot{x}(t),\ldots,x^{(n-2)}(t))u(t), \tag{1}$$

where $u(t)$ is the control input, $f(\cdot)$ is an *unknown* linear or nonlinear function and $b(\cdot)$ is the control gain. It should be noted that more general classes of nonlinear systems could be transformed into this structure [22].

The control objective is to force the state $X(t)=[x(t),\dot{x}(t),\ldots,x^{(n-1)}(t)]^{\mathrm{T}}$ to follow a specified desired trajectory, $X_{\mathrm{d}}(t)=[x_{\mathrm{d}}(t),\dot{x}_{\mathrm{d}}(t),\ldots,x_{\mathrm{d}}^{(n-1)}(t)]^{\mathrm{T}}$. Defining the tracking error vector, $\tilde{X}(t)=X(t)-X_{\mathrm{d}}(t)$, the problem is to design a control law $u(t)$ which ensures that $\tilde{X}(t)\to 0$ as $t\to\infty$. For simplicity in this initial discussion, we take $b=1$ in the subsequent development.

Most of the results in adaptive control have focused on the situation where an explicit linear parameterization of the unknown function $f(X)$ is possible. The parameterization $f(X)$ can be expressed as $f(X)=\sum_{j=1}^{N}\theta_j Y_j(X)$, where θ_j is a set of unknown parameters which appear linearly, and $Y_j(X)$ is a set of known regressors or basis functions. Then, the function $f(X)$ can be approximated as $\hat{f}(X)=\sum_{j=1}^{N}\hat{\theta}_j Y_j(X)$. As the approximation error only occurs on the parameters $\hat{\theta}_j$, adaptive methods can be used to adjust the parameters $\hat{\theta}_j$ to achieve the control objective. The challenge addressed in this paper is the development of adaptive controllers when an explicit linear parameterization of the function $f(X)$ is *either unknown or impossible*. In the following section, it will be shown that using fuzzy IF–THEN rules, the unknown function $f(X)$ can be approximated by a parameterized fuzzy approximator. The parameters in this parameterized fuzzy approximator can then be stably tuned to provide an effective tracking control architecture.

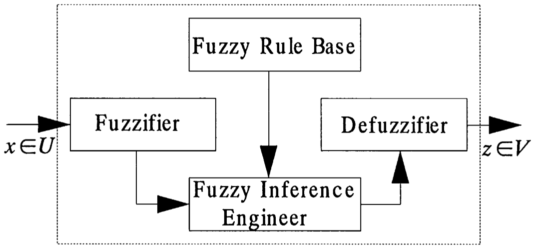

Fig. 1. Basic configuration of fuzzy logic system.

3. Fuzzy systems

Consider a fuzzy system whose basic configuration is shown in Fig. 1 [7]. There are four principal elements in such a fuzzy system: fuzzifier, fuzzy rule base, fuzzy inference engine, and defuzzifier. Let input space $X \in R^n$ be a compact product space. Assume that there are N rules in the rule base and each of which has the following form:

R_j: *IF* x_1 *is* A_j^1 and x_2 *is* A_j^2 and $\cdots$ and x_n *is* A_j^n, *THEN* z *is* B_j,

where $j = 1, 2, \ldots, N$, x_i $(i = 1, 2, \ldots, n)$ are the input variables to the fuzzy system, z is the output variable of the fuzzy system, and A_j^i and B_j are linguistic terms characterized by fuzzy membership functions $\mu_{A_j^i}(x_i)$ and $\mu_{B_j}(z)$, respectively.

As in [3,7], we consider a subset of the fuzzy systems with *singleton fuzzifier*, *product inference*, and *Gaussian membership function*. In this case, such a fuzzy system can be written as

$$h(X) = \sum_{j=1}^{N} \omega_j \left(\prod_{i=1}^{n} \mu_{A_j^i}(x_i) \right), \tag{2}$$

where $h : U \subset R^n \to R$, $X = (x_1, x_2, \ldots, x_n) \in U$; ω_j is the point in R at which $\mu_{B_j}(\omega_j) = 1$, named as the connection weight; $\mu_{A_j^i}(x_i)$ is *the Gaussian membership function*, defined by

$$\mu_{A_j^i}(x_i) = \exp[-(\sigma_j^i(x_i - \xi_j^i))^2], \tag{3}$$

where ξ_j^i, σ_j^i are real-valued parameters.

Remark. Contrary to the traditional notation, in this paper $1/\sigma_j^i$ is used to represent the variance for the convenience of later development.

Definition 1. Define *fuzzy basis functions* (FBFs) as

$$g_j(\sigma_j \|X - \xi_j\|) = \prod_{i=1}^{n} \mu_{A_j^i}(x_i), \quad j = 1, 2, \ldots, N, \tag{4}$$

where $\mu_{A_j^i}(x_i)$ are the Gaussian membership functions defined in (3), $\xi_j = (\xi_j^1, \xi_j^2, \ldots, \xi_j^n)$ and $\sigma_j = (\sigma_j^1, \sigma_j^2, \ldots, \sigma_j^n)$. Then, the fuzzy system (2) is equivalent to a FBF expansion

$$h(X) = \sum_{j=1}^{N} \omega_j g_j(\sigma_j \|X - \xi_j\|). \tag{5}$$

Lemma 1 (Su and Stepanenko [3], Wang and Mendel [7]). *For any given real continuous function f on the compact set $U \in R^n$ and arbitrary $\varepsilon_h > 0$, there exists optimal* FBF expansion $h^*(X) \in A$ *such that*

$$\sup_{X \in U} |f(X) - h^*(X)| < \varepsilon_h. \tag{6}$$

This theorem states that the FBF expansion (5) is a universal approximator on a compact set. Since the fuzzy universal approximator is characterized by the parameters ω_j, σ_j and ξ_j, the optimal $h^*(X)$ contains the optimal parameters ω_j^*, σ_j^* and ξ_j^*.

Remarks. (1) It is important to note that since the fuzzy approximator (5) is constructed from the fuzzy IF–THEN rules, linguistic information describing the plant from a human expert can be directly incorporated to estimate the unknown function f. In this case, one may synthesize the controllers assuming that the fuzzy logic system represents (approximately) the true plant.

(2) We should note that the above membership function could also be selected as other forms such as sigmoidal functions [8] or triangular functions. However, it is shown in [9,10] that Gaussian basis functions do have the best approximation property. This is the principal reason being the selection of the Guassian function to characterize this membership function.

4. Adaptive fuzzy control system

When an explicit linear parameterization of the function $f(X)$ is *either unknown or impossible*, by applying Lemma 1, unknown function $f(X)$ in (1) is approximated by a fuzzy approximator $f^*(X)$,

$$f^*(X) = \sum_{j=1}^{N} \omega_j^* g_j(\sigma_j^* \|X - \xi_j\|) \equiv \sum_{j=1}^{N} \omega_j^* g_j(\sigma_j^*), \tag{7}$$

where the centers ξ_j are placed on a regular mesh, and $\omega_j^* \in R$ and $\sigma_j^{*\mathrm{T}} = (\sigma_j^{1*}, \sigma_j^{2*}, \ldots, \sigma_j^{n*})$ are optimal parameters for approximating the unknown function $f(X)$ in (1). The approximation error ε can then be expressed as

$$\varepsilon \equiv f(X) - f^*(X). \tag{8}$$

To construct $f^*(X)$, the values of the optimal parameters ω_j^* and σ^* are required. However, Lemma 1 does not provide any insight on how to find the optimal parameters ω_j^* and σ^* for a *given* number of fuzzy IF–THEN rules. Normally the unknown parameters values ω_j^* and σ_j^* are replaced by their estimates $\hat{\omega}_j$ and $\hat{\sigma}_j$. Then the estimation function $\hat{f}(X) = \sum_{j=1}^{N} \hat{\omega}_j g_j(\hat{\sigma}_j \|X - \xi_j\|) \equiv \sum_{j=1}^{N} \hat{\omega}_j g_j(\hat{\sigma}_j)$ is used instead of $f^*(X)$ to approximate the unknown function $f(X)$. The parameters in the estimated $\hat{f}(X)$ should then be stably tuned to provide effective tracking control architecture. However, in recent adaptive fuzzy controllers [3,7,14], only parameter vector $\hat{\omega}_j$ which appears linearly in the estimation function, is tuned. This implies that the shape of the fuzzy member function is fixed. As shown in [10], when centers are placed on a regular mesh, the approximation error of the RBF expansion relies crucially on the choices of the weights and variances. Without any doubt, the desirable approach is to tune the weights and variances simultaneously to achieve the best approximation. Since the parameter $\hat{\sigma}_j$ appears nonlinearly in the FBF expansion, it is a nontrivial task for the adjustment of nonlinearly parameterized systems [23]. Though the gradient approaches were used [11–13], the way of fusing this approach with the adaptive fuzzy control schemes to generate global stability, is still an open problem. The challenge addressed here to develop a method that can adaptively adjust the shape of the fuzzy member function to achieve the best approximation. To accomplish such an objective, based on a min–max optimization technique [15], a new adaptive controller, which tunes $\hat{\omega}_j$ and $\hat{\sigma}_j$ simultaneously as described below, is proposed that leads to global stability.

To develop the controller, as in [3,17], an error metric is defined as

$$s(t) = \left(\frac{\mathrm{d}}{\mathrm{d}t} + \lambda\right)^{n-1} \tilde{x}(t) \quad \text{with } \lambda > 0, \tag{9}$$

which can be rewritten as $s(t) = \Lambda^{\mathrm{T}}\tilde{X}(t)$, with $\Lambda^{\mathrm{T}} = [\lambda^{n-1}, (n-1)\lambda^{n-2}, \ldots, 1]$. The equation $s(t) = 0$ defines a time-varying hyperplane in R^n on which the tracking error vector decays exponentially to zero, so that perfect tracking can be asymptotically obtained by maintaining this condition [17]. In this case the control objective becomes the design of a controller to force $s(t) = 0$. The time derivative of the error metric can be written as

$$\dot{s}(t) = -x_{\mathrm{d}}^{(n)}(t) + \Lambda_V^{\mathrm{T}}\tilde{X}(t) + u - f(X), \tag{10}$$

where $\Lambda_V^{\mathrm{T}} = [0, \lambda^{n-1}, (n-1)\lambda^{n-2}, \ldots, (n-1)\lambda]$.

The structure of the plant in Eq. (10) suggests that when $f(X)$ is known, a control input of form

$$u(t) = -k_{\mathrm{d}}s(t) + x_{\mathrm{d}}^{(n)}(t) - \Lambda_V^{\mathrm{T}}\tilde{X}(t) + f(X)$$

leads to a closed-loop system $\dot{s}(t) = -k_{\mathrm{d}}s(t)$, and hence, $\tilde{X}(t) \to 0$ as $t \to \infty$. The question is how $u(t)$ can be determined when $f(X)$ is unknown. Before answering this question, it is reasonable to assume that there exists a known constant $\varepsilon_f > 0$, so that the approximation error ε, defined as in (8), satisfies $|\varepsilon| \leqslant \varepsilon_f$.

Inspired by the above control structure, using the fuzzy approximator $\hat{f}(X) = \sum_{j=1}^{N} \hat{\omega}_j g_j(\hat{\sigma}_j\|X - \xi_j\|) \equiv \sum_{j=1}^{N} \hat{\omega}_j g_j(\hat{\sigma}_j)$ a natural choice for the adaptive control law is

$$u(t) = -k_{\mathrm{d}}s(t) + x_{\mathrm{d}}^{(n)}(t) - \Lambda_V^{\mathrm{T}}\tilde{X}(t) + \sum_{j=1}^{N} \hat{\omega}_j g_j(\hat{\sigma}_j\|X - \xi_j\|) - (a^* + \varepsilon_f)\,\mathrm{sgn}(s), \tag{11}$$

$$\dot{\hat{\omega}}_j(t) = -\lambda_j s(t) g_j(\hat{\sigma}_j), \tag{12}$$

$$\dot{\hat{\sigma}}_j = -\beta_j \kappa_j^* s(t), \tag{13}$$

where λ_j and $\beta_j \in R^{n\times n}$ are rates of adaptation, $a^* \in R$ and $\kappa_j^* \in R^{n\times 1}$ are time functions which are determined as follows.

Before giving $a^* \in R$ and $\kappa_j^* \in R^{n\times 1}$, we should note that Eqs. (12) and (13) are used to adjust $\hat{\omega}_j$ and $\hat{\sigma}_j$ in the fuzzy approximator $\hat{f}(X)$. The role of the term $(a^* + \varepsilon_f)\,\mathrm{sgn}(s)$ overcomes the effects of approximation error due to the estimations of $\hat{\omega}_j$ and $\hat{\sigma}_j$ as well as uncertain nonlinearities such as unmodeled dynamics. Now, consider Lyapunov function candidate

$$V(t) = \frac{1}{2}\left(s^2(t) + \sum_{j=1}^{N} \lambda_j^{-1}\tilde{\omega}_j^2 + \sum_{j=1}^{N} \tilde{\sigma}_j^{\mathrm{T}}\beta_j^{-1}\tilde{\sigma}_j\right), \tag{14}$$

where the estimation errors of the parameters are defined as

$$\tilde{\omega}_j \equiv \omega_j^* - \hat{\omega}_j, \quad \tilde{\sigma}_j \equiv \sigma_j^* - \hat{\sigma}_j. \tag{15}$$

Time derivative of V is given by

$$\begin{aligned}
\dot{V}(t) &= s(t)\dot{s}(t) - \sum_{j=1}^{N} \lambda_j^{-1}\tilde{\omega}_j\dot{\hat{\omega}}_j - \sum_{j=1}^{N} \dot{\hat{\sigma}}_j^{\mathrm{T}}\beta_j^{-1}\tilde{\sigma}_j \\
&= -k_{\mathrm{d}}s^2(t) + s(t)[\hat{f}(X) - f(X) - (a^* + \varepsilon_f)\,\mathrm{sgn}(s)] \\
&\quad + s(t)\sum_{j=1}^{N} \tilde{\omega}_j g_j(\hat{\sigma}_j) + s(t)\sum_{j=1}^{N} \kappa_j^{*\mathrm{T}}\tilde{\sigma}_j \\
&= -k_{\mathrm{d}}s^2(t) + s(t)\left[\sum_{j=1}^{N} \hat{\omega}_j g_j(\hat{\sigma}_j) - \sum_{j=1}^{N} \omega_j^* g_j(\sigma_j^*) - \varepsilon - (a^* + \varepsilon_f)\,\mathrm{sgn}(s) + \sum_{j=1}^{N} \tilde{\omega}_j g_j(\hat{\sigma}_j) + \sum_{j=1}^{N} \kappa_j^{*\mathrm{T}}\tilde{\sigma}_j\right] \\
&= -k_{\mathrm{d}}s^2(t) - \varepsilon s(t) - \varepsilon_f|s(t)| \\
&\quad + s(t)\left[\sum_{j=1}^{N} \hat{\omega}_j g_j(\hat{\sigma}_j) - \sum_{j=1}^{N} \omega_j^* g_j(\sigma_j^*) - a^*\,\mathrm{sgn}(s) + \sum_{j=1}^{N} \tilde{\omega}_j g_j(\hat{\sigma}_j) + \sum_{j=1}^{N} \kappa_j^{*\mathrm{T}}\tilde{\sigma}_j\right] \\
&= -k_{\mathrm{d}}s^2(t) - \varepsilon s(t) - \varepsilon_f|s(t)| + s(t)\left[\sum_{j=1}^{N} \omega_j^* g_j(\hat{\sigma}_j) - \sum_{j=1}^{N} \omega_j^* g_j(\sigma_j^*) - a^*\,\mathrm{sgn}(s) + \sum_{j=1}^{N} \kappa_j^{*\mathrm{T}}\tilde{\sigma}_j\right] \\
&= -k_{\mathrm{d}}s^2(t) - \varepsilon s(t) - \varepsilon_f|s(t)| - s(t)\left[\sum_{j=1}^{N} \omega_j^*(g_j(\sigma_j^*) - g_j(\hat{\sigma}_j)) + \sum_{j=1}^{N} \kappa_j^{*\mathrm{T}}(\hat{\sigma}_j - \sigma_j^*) + a^*\,\mathrm{sgn}(s)\right] \\
&= -k_{\mathrm{d}}s^2(t) - \varepsilon s(t) - \varepsilon_f|s(t)| - s(t)\left[\sum_{j=1}^{N} (\omega_j^*(g_j(\sigma_j^*) - g_j(\hat{\sigma}_j)) + \kappa_j^{*\mathrm{T}}(\hat{\sigma}_j - \sigma_j^*)) + a^*\,\mathrm{sgn}(s)\right].
\end{aligned} \tag{16}$$

We now consider two distinct cases, (a) $s(t)<0$ and (b) $s(t)>0$, and show that $\dot{V}(t)\leqslant 0$ with proper choices of $a^*\in R$ and $\kappa_j^*\in R^{n\times 1}$ in both cases.

(a) $s(t)<0$:

It follows that $\dot{V}(t)\leqslant 0$ if

$$a^* \geqslant \sum_{j=1}^{N} (\omega_j^*(g_j(\sigma_j^*) - g_j(\hat{\sigma}_j)) + \kappa_j^{*\mathrm{T}}(\hat{\sigma}_j - \sigma_j^*)). \tag{17}$$

Therefore, we can choose

$$a^* = \max_{\sigma\in\Theta_s} \sum_{j=1}^{N} (\omega_j^*(g_j(\sigma_j^*) - g_j(\hat{\sigma}_j)) + \kappa_j^{*\mathrm{T}}(\hat{\sigma}_j - \sigma_j^*)). \tag{18}$$

Since the form of the controller in Eq. (11) suggests that the quantity a^* is like a gain, we seek to find an κ_j^* so that a^* is minimized. Hence our goal is to choose κ_j^* as

$$a^* = \min_{k_j \in R^n} \max_{\sigma \in \Theta_s} \sum_{j=1}^{N} (\omega_j^*(g_j(\sigma_j^*) - g_j(\hat{\sigma}_j)) + \kappa_j^{*\mathrm{T}}(\hat{\sigma}_j - \sigma_j^*)). \tag{19}$$

At this stage, we need to perform a min–max optimization to find a^* and κ_j^* to complete the controller design. For the above optimization problem, several algorithms [20,21] have been proposed to obtain the values of a^* and κ_j^*. However, they generally involve a numerical search over the feasible set of solutions with complicated computing procedures. To simplify the problem, we notice that the nonlinear function $g_j(\cdot)$ is a convex and $-g_j(\cdot)$ is concave with respect to σ_j. In this case, the concave/convex optimization technique developed by Annaswamy et al. [15] can be utilized to obtain an analytic solution for a^* and κ_j^*. The detailed description of the concave/convex optimization technique is listed in the appendix. However, we should note that, the concave/convex optimization technique could not be applied directly. To determine whether the function $\omega_j^* g_j(\sigma_j)$ is convex or concave, the sign of ω_j^* must be known, because $g_j(\sigma_j)$ is convex on Θ, so $\omega_j^* g_j(\sigma_j)$ will be convex when $\omega_j^* \geqslant 0$, and concave when $\omega_j^* < 0$.

Since ω_j^* is the optimal weights in (6), it is reasonable to assume that the range of ω_j^* is known, i.e., $\omega_j^* \in [\omega_{\min}^*, \omega_{\max}^*]$. Now, we define a new parameter $p_j \in [p_{\min}, p_{\max}]$, the boundaries $p_{\min}$ and $p_{\max}$ are positive constants, which can be chosen by the designer. To deal with the problem of sign of ω_j^*, we introduce the following one-to-one mapping:

$$\omega_j^* = m + np_j, \quad p_j > 0, \tag{20}$$

where

$$m = \omega_{\min}^* - \frac{\omega_{\max}^* - \omega_{\min}^*}{p_{\max} - p_{\min}} p_{\min}, \tag{21}$$

$$n = \frac{\omega_{\max}^* - \omega_{\min}^*}{p_{\max} - p_{\min}} \quad (>0). \tag{22}$$

Substituting (20)–(22) into (19), we have

$$\begin{aligned} a^* &= \min_{k_j \in R^n} \max_{\sigma \in \Theta_s} \sum_{j=1}^{N} ((m + np_j)(g_j(\sigma_j^*) - g_j(\hat{\sigma}_j)) + \kappa_j^{*\mathrm{T}}(\hat{\sigma}_j - \sigma_j^*)) \\ &= \min_{k_j \in R^n} \max_{\sigma \in \Theta_s} \sum_{j=1}^{N} ((\omega_{\min}^* - np_{\min} + np_j)(g_j(\sigma_j^*) - g_j(\hat{\sigma}_j)) + \kappa_j^{*\mathrm{T}}(\hat{\sigma}_j - \sigma_j^*)) \\ &= a_1^* + a_2^*, \end{aligned} \tag{23}$$

where

$$a_1^* = \min_{k_{1j} \in R^n} \max_{\sigma \in \Theta_s} \sum_{j=1}^{N} (\omega_{\min}^*(g_j(\sigma_j^*) - g_j(\hat{\sigma}_j)) + \kappa_{1j}^{*\mathrm{T}}(\hat{\sigma}_j - \sigma_j^*)), \tag{24}$$

$$a_2^* = \min_{k_{2j} \in R^n} \max_{\sigma \in \Theta_s} \sum_{j=1}^{N} (n(p_j - p_{\min})(g_j(\sigma_j^*) - g_j(\hat{\sigma}_j)) + \kappa_{2j}^{*\mathrm{T}}(\hat{\sigma}_j - \sigma_j^*)). \tag{25}$$

Remark. The either convex or concave of $\omega^*_{\min} g_j(\sigma_j)$ and $n(p_j - p_{\min})g_j(\sigma_j)$ could be determined in both (24) and (25) via the one-to-one mapping, since the sign of $\omega^*_{\min}$ is known by the assumption and $n(p_j - p_{\min}) > 0$.

Now applying the concave/convex optimization results described in the appendix, we can get the solutions for $a^*_1, a^*_2, \kappa^*_{1j}$ and κ^*_{2j} that lead to $\dot{V}(t) \leqslant 0$.

$$a^*_1 = \begin{cases} \sum_{j=1}^{N} A_{j1} & \text{if } \omega^*_{\min} \geqslant 0, \\ 0 & \text{if } \omega^*_{\min} < 0, \end{cases} \tag{26}$$

$$\kappa^*_{1j} = \begin{cases} \omega^*_{\min} A_{j2} & \text{if } \omega^*_{\min} \geqslant 0, \\ \omega^*_{\min} \nabla g_{j\hat{\sigma}_j} & \text{if } \omega^*_{\min} < 0, \end{cases} \tag{27}$$

where $A_j = [A_{j1}, A_{j2}]^{\mathrm{T}} = G_j^{-1} b_j$, A_{j1} is a scalar, $A_{j2} \in R^n$,

$$G_j = \begin{bmatrix} -1 & \omega^*_{\min}(\hat{\sigma}_j - \sigma_{js1})^{\mathrm{T}} \\ -1 & \omega^*_{\min}(\hat{\sigma}_j - \sigma_{js2})^{\mathrm{T}} \\ \vdots & \vdots \\ -1 & \omega^*_{\min}(\hat{\sigma}_j - \sigma_{jsn+1})^{\mathrm{T}} \end{bmatrix}, \quad b_j = \begin{bmatrix} \omega^*_{\min}(\hat{g}_j - g_{js1}) \\ \omega^*_{\min}(\hat{g}_j - g_{js2}) \\ \vdots \\ \omega^*_{\min}(\hat{g}_j - g_{jsn+1}) \end{bmatrix}, \quad g_{jsi} = g_j(\sigma_{si}) \tag{28}$$

and

$$a^*_2 = \sum_{j=1}^{N} B_{j1}, \tag{29}$$

$$\kappa^*_{2j} = n(p_j - p_{\min}) B_{j2}, \tag{30}$$

where $B_j = [B_{j1}, B_{j2}]^{\mathrm{T}} = G_j^{-1} b_j$, B_{j1} is a scalar, $B_{j2} \in R^n$,

$$G_j = \begin{bmatrix} -1 & n(p_j - p_{\min})(\hat{\sigma}_j - \sigma_{js1})^{\mathrm{T}} \\ -1 & n(p_j - p_{\min})(\hat{\sigma}_j - \sigma_{js2})^{\mathrm{T}} \\ \vdots & \vdots \\ -1 & n(p_j - p_{\min})(\hat{\sigma}_j - \sigma_{jsn+1})^{\mathrm{T}} \end{bmatrix}, \quad b = \begin{bmatrix} n(p_j - p_{\min})(\hat{g}_j - g_{js1}) \\ n(p_j - p_{\min})(\hat{g}_j - g_{js2}) \\ \vdots \\ n(p_j - p_{\min})(\hat{g}_j - g_{jsn+1}) \end{bmatrix}. \tag{31}$$

(b) $s(t) > 0$:
Similar to the argument in (a), it follows that $\dot{V}(t) \leqslant 0$ if

$$\begin{aligned} a^* &= \min_{k_j \in R^n} \max_{\sigma \in \Theta_s} \sum_{j=1}^{N} [-(\omega^*_j (g_j(\sigma^*_j) - g_j(\hat{\sigma}_j)) + \kappa^{*\mathrm{T}}_{1j}(\hat{\sigma}_j - \sigma^*_j))] \\ &= a^*_1 + a^*_2, \end{aligned} \tag{32}$$

where

$$a_1^* = \min_{k_{1j}\in R^n} \max_{\sigma\in\Theta_s} \sum_{j=1}^{N} [-(\omega_{\min}^*(g_j(\sigma_j^*) - g_j(\hat{\sigma}_j)) + \kappa_{1j}^{*\mathrm{T}}(\hat{\sigma}_j - \sigma_j^*))], \tag{33}$$

$$a_2^* = \min_{k_{2j}\in R^n} \max_{\sigma\in\Theta_s} \sum_{j=1}^{N} [-(n(p_j - p_{\min})(g_j(\sigma_j^*) - g_j(\hat{\sigma}_j)) + \kappa_{2j}^{*\mathrm{T}}(\hat{\sigma}_j - \sigma_j^*))]. \tag{34}$$

The solutions of $a_1^*, a_2^*, \kappa_{1j}^*$ and κ_{2j}^* are given as follows:

$$a_1^* = \begin{cases} 0 & \text{if } \omega_{\min}^* \geqslant 0, \\ \sum_{j=1}^{N} A_{j1} & \text{if } \omega_{\min}^* < 0, \end{cases} \tag{35}$$

$$\kappa_{1j}^* = \begin{cases} -\omega_{\min}^* \nabla g_{j\hat{\sigma}_j} & \text{if } \omega_{\min}^* \geqslant 0, \\ -\omega_{\min}^* A_{j2} & \text{if } \omega_{\min}^* < 0, \end{cases} \tag{36}$$

where $A_j = [A_{j1}, A_{j2}]^{\mathrm{T}} = G_j^{-1} b_j$, A_{j1} is a scalar, $A_{j2} \in R^n$,

$$G_j = \begin{bmatrix} -1 & -\omega_{\min}^*(\hat{\sigma}_j - \sigma_{js1})^{\mathrm{T}} \\ -1 & -\omega_{\min}^*(\hat{\sigma}_j - \sigma_{js2})^{\mathrm{T}} \\ \vdots & \vdots \\ -1 & -\omega_{\min}^*(\hat{\sigma}_j - \sigma_{jsN+1})^{\mathrm{T}} \end{bmatrix}, \quad b_j = \begin{bmatrix} -\omega_{\min}^*(\hat{g}_j - g_{js1}) \\ -\omega_{\min}^*(\hat{g}_j - g_{js2}) \\ \vdots \\ -\omega_{\min}^*(\hat{g}_j - g_{jsn+1}) \end{bmatrix} \tag{37}$$

and

$$a_2^* = 0, \tag{38}$$

$$\kappa_{2j}^* = -n(p_j - p_{\min})\nabla g_{j\hat{\sigma}_j}. \tag{39}$$

The above shows that $\dot{V}(t) \leqslant 0$. Therefore, all signals in the system are bounded. Since $s(t)$ is uniformly bounded, it is easily shown that, if $\tilde{X}(0)$ is bounded, then $\tilde{X}(t)$ is also bounded for all t, and since $X_{\mathrm{d}}(t)$ is bounded by design, $X(t)$ is as well. To complete the proof and establish asymptotic convergence of the tracking error, it is necessary to show that $s(t) \to 0$ as $t \to \infty$. This can be accomplished by applying Barbalat's lemma to the continuous, nonnegative function:

$$V_1(t) = V(t) - \int_0^t (\dot{V}(\tau) + k_{\mathrm{d}} s^2(\tau))\,\mathrm{d}\tau \quad \text{with } \dot{V}_1(t) = -k_{\mathrm{d}} s^2(t). \tag{40}$$

It can easily be shown that every term in (10) is bounded, hence $\dot{s}(t)$ is bounded, which implies that $\dot{V}_1(t)$ is a uniformly continuous function of time. Since V_1 is bounded below by 0, and $\dot{V}_1(t) \leqslant 0$ for all t, use of Barbalat's lemma proves that $\dot{V}_1(t) \to 0$ and hence from (40) that $s(t) \to 0$ as $t \to \infty$. The stability of the closed-loop system described by (1), (11)–(13), (26)–(31) and (35)–(39) is thus established in the following theorem.

Theorem 1. *If the robust adaptive control laws* (11)–(13), (26)–(31) *and* (35)–(39) *are applied to the nonlinear plant* (1), *then all closed-loop signals are bounded and* $\tilde{X}(t)\to 0$ *as* $t\to\infty$.

Remarks. (1) Thanks to the concave/convex property of the nonlinear function $g_j(\cdot)$ for σ_j, it makes use of the optimization technique [15] possible. In this case, a globally stable adaptive system can be established. This is possible mainly due to the analytic solution for a^* and κ_j^* as a min–max optimization problem. The main feature of the new adaptive controller is that both parameters $\hat{\omega}_j$ that occur linear in the fuzzy approximator and parameters $\hat{\sigma}_j$ that occur nonlinearly can be tuned, simultaneously.

(2) Continuity with respect to switches of $g_j(\cdot)$ between concavity and convexity in (11)–(13) can easily be proved [15]. However, the control law (11)–(13) is discontinuous along the surface $s(t)=0$ due to the existence of sgn(•). It can lead to control chatter. Chattering is undesirable in practice because it involves high control activity and further may excite high frequency unmodelled dynamics. This can be remedied by approximating these discontinuous control laws by continuous ones inside a boundary layer [17]. To do this, sgn(•) in (11) is replaced by sat(•/ς), where ς is the boundary layer thickness. This leads to tracking within a guaranteed precision.

5. Simulation examples

To illustrate and clarify the proposed design procedure, we apply the adaptive fuzzy controller developed in Section 4 to control an unstable nonlinear system (Example 1) and a chaotic system (Example 2).

Example 1. A nonlinear system is described as [2,3]

$$\dot{x}(t)=\frac{1-\mathrm{e}^{-x(t)}}{1+\mathrm{e}^{-x(t)}}+u(t). \tag{41}$$

Without the control and disturbance, i.e., $u(t)=0$, it can be easily seen that the system is unstable, because of $\dot{x}(t)=(1-\mathrm{e}^{-x(t)})/(1+\mathrm{e}^{-x(t)})>0$ for $x(t)>0$, and $\dot{x}(t)=(1-\mathrm{e}^{-x(t)})/(1+\mathrm{e}^{-x(t)})<0$ for $x(t)<0$. The control objective is to force the system state $x(t)$ to the origin; i.e., $x_{\mathrm{d}}=0$. The simulation is conducted with the following linguistic descriptions:

R_f^k: *IF* x *is near* k, *THEN* f *is near* B_k

where *near* k, $k=-3,-2,-1,0,1,2,3$, is a fuzzy set with membership functions $\mu_k(x)=\exp(-(\sigma(x-k))^2)$, which are shown in Fig. 2 with initial variances $\sigma=1$. B_k are obtained by evaluating f at points $x=-3,-2,-1,0,1,2,3$. The values of B_k and σ are not required here since the exact W^* and σ^* are not required in the control law. However, with the knowledge about the FBF expansion it will be helpful for the choice of initial $\hat{W}(0)$ and $\hat{\sigma}(0)$ to speed up the adaptation process. In this example, those initial values $\hat{W}(0)$, and $\hat{\sigma}(0)$ are selected as $\hat{W}(0)=[-0.8,-0.6,-0.4,0,0.4,0.6,0.8]^{\mathrm{T}}$ and $\hat{\sigma}(0)=[1,1,1,1,1,1,1]^{\mathrm{T}}$.

Control law (11) was used with $k_{\mathrm{d}}=10$. The rates of the adaptation (12) and (13) are selected as $\lambda_j=0.1$ and $\beta_j=0.8$ where $j=1,2,\ldots,7$. Referring to the nonlinearity f being uniformly upper bounded, namely $(1-\mathrm{e}^{-x(t)})/(1+\mathrm{e}^{-x(t)})\leqslant 1$, the upper bound ε_f in (11) is thus selected as $\varepsilon_f=1$. To avoid the control chatter, as mentioned in Remark (2) in Section 4, we adopted the saturation function *sat*$(\cdot)$ with a boundary layer $\varsigma=0.05$ instead of the sign function sgn$(\cdot)$. The initial state and sampling time are chosen as $x(0)=2$ and $t=0.005$.

Since the parameter σ in each Guassian function is a scalar, the hypercube, Θ and the simplex, Θ_s coincide. In this case, we choose $\Theta=\Theta_s=[\sigma_{\min},\sigma_{\max}]$. For this simulation example, $\sigma_{\min}$ are chosen as

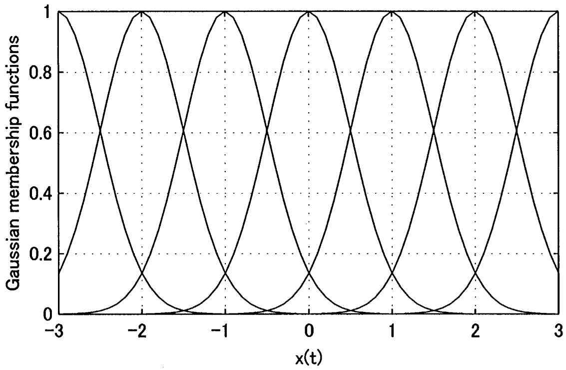

Fig. 2. Initial membership functions in antecedent.

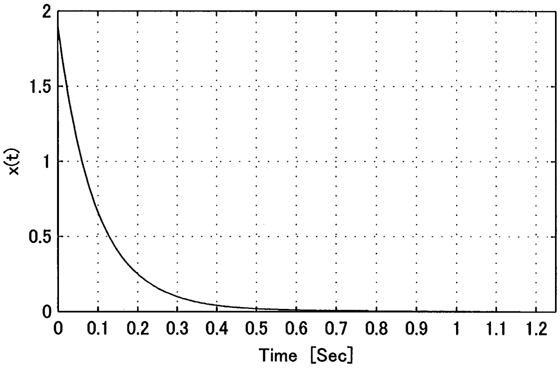

Fig. 3. Closed-loop state $x(t)$ using the controller law (11)–(13).

$\sigma_{\min} = [-0.9, -0.7, -0.5, -0.1, 0.3, 0.5, 0.7]^T$, and $\sigma_{\max}$ are chosen as $\sigma_{\max} = [-0.7, -0.5, -0.3, 0.1, 0.5, 0.7, 0.9]^T$.

Simulation results are shown in Figs. 3–5. Fig. 3 shows the evolution $x(t)$. A dramatic improvement on the system performance is observed, if compared with the result in [2]. Therefore, tuning all the parameters in FBF expansion clearly results in a superior tracking performance. The amount of control effort required to achieve the above level of performance is illustrated in Fig. 4, which confirms the smoothness of the control signal. The final tuned membership functions in antecedent are shown in Fig. 5. Comparing the initial membership function in Fig. 2, it is obvious that the variances in FBF expansion have been changed. Since the state x has an initial value $x(0) = 2$ (>0) and speedily converged to the origin $x_d = 0$ without any overshoots, the variances in positive region are tuned and the others in negative region are not tuned.

To show the robustness of the control scheme, an unknown disturbance or unmodeled dynamics $d(t)$ is added to the system model. The corresponding system becomes

$$\dot{x}(t) = \frac{1 - e^{-x(t)}}{1 + e^{-x(t)}} + d(t) + u(t) \quad \text{with } d(t) = \begin{cases} 0.5 & 0.4 \leqslant t \leqslant 0.6, \\ -0.5 & 0.7 \leqslant t \leqslant 0.9, \\ 0 & \text{otherwise.} \end{cases} \tag{42}$$

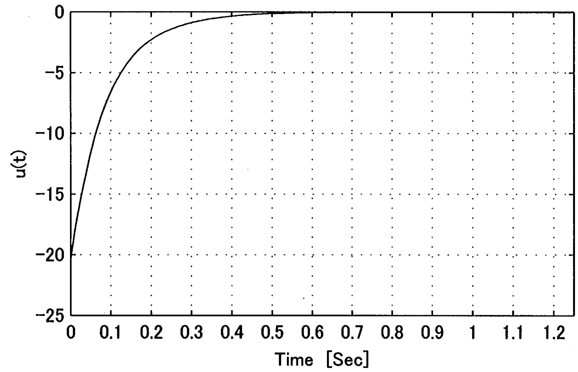

Fig. 4. Control signal $u(t)$ using the controller law (11)–(13).

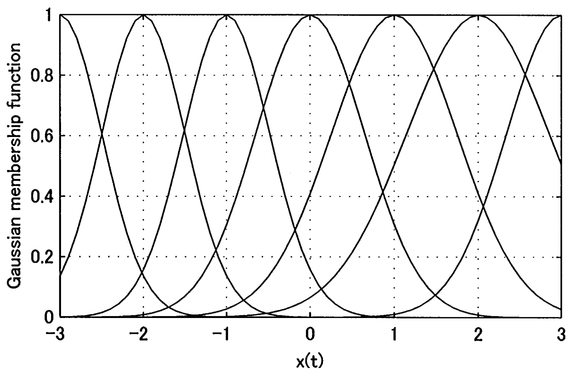

Fig. 5. Final membership function in antecedent.

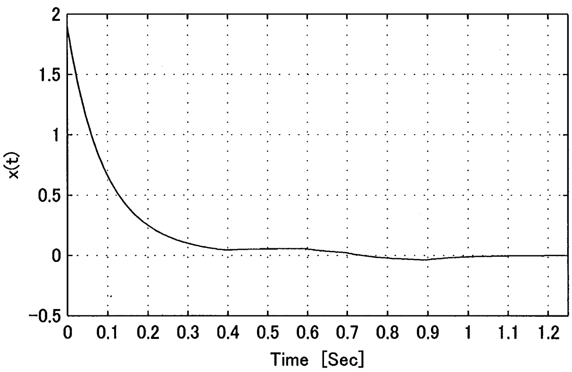

Fig. 6. Closed-loop state $x(t)$ using the controller law (11)–(13) with disturbance $d(t)$.

Without the change of the controller parameters and fuzzy descriptions described above, simulation results are shown in Figs. 6 and 7. It is obvious that the proposed control scheme has strong robustness over the unknown disturbance or unmodeled dynamics.

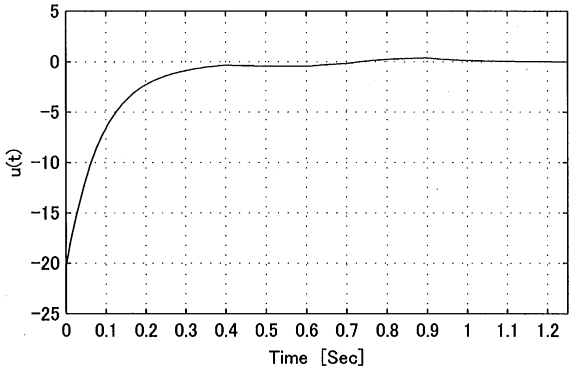

Fig. 7. Control signal $u(t)$ using the controller law (11)–(13) with disturbance $d(t)$.

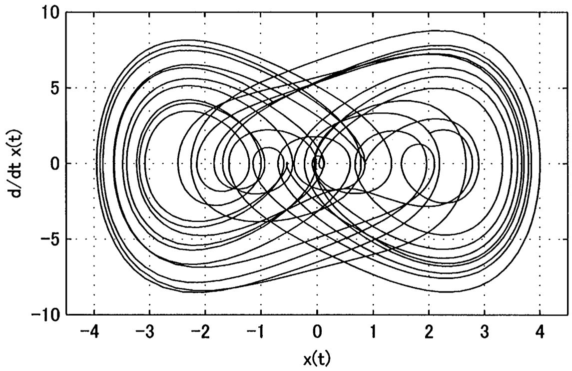

Fig. 8. Unforced trajectories $(x, \dot{x})$.

Example 2. In this example, we consider the Duffing forced-oscillation system:

$$\ddot{x}(t) = -a\dot{x}(t) - bx^3(t) + c\cos(t) + u, \tag{43}$$

which is chaotic in unforced case, i.e., $u = 0$. The unforced trajectory of the system is shown in Fig. 8 in phase plane $(x, \dot{x})$ for $x(0) = \dot{x}(0) = 2$, $a = 0.11$, $b = 1$, $c = 12$ and time period $t_0 = 0$ to $t_f = 60$. Now, we use the control algorithm proposed in the previous section to force the state $x(t)$ to follow a desired trajectory $x_d(t) = \sin(t)$. In the phase plane, the desired trajectory is a unit circle: $x_d(t) + \dot{x}_d(t) = 1$. In this simulation, we choose the initial membership functions as shown in Fig. 2 for both $x(t)$ and $\dot{x}(t)$. Clearly, the two input variables can set up $7 \times 7 = 49$ fuzzy rules as follows:

R_j: *IF* x *is* PB_j^1 and $\dot{x}$ *is* NZ_j^2, *THEN* f *is* w_j.

To verify the control scheme, suppose that we have no knowledge regarding the function $f = -a\dot{x}(t) - bx^3(t) + c\cos(t)$, so the initial consequent variables, for example, w_j, are selected randomly. To find a^* and κ_j^*, the construction of a simples Θ_s is needed. Since in each fuzzy rule there are two variances in antecedent that have to be tuned by adaptive law (13), the simplex is chosen by Fig. 9.

As mentioned before, because we do not have any knowledge about function f, it is impossible to give a suitable range, $[\sigma_{ji\,\min}, \sigma_{ji\,\max}]$, of each variance σ_{ji} on specifics. Hence, the only thing we can do is to

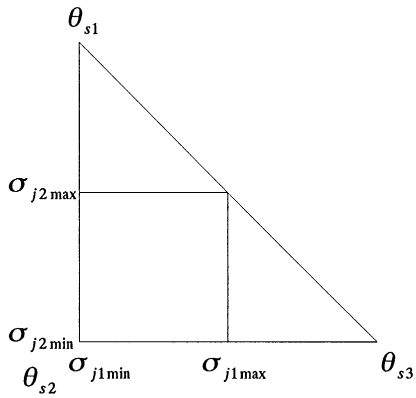

Fig. 9. Simplex Θ_s.

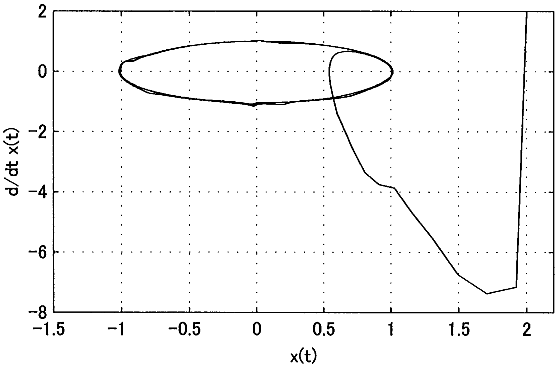

Fig. 10. Closed-loop trajectories $(x, \dot{x})$ using the controller law (11)–(13).

choose a large range $[\sigma_{\min}, \sigma_{\max}]$, and suppose that this is a common range for every σ_{ji}. In this simulation, we take $\sigma_{\min} = 0.01$ and $\sigma_{\max} = 2$. The closed-loop trajectory is shown in Fig. 10, where the initial conditions are $x(0) = \dot{x}(0) = 2$. We see that our control scheme can control the system with some unknown time-variable facts such as $\cos(t)$ to track the desired trajectory well.

6. Conclusion

In this paper a new adaptive fuzzy control law is presented. The main feature of the proposed method lies in the fact that the parameters, which appear nonlinearly in the fuzzy approximator, can stably be tuned. In this case the approximation error can be reduced and control performance can, therefore, be improved. The developed controller guarantees the global stability of the resulting closed-loop system. Simulation results for different nonlinear systems confirmed the theoretic analysis.

Appendix A solution for min–max problem [15]

Definition. A function $f(\theta)$ is said to be (i) convex on Θ if it satisfies the following inequality:

$$f(\lambda\theta_1 + (1-\lambda)\theta_2) \leqslant \lambda f(\theta_1) + (1-\lambda) f(\theta_2), \quad \forall \theta_1, \theta_2 \in \Theta \tag{A.1}$$

and (ii) concave if it satisfies the inequality

$$f(\lambda\theta_1 + (1-\lambda)\theta_2) \geqslant \lambda f(\theta_1) + (1-\lambda) f(\theta_2), \quad \forall \theta_1, \theta_2 \in \Theta, \tag{A.2}$$

where $\forall \lambda \in [0,1]$.

Let us consider a scalar function $f(\phi(t),\theta)$, which is continuous and bounded with respect to its arguments. θ is a parameter vector and belongs to a known hypercube $\Theta \subset R^n, \phi(t) \in R^n$ is a known bounded function of X, and for any $\phi(t)$, $f(\phi(t),\theta)$ is either convex or concave on Θ_s, where Θ_s is a simplex in R^n such that $\Theta_s \supset \Theta$. Suppose that vertices of Θ_s are denoted as θ_{si} $(i=1,2,\ldots,n)$. Then Θ_s may be expressed as

$$\Theta_s = \left\{ \theta_s \,\middle|\, \theta_s = \sum_{i=1}^{n+1} \alpha_i \theta_{si} \right\}, \quad 0 \leqslant \alpha_i \leqslant 1, \tag{A.3}$$

$$\sum_{i=1}^{n+1} \alpha_i = 1. \tag{A.4}$$

Lemma. *Let*

$$J(\omega,\theta) = f(\phi,\theta) - f(\phi,\hat{\theta}) + \kappa^{\mathrm{T}}(\hat{\theta} - \theta), \tag{A.5}$$

$$a^* = \min_{\omega \in R^n} \max_{\theta \in \Theta_s} \beta J(\omega,\theta), \tag{A.6}$$

$$\kappa^* = \arg\min_{\omega \in R^n} \max_{\theta \in \Theta_s} \beta J(\omega,\theta), \tag{A.7}$$

where $\hat{\theta} \in \Theta_s$ and β is a known non-zero constant. Then

$$a^* = \begin{cases} A_1 & \text{if } \beta f \text{ is convex on } \Theta_s, \\ 0 & \text{if } \beta f \text{ is concave on } \Theta_s, \end{cases} \tag{A.8}$$

$$\kappa^* = \begin{cases} A_2 & \text{if } \beta f \text{ is convex on } \Theta_s, \\ \nabla f_\theta & \text{if } \beta f \text{ is concave on } \Theta_s, \end{cases} \tag{A.9}$$

where $\nabla f_\theta = \partial f/\partial\theta$, $A = [A_1, A_2]^{\mathrm{T}} = G^{-1}b$, A_1 *is a scalar*, $A_2 \in R^m$,

$$G = \begin{bmatrix} -1 & \beta(\hat{\theta} - \theta_{s1})^{\mathrm{T}} \\ -1 & \beta(\hat{\theta} - \theta_{s2})^{\mathrm{T}} \\ \vdots & \vdots \\ -1 & \beta(\hat{\theta} - \theta_{sm+1})^{\mathrm{T}} \end{bmatrix}, \quad b = \begin{bmatrix} \beta(\hat{f} - f_{s1}) \\ \beta(\hat{f} - f_{s2}) \\ \vdots \\ \beta(\hat{f} - f_{sm+1}) \end{bmatrix} \tag{A.10}$$

and $f_{si} = f(\phi, \theta_{si})$.

References

[1] C.C. Lee, Fuzzy logic in control systems: fuzzy logic controller, part I and part II, IEEE Trans. Systems Man Cybernet. 20 (1990) 404–435.
[2] L.-X. Wang, Stable adaptive fuzzy control of nonlinear system, IEEE Trans. Fuzzy Systems 1 (1993) 146–155.
[3] C.-Y. Su, Y. Stepanenko, Adaptive control of a class of nonlinear systems with fuzzy logic, IEEE Trans. Fuzzy Systems 2 (1994) 258–294.
[4] J.T. Spooner, K.M. Passino, Stable adaptive control using fuzzy systems and neural networks, IEEE Trans. Fuzzy Systems 4 (1996) 339–359.
[5] B.S. Chen, C.H. Lee, Y.C. Chang, H_∞ tracking design of uncertain nonlinear SISO systems: adaptive fuzzy approach, IEEE Trans. Fuzzy Systems 4 (1996) 32–43.
[6] H. Lee, M. Tomizuka, Robust adaptive control using a universal approximator for SISO nonlinear systems, Technical Report, UC Berkeley, 1998.
[7] L.-X. Wang, J.M. Mendel, Fuzzy basis functions, universal approximation, and orthogonal least-squares learning, IEEE Trans. Neural Networks 3 (1992) 807–814.
[8] X.-J. Zeng, M.G. Singh, Approximation accuracy analysis of fuzzy systems as function approximators, IEEE Trans. Fuzzy Systems 4 (1996) 44–63.
[9] T. Poggio, F. Girosi, Networks for approximation and learning, Proc. IEEE 78 (1990) 1479–1481.
[10] B. Liu, J. Si, The best approximation to C^2 function and its error bounds using regular-center Gaussian networks, IEEE Trans. Neural Networks 5 (1994) 847–848.
[11] L.X. Wang, J.M. Mendel, Back-propagation fuzzy system as nonlinear dynamic system identifiers, Proc. IEEE Conf. on Fuzzy Systems, San Diego, 1992, pp. 1409–1418.
[12] X. Ye, N.K. Loh, Dynamic system identification using recurrent radial basis function network, Proc. American Control Conf., 1993, pp. 2912–2916.
[13] E. Kim, M. Park, S. Ji, M. Park, A new approach to fuzzy modelling, IEEE Trans. Fuzzy Systems 5 (1997) 328–337.
[14] T. Chai, S. Tong, Fuzzy direct adaptive control for a class of nonlinear systems, Fuzzy Sets and Systems 103 (1999) 379–384.
[15] A.M. Annaswamy, F.P. Skantze, A.-P. Loh, Adaptive control of continuous time systems with convex/concave parameterization, Automatica 34 (1998) 33–49.
[16] F. Girosi, G. Anzelloti, Rates of convergence of a approximation by translates, Artificial Intelligence Lab. Memo. No. 1288, MIT, Cambridge, MA, February 1992.
[17] J.-J.E. Slotine, J.A. Coetsee, Adaptive sliding controller synthesis for nonlinear systems, Int. J. Control 43 (1986) 1631–1651.
[18] C.-J. Chien, L.-C. Fu, Adaptive control of interconnected systems using neural network design, Proc. 37th IEEE Conf. on Decision and Control, Tampa, FL, 1998.
[19] W. Rudin, Principles of Mathematical Analysis, McGraw-Hill, New York, 1976.
[20] K. Shimizu, K. Ito, Design of neural stabilizing controller for nonlinear systems via Lyapunov's direct method, Trans. Soc. Instr. Control Engrs. 35 (4) (1999) 489–495.
[21] K. Shimizu, E. Aiyoshi, Mathematical Programming, Shokodo, Tokyo, 1984.
[22] S. Sastry, M. Bodson, Adaptive Control, Prentice-Hall, Englewood Cliffs, NJ, 1989.
[23] H. Han, C.-Y. Su, Further results on adaptive control of a class of nonlinear systems with fuzzy logic, Proc. 8th IEEE Internat. Conf. on Fuzzy Systems, Seoul, 1999.