FUZZY LOGIC CONTROL OF A CLASS OF FEEDBACK LINEARIZABLE NONLINEAR DYNAMICAL SYSTEMS

M.W. Vandegrift[1], F.L. Lewis[1], S. Zhu[2], and K. Liu[1]

[1]Automation and Robotics Research Institute,The University of Texas at Arlington, 7300 Jack Newell Blvd S, Fort Worth,TX 76118
[2]Center for Scientific Computation, North Carolina State University, Raleigh, NC 27695-8205

Abstract

The object of this paper is to achieve tracking control of a class of unknown nonlinear dynamical systems using a fuzzy logic controller. We present a repeatable design algorithm and a stability proof for an adaptive fuzzy logic controller that uses basis vectors based on the fuzzy system, unlike most adaptive approaches which use basis vectors depending on the unknown plant (e.g. a tediously computed "regression matrix"). We select an e-modification sort of approach to adapt the fuzzy system parameters. With mild assumptions on the class of nonlinear systems, using this adaptive fuzzy logic controller we prove uniform ultimate boundedness of the closed-loop signals and that the controller achieves tracking. In fact, the fuzzy system designed is a model-free universal fuzzy controller that works for any system in the given class of systems.

1 Introduction

In recent years, fuzzy logic control has become popular in both the literary community and industry. The challenge of controlling complex ill-defined processes without model-based knowledge of their underlying dynamics has motivated many researchers to the application of fuzzy set theory [18]. However, problems with high dimensionality occur due to our intuitive limitations and to mathematical complications associated with higher-dimensional fuzzy systems. More systematic and guaranteed methods for controller design and synthesis are required, such as the subsequently proposed adaptive fuzzy-logic controller (FLC). In order to conceive this novel adaptive FLC, two areas of recent work in fuzzy systems are used: (1) an n-dimensional mathematical framework, and (2) the theory of function approximation using FLC.

Considerable work has been done in setting a mathematical framework for n-dimensional fuzzy systems. Buckley [1], and Ying [16, 17] have laid mathematical foundations for FLCs, the latter constructing a FAM function that consists of a proportional-plus-integral part plus a nonlinear offset part. To remedy the problem of extending FLC design techniques to dimensions higher than two, [7] proposes a repeatable FLC design algorithm using n-D membership vectors. There, a rigorous mathematical expression was given for the reasoning surface manufactured by a fuzzy associative memory (FAM), and it was shown that the rulebase must satisfy some important restrictions in any closed-loop control application.

Recently, many researchers have explored the approximation properties of fuzzy systems [9, 13, 16, 19], often using the Stone-Weierstrass theorem. The techniques in [16] show that FAM functions are universal approximators for certain classes of functions. While [19] gives fuzzy system approximation results that do not depend on the Stone Weierstrass theorem, the results are only given for scalar (single-input/single-output) systems.

Some rigorous stability analysis has been done for FLCs. Langari and Tomizuka [5] provided a Lyapunov stability analysis for a general sort of membership function. Chiu and Chand [3] offered a cell-by-cell Lyapunov stability analysis for an aircraft FLC. Chen and Ying [2] have related FLCs to classical PID controllers showing stability using the small gain theorem. Otherwise, most fuzzy logic applications in control have been ad hoc, with no stability proofs or repeatable design algorithms given.

In adaptive control, one must perform tedious preliminary analysis for each prescribed system to determine a so-called "regression matrix" that is needed for the controller design. This regression matrix is effectively a "set of basis vectors" for that specific system under a "linear-in-the-parameters" assumption. By contrast, in the sequel, we use the work of [7, 9] to design an adaptive FLC that uses basis vectors based on the fuzzy system. We propose a design algorithm and derive the fuzzy controller structure. With mild assumptions on a general class of nonlinear systems, using this adaptive FLC we prove that the closed-loop signals are uniformly ultimately bounded and that the controller achieves tracking. The result is a model-free universal fuzzy logic controller that works for any system in a general class of nonlinear systems. The universal controller property of FLC accounts for their success in the literature despite the lack of formal design algorithms or formal stability proofs.

2 Background and Problem statement

For a general class of nonlinear systems defined below, we wish to design a fuzzy logic (FL) controller so that the output follows a prescribed trajectory with bounded error. Some system theory notions are given in this section.

2.1 A Class of State-Feedback Linearizable Nonlinear Systems

We investigate the class of single-input single-output (SISO) state-feedback linearizable systems having a state-space representation in the controllability canonical form

$$\begin{aligned} \dot{x}^1 &= x^2 \\ \dot{x}^2 &= x^3 \\ &\vdots \\ \dot{x}^n &= f(\mathbf{x}) + u + d \\ y &= x^1 \end{aligned} \tag{2.1}$$

with state $\mathbf{x} = [x^1\ x^2 \dots x^n]^T \in \Re^n$, output $y(t) \in \Re$, and control $u(t) \in \Re$. We shall assume that the unknown disturbance $d(t)$ has a constant known upper bound so that $|d(t)| < b_d$, and that $f:\Re^n \rightarrow \Re$ is a smooth unknown function. This is sufficient for the existence and uniqueness of solutions for (2.1). The upcoming development can be directly extended to "square" multi-input/multi-output systems where $x^i \in \Re^q$.

Note that while adaptive control needs an additional linear-in-the-parameters assumption on $f(\mathbf{x})$ for the controller design problem, this assumption is not required for the method used here. That is, it is not necessary to find a "regression matrix" by preliminary analysis of the system.

2.2 Tracking Problem

The primary goal of this note is to track a desired output $y_d(t)$ while keeping the states and control bounded. That is, the output error $y(t)-y_d(t)$ should be small. A feedback linearization approach will be used to achieve acceptable tracking accuracy. Thus, an adaptive FL controller will be designed that effectively feedback linearizes (2.1).

To this end we will make some mild assumptions that are widely used and hold in any practical design. First define a vector

$$\mathbf{x}_d(t) \equiv [y_d\ \dot{y}_d\ \dots\ y_d^{(n-1)}]^T \tag{2.2}$$

where the superscript in parenthesis indicates the order of the operator $\dfrac{d}{dt}$.

Assumption 2.1: The desired trajectory vector $\mathbf{x}_d$ is assumed to be continuous, known, and $\|\mathbf{x}_d(t)\| \le Q$ with Q a known bound. ■

Define a state error vector as

$$\mathbf{e} = \mathbf{x} - \mathbf{x}_d \tag{2.3}$$

and a filtered error, in standard use in robotics, as

$$r = \Lambda^T \mathbf{e}, \tag{2.4}$$

where $\Lambda = [\overline{\Lambda}\ 1] = [\lambda_1\ \lambda_2\ \dots\ \lambda_{n-1} 1]^T$ is an appropriately chosen coefficient vector so that $e(t) \rightarrow 0$ exponentially as $r(t) \rightarrow 0$ (i.e. $s^{n-1} + \lambda_{n-1} s^{n-2} + \dots + \lambda_1$ is Hurwitz). Then the dynamics (2.1) can be written in terms of the filtered error as

$$\dot{r} = f(\mathbf{x}) + u + d + Y_d \tag{2.5}$$

where

$$Y_d \equiv -y_d^{(n)} + \sum_{i=1}^{n-1} \lambda_i e_{i+1} \tag{2.6}$$

is a known signal. Note that $e_{i+1} \equiv y^{(i)}(t) - y_d^{(i)}(t)$ for $i = 1, 2, \dots, n-1$. It is assumed that all states are available as measurements.

We will design a controller $u(t)$ using a subsequently defined fuzzy system with adaptive law to keep $r(t)$ bounded and small. If we can show that (2.5) is a stable system, then (2.4) implies that $\mathbf{e}(t)$ remains bounded so that the tracking objective is achieved.

Fact 2.1: For each time t, $\mathbf{x}(t)$ is bounded by

$$\|\mathbf{x}\| \le Q + \|\Lambda^T\|^{-1}\|r\| = Q + C_1\|r\|. \tag{2.7}$$

■

2.3 Stable Systems

In order to study the stability of the forthcoming FLC we use some basic mathematical concepts and results; in addition, an important stability notion is defined. For instance, the next concept of a Lipschitz continuous function is essential for closed-loop control applications.

Definition 2.1: *Lipschitz Continuous*. Let $f(.)$ be defined such that $f(\mathbf{x})$: $\Omega \rightarrow \Re$ where Ω is a compact subset of $\Re^n$. We say $f(.)$ is Lipschitz if there exists an $L > 0$ such that

$$|f(\mathbf{x}_2) - f(\mathbf{x}_1)| \le L|\mathbf{x}_2 - \mathbf{x}_1|, \text{ for every } \mathbf{x}_1, \mathbf{x}_2 \in \Omega. \quad ■$$

Let $u = G(\mathbf{x})$ be a feedback controller in (2.1). If $f(\mathbf{x})$ and $G(\mathbf{x})$ are both Lipschitz, then there exist unique solutions $\mathbf{x}(t)$ in closed-loop. Therefore, any proposed controller $G(\mathbf{x})$ must be at least Lipschitz (of course it must also guarantee stability.)

Definition 2.2: We say the solution of (2.1) is *uniformly ultimately bounded* (UUB) if there exists a compact set $U \subset \Re^n$ such that for all $\mathbf{x}(t_0) = \mathbf{x}_0 \in U$ there exists an $\varepsilon > 0$ and a number $T(\varepsilon, \mathbf{x}_0)$ such that $\|\mathbf{x}(t)\| < \varepsilon$ for all $t \ge t_0 + T$. ■

As we shall see in the proof of Theorem 4.1, the compact set U is related to the compact set on which the fuzzy system approximation property (3.23) holds.

3 N-Dimensional Fuzzy Associative Memories

This section introduces the fuzzy system structure that will be used in our controller. We plan to show that there are important restrictions on the rules allowed in any fuzzy system used for closed-loop control purposes (Theorem 3.1). Moreover, a major role in controls design is played by the universal approximation result in Theorem 3.2. Using the results in [7], the following provides a streamlined mathematical framework for n-dimensional fuzzy associative memories. It is necessary to discuss the five main components of a FAM: The state membership functions (MFs), the control representative values, the rulebase, the inferencing for the rule antecedents, and the defuzzification method used to determine a single crisp control action from multiple fuzzy numbers corresponding to active rules.

3.1 Vector Fuzzy Numbers and Membership Vectors

This subsection provides some basic definitions connected primarily with extending standard fuzzy notions to the case of a vector $\mathbf{x}$. The following [7] formalizes and extends some ideas in [1, 16, 17], etc. Let I denote the interval $[0,1]$ and $I^n := I \times I \times \dots \times I$ (n times).

Definition 3.1: *Vector Fuzzy Number*. Given an integer $n > 0$, define $\overline{n} := \{1\ 2 \cdots n\}$. Given $\mathbf{x} = [x^1 \dots x^n]^T \in \Re^n$ and integers N_k, $k \in \overline{n}$, define fuzzy numbers

$$X_i^k \equiv (x^k, \mu_i^k(x^k)), \quad i \in \overline{N}_k, \quad k \in \overline{n} \tag{3.1}$$

where the i-th membership function (MF) of the k-th state is $\mu_i^k : \Re \rightarrow I$. Define the Cartesian products

$$X = \{X_i^1\} \times \{X_i^2\} \times \dots \times \{X_i^n\}, \tag{3.2}$$

which is termed the library of fuzzy numbers, and

$$\mu(\mathbf{x}) = \{\mu_i^1(x^1)\} \times \{\mu_i^2(x^2)\} \times \cdots \times \{\mu_i^n(x^n)\}. \quad (3.3)$$

Given a set of natural numbers $i_1, i_2, \cdots i_n$ with $i_k \in \overline{N}_k$ a fuzzy number on $\Re^n$ is defined as

$$X_{i_1 i_2 \cdots i_n} \equiv (X_{i_1}^1, X_{i_2}^2, \ldots, X_{i_n}^n) \in X, \quad (3.4)$$

with n-dimensional membership function $\Re^n \to I^n$ defined by

$$\mu_{i_1 i_2 \cdots i_n}(\mathbf{x}) \equiv (\mu_{i_1}^1, \mu_{i_2}^2, \ldots, \mu_{i_n}^n) \in \mu(\mathbf{x}). \quad (3.5)$$

■

In the sequel, considerable profit is obtained by selecting MFs based on the triangle function. This, together with the choice of product inferencing [6], allows one to obtain rigorous uniform approximation and convergence results very simply [7, 9]. It is important to note that the triangular MFs include the trapezoid MFs and the pseudotrapezoid-shaped (PTS) functions in [19], since, by selecting the control representative values u_j in

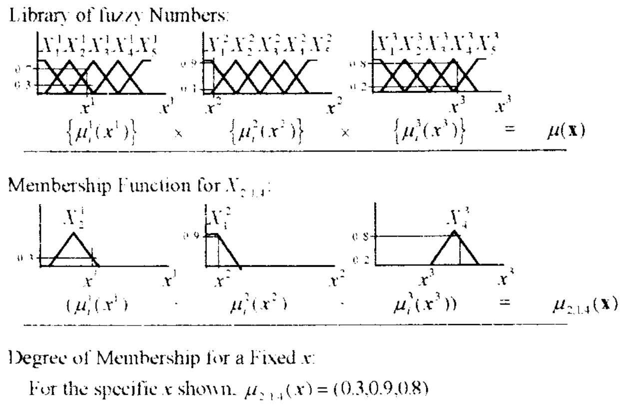

Figure 1: Sample triangle membership functions for n = 3, N_j = 5.

the rules (3.14) the same for two adjacent triangular MFs for the same component x^k, one has in fact equivalently used a single trapezoid MF.

Given, therefore, $y \in \Re$ and fixed real parameters $a \leq b \leq c$ define the (nonsymmetric) triangle function as

$$\Lambda(a,b,c)(y) = \begin{cases} \dfrac{y-a}{b-a}, & a \leq y \leq b \quad (= 1 \text{ if } a = -\infty) \\ \dfrac{c-y}{c-b}, & b \leq y \leq c \quad (= 1 \text{ if } c = \infty) \\ 0, & \textit{otherwise}. \end{cases} \quad (3.6)$$

Definition 3.2: *Triangular Membership Functions.* Given $\mathbf{x} = [x^1 \ldots x^n]^T \in \Re^n$, select compact sets $[x_{\min}^1, x_{\max}^1] \in \Re$, $[x_{\min}^2, x_{\max}^2] \in \Re, \ldots$ for each component x^k, and integers N_k, $k \in \overline{n}$. Respectively select strictly increasing partitions $\{x_i^1\}^{N_1}$, $\{x_i^2\}^{N_2}, \ldots$, that is

$$x_{\min}^1 = x_1^1 < x_2^1 < \ldots < x_{N_1}^1 = x_{\max}^1 \quad (3.7)$$

and so on. The Cartesian product of the compact sets $[x_{\min}^1, x_{\max}^1]$ through $[x_{\min}^n, x_{\max}^n]$ defines a compact hypercube Ω that will be useful in demonstrating the approximation properties of fuzzy systems. Select, for each component of $\mathbf{x}$, triangular membership functions of the form

$$\mu_1^k(x^k) = \Lambda(-\infty, x_1^k, x_2^k)(x^k)$$
$$\mu_i^k(x^k) = \Lambda(x_{i-1}^k, x_i^k, x_{i+1}^k)(x^k) \;, \qquad 1 < i < N_k \;. (3.8)$$
$$\mu_{N_k}^k(x^k) = \Lambda(x_{N_k-1}^k, x_{N_k}^k, \infty)(x^k)$$

The leftmost and rightmost MFs are selected so that all values of $x^k \in \Re$ correspond to at least one MF. ■

Note that these MFs are normal, complete, and consistent in the sense of [19]. The partitions corresponding to the components x^k of $\mathbf{x}$ are generally nonuniform. In addition, the MFs defined in this fashion satisfy the important property

$$\sum_{i=1}^{N_k} \mu_i^k(x^k) = 1 \;, \qquad \text{all } x^k \in \Re, \; k \in \overline{n}. \quad (3.9)$$

Sample N-D triangular MFs are shown in Figure 1.

Definition 3.3: *Degree of Membership Vector.* Given a fixed $\mathbf{x} = [x^1 \cdots x^n]^T \in \Re^n$, define the n-dimensional membership vector for $\mathbf{x}$ as $\mu(\mathbf{x})$ in (3.3) evaluated at $\mathbf{x}$. For a specified set of natural numbers $i_1, i_2, \cdots i_n$, $i_k \in \overline{N}_k$ the value $\mu_{i_1 i_2 \cdots i_n}(\mathbf{x})$ is the degree of membership of $\mathbf{x}$ in $X_{i_1 i_2 \cdots i_n}$. Note that for fixed $\mathbf{x}$, $\mu(x) \subset I^n$ and $\mu_{i_1 i_2 \cdots i_n}(x) \in I^n$. ■

Note that we define the 'degree of membership' of this logical expression as a member of I^n, so that we have not yet addressed the problem of 'fuzzy inferencing', i.e. combining the $\mu_{i_k}^k$ into a single element of I.

Remark: It is common in the literature to denote $\mathbf{x} = X_{i_1 i_2 \cdots i_n}$ by

$$(x^1 \text{ is } X_{i_1}^1) \text{ .and. } (x^2 \text{ is } X_{i_2}^2) \text{ .and. } \ldots \text{ .and. } (x^n \text{ is } X_{i_n}^n) \quad (3.10)$$

The fuzzy numbers X_i^k, are generally described in linguistic terms such as 'positive big', negative small', etc.

Definition 3.4: *Control Representative Values.* Given a scalar output or control variable $u \in \Re$, select an integer M and a finite set of M representative values u_j of u. The representative values may be viewed as the centroids of 'control MFs' having the form of impulse functions centered at the values u_j (i.e. 'singleton' MFs). ■

This definition avoids the usual quandary in FAM design where control MFs, generally triangular, are defined and one is then faced with the fact that the actual MFs are irrelevant; only their representative values are important. The representative values are generally assigned linguistic descriptors such as 'positive big', 'near zero', and so on.

3.2 FAM Rulebase

The upcoming development is associated with providing a rigorous expression for the function $u = g(\mathbf{x})$ manufactured by a FAM in terms of the selected MFs, rulebase, inferencing method, control representative values, and defuzzification scheme.

First define a class of rules relating the input fuzzy numbers X and the control representative values u_j.

Definition 3.5: *Fuzzy Associative Memory (FAM) Rulebase.* Given input fuzzy numbers X and control representative values $\{u_j\}$, define the rulebase $\mathcal{R}$ as a set of L rules where, for each rule $R_l \in \mathcal{R}$, $l \in \bar{L}$:

1. The left-hand side associates a member of X to the antecedent.
2. The right-hand side associates a value of u_j with the consequent. ■

The rules are thus of the form

$$\text{If } (\mathbf{x} = X_{i_1,i_2,\dots,i_n}) \text{ then } (u_j), \tag{3.11}$$

which is usually expressed in the literature as

$$\begin{aligned} &\text{If } [(x^1 \text{ is } X^1_{i_1}) \text{ .and. } (x^2 \text{ is } X^2_{i_2}) \text{ .and. } \dots \\ &\text{.and. } (x^n \text{ is } X^n_{i_n})] \text{ then } (u \text{ is } u_j) \end{aligned} \tag{3.12}$$

We denote this rule by

$$R_{i_1,i_2,\dots,i_n,j}. \tag{3.13}$$

It is clear that the rulebase can be considered as a relation on $\bar{N}_1 \times \bar{N}_2 \times \dots \times \bar{N}_n \times \bar{M}$.

It is convenient to denote the value u_j of u associated by the rulebase to $X_{i_1,i_2,\dots,i_n}$ by $u_{i_1,i_2,\dots,i_n}$ so that the rule becomes

$$\begin{aligned} &\text{If } [(x^1 \text{ is } X^1_{i_1}) \text{ .and. } (x^2 \text{ is } X^2_{i_2}) \text{ .and. } \dots \\ &\text{.and. } (x^n \text{ is } X^n_{i_n}) \text{ then } (u \text{ is } u_{i_1,i_2,\dots,i_n}) \end{aligned} \tag{3.14}$$

A more general class of rules is defined if the antecedent can be a logical function defined on X. However, this complicates things, and our framework can be extended to this more general case in a straightforward manner. Moreover, the definition suffices for a large class of control systems, namely, almost all FLC contained to date in the literature. It also suffices for the class of stabilizing FLC in this paper.

3.3 Inferencing and Defuzzification

The final components of the FAM are defined in this subsection. The inferencing function determines the method of inferencing (i.e. computing truth values of composite and/or expressions) when the antecedent side of the rules contains several fuzzy numbers X^k_i. The defuzzification function is used to determine crisp values of the control $u(t)$ when several rules dictate using different values of control.

With regard to the rule antecedents, let there be prescribed an element $X_{i_1,i_2,\dots,i_n}$ of X with MF $\mu_{i_1,i_2,\dots,i_n}(\mathbf{x})$. For fixed $\mathbf{x} \in \Re^n$ the degree of fulfillment of the statement $\mathbf{x} = X_{i_1,i_2,\dots,i_n}$ (i.e. the truth value of ($X_{i_1,i_2,\dots,i_n}$), see (3.10)) must be specified as a member of I. Computing this value as a function of the degree of membership $\mu_{i_1,i_2,\dots,i_n}(\mathbf{x}) \in I^n$, as per Definition 3.3, is termed inferencing, and is accomplished by defining a function from $I^n \to I$. In the rules (3.11), (3.14) the components $X^k_{i_k}$ of the antecedent are combined using the binary logical 'and' operator. To turn the map $X^k_{i_k} \to \mu^k_i(x^k)$ for a fixed x^k into a semigroup homomorphism, a binary operator $\otimes$ must be defined on the μ^k_i. We define this operation to be Larsen's product inferencing which uses the product

$$\mu^1_{i_1} \otimes \mu^2_{i_2} \otimes \dots \otimes \mu^n_{i_n} = \mu^1_{i_1}\mu^2_{i_2},\dots,\mu^n_{i_n}.$$

Definition 3.6: *Degree of Participation.* For the rule (3.14) and a fixed $\mathbf{x} \in \Re^n$, the degree of participation of the consequent is taken as

$$\pi_{i_1,i_2,\dots,i_n}(\mathbf{x}) = \prod_{k=1}^{n} \mu^k_{i_k}(x^k) \tag{3.15}$$

(assuming all rules have certainty factor of 1.0). ■

Definition 3.7: *Defuzzification and FAM Function*

Select the centroid defuzzification function, so that the FAM function manufactured by the fuzzy system is given by

$$u = g(\mathbf{x}) = \frac{\sum \pi_{i_1,i_2,\dots,i_n} u_{i_1,i_2,\dots,i_n}}{\sum \pi_{i_1,i_2,\dots,i_n}}, \tag{3.16}$$

where the sum is over all active rules given a specific $\mathbf{x} \in \Re^n$. ■

The FAM function is also often called the "reasoning surface."

3.4 System Theory Properties of the FAM Function

Given the machinery of sections 3.1-3.3, it is very easy to prove a string of theorems including the following. Recall that for existence and uniqueness of solutions to nonlinear differential equations, a *Lipschitz requirement* on the system functions suffices [7, 12]. This restricts the rules allowed for any FL system used for closed-loop control applications. In fact, the next result states that there should be 2^n active rules in each cell of the hypercube (Definition 3.2) for closed-loop control applications.

Theorem 3.1 (Lipschitz Requirement and Restriction on FL Rulebase): Let $x=[x^1\ x^2\ \dots\ x^n]^T \in \Re^n$ and the state MFs for x^k be triangular of the form given in Definition 3.2 with respective cardinalities N_k, and let the control representative values have cardinality M. Suppose that product inferencing and centroid defuzzification are used. Suppose that the rulebase provides a mapping from $\bar{N}_1 \times \bar{N}_2 \times \dots \times \bar{N}_n$ to $\bar{M}$, so that every fuzzy number $(X^1_{i_1}, X^2_{i_2}, \dots, X^n_{i_n})$ is mapped to a single $u_{i_1,i_2,\dots,i_n}$. Then:

1. The denominator of (3.16) evaluates to 1. That is,

$$\sum \pi_{i_1,i_2,\dots,i_n}(\mathbf{x}) = 1. \tag{3.17}$$

Then, the FAM function may be written as

$$g(\mathbf{x}) = \sum u_{i_1,i_2,\dots,i_n} \pi_{i_1,i_2,\dots,i_n}(\mathbf{x}). \tag{3.18}$$

2. For every compact set Ω, the FAM function $g(\mathbf{x}): \Omega \subset \Re^n \to \Re$ is a Lipschitz continuous function such that

$$g(x^1_{i_1}, x^2_{i_2}, \dots, x^n_{i_n}) = u_{i_1,i_2,\dots,i_n}. \tag{3.19}$$

Proof: 1. Use induction and proceed as in [9].

2. Use the fact that (3.18) is a polynomial of order n and proceed as in [9]. ■

It is easy to demonstrate that if the rulebase does not provide the required mapping, but only a relation, then the FAM function g(**x**) may not pass through the points in (3.19) and *may not be Lipschitz*. Since the feedback controller to be proposed in section 4 relies on $g(\mathbf{x})$, this could have negative ramifications on the closed-loop system properties. Unfortunately, in many FLC designs in the literature, attempts are made to simplify computations by 'skipping rules'; that is, by omitting rules for some of the state MFs. This violates the conditions of the theorem since the rulebase is not then a mapping but only a relation. Therefore, even *the existence and uniqueness of closed-loop solutions cannot be guaranteed if rules are omitted.* This problem relates closely to some detailed discussions in [7].

3.5 Approximation Properties of the FAM Function

Under the hypotheses of Theorem 3.1 the FAM function defined by (3.18) is a convex combination due to property (3.17). As shown in [9], one may use the convex combination property of (3.18) to prove rigorously the following approximation abilities of fuzzy systems.

Theorem 3.2 (Universal Approximation): Let Ω be a bounded hypercube in $\Re^n$. For any given continuous function $f(.): \Omega \rightarrow \Re$, and any positive number ε, there is a fuzzy system with FAM function $g(.)$ defined in (3.18) such that

$$|f(\mathbf{x}) - g(\mathbf{x})| \le \varepsilon, \ \forall \mathbf{x} \in \Omega. \qquad (3.20) \blacksquare$$

To design the approximating fuzzy system given the compact hypercube Ω of $\Re^n$ mentioned in the theorem, and $\mathbf{x} = [x^1\, x^2 \ldots x^n]^T \in \Re^n$, select integers N_k, $k \in \overline{n}$ and uniform partitions $\{x^k_{i_k} | i_k \in \overline{N_k}\}$ along x^k such that Ω is covered and partitioned into smaller hypercube cells. Define $\delta^k \equiv x^k_{i_k+1} - x^k_{i_k}$ as the partition interval in x^k, and $\delta = \left\|[\delta^1\, \delta^2 \ldots \delta^n]^T\right\|_2$. Based on this partition of Ω, define a fuzzy system as above, using triangular membership functions (Definition 3.2.)

The next result goes beyond results currently available in the literature by showing specifically how to select the partition norm δ (and hence the numbers of membership functions in each component x^k) to achieve a prescribed approximation accuracy.

Theorem 3.3 (Uniform Error Bound of Fuzzy System Approximation):

Let the conditions of Theorem 3.1 on the fuzzy system hold. Consider a class C_B of continuous functions $f(\mathbf{x}): \Omega \rightarrow \Re$, whose Jacobians are uniformly bounded by a positive real number B, that is $\|f'(\mathbf{x})\| \le B$ for all $\mathbf{x} \in \Omega$ and $f(\mathbf{x}) \in C_B$. Define $M = N_1 \cdot N_2 \cdot \ldots \cdot N_n$. Select the control representative values as the samples of $f(.)$ at the vertices of the hypercube cells,

$$u_{i_1, i_2, \ldots, i_n} = f(x^1_{i_1}, x^2_{i_2}, \ldots, x^n_{i_n}). \qquad (3.21)$$

Then, for any $f(\mathbf{x}) \in C_B$, $\mathbf{x} \in \Omega$ we have

$$|f(\mathbf{x}) - g(\mathbf{x})| \le B\delta \qquad (3.22)$$

with $g(\mathbf{x})$ the reasoning surface (3.18) of the fuzzy system and δ the size of the partition interval for Ω defined by the selected membership functions. ■

Therefore, given a desired approximation accuracy ε and bound B that describes the maximum variation of the class of functions to be approximated, the required partition norm of the fuzzy system is $\delta = \varepsilon / B$. This, and the extent of Ω, define the number of MF's required in each component of $\mathbf{x}$.

Hence, on any bounded hypercube Ω, a general continuous function $f(\mathbf{x})$ can be written as

$$f(\mathbf{x}) = U^T P(\mathbf{x}) + \varepsilon \qquad (3.23)$$

where

$$U^T P(\mathbf{x}) = \sum u_{i_1, i_2, \ldots, i_n} \pi_{i_1, i_2, \ldots, i_n}(\mathbf{x})$$

and U is the control representative value vector $P(\mathbf{x})$ is a vector of degrees of participation. The fuzzy system functional reconstruction error is ε.

The definition of U and $P(\mathbf{x})$ is not unique. Let the elements of U be denoted u_j and the elements of $P(\mathbf{x})$ be denoted as $p_j(x)$. Algorithm 3.1 gives one technique for defining U and $P(\mathbf{x})$ in terms of $u_{i_1, i_2, \ldots, i_n}$ and $\pi_{i_1, i_2, \ldots, i_n}$ respectively.

Algorithm 3.1:

Using left-hand odometer ordering, each member of the set of indices $\{i_1, i_2, \ldots, i_n\}$, $i_k \in \overline{N_k}$, can be assigned a unique integer $j \in \overline{M}$, $M = N_1 \cdot N_2 \cdot \ldots \cdot N_n$. That is, there exists an isometry from $\{i_1, i_2, \ldots, i_n\} \rightarrow j$. Let

$$\begin{aligned} P(\mathbf{x}) &= \begin{bmatrix} p_1 & p_2 & \cdots & p_j & \cdots & p_M \end{bmatrix}^T \\ &\equiv \begin{bmatrix} \pi_{1,1,\ldots,1} & \pi_{2,1,\ldots,1} & \cdots & \pi_{i_1, i_2, \ldots, i_n} & \cdots & \pi_{N_1, N_2, \ldots, N_n} \end{bmatrix}^T, \end{aligned} \qquad (3.24)$$

where the indices increase in left-hand odometer order. Then, j is given by

$$j = i_1 + \sum_{k=2}^{n} (i_k - 1) \prod_{l=1}^{k-1} N_l. \qquad (3.25)$$

Now use this association to define $p_j(\mathbf{x}) = \pi_{i_1, i_2, \ldots, i_n}(\mathbf{x})$ as given by (3.15). It should be noted that the components of U are also ordered according to this algorithm (see (3.18) and (3.21)). ■

Note that, for approximation, u_j are the samples of $f(.)$ at the vertices of the hypercube cell partition as shown in (3.21). In controls applications, the function $f(.)$, and therefore the required samples for approximation are unknown. However, as shown in the next section, it is only necessary to know that an approximating fuzzy system exists. Then, the adaptation rule given there identifies the unknown samples of $f(.)$. Since $f(\mathbf{x})$ is continuous there exists a number of quantized points (i.e. M) in the compact set Ω such that the approximation error ε can be made arbitrarily small. Note moreover, that ε is bounded on any compact set by a known bound. That is, $\|\varepsilon\| \le \varepsilon_N$, with ε_N known and depending on Ω, B, and δ.

Corollary 3.1 (*Known bound on U*): If $f(\mathbf{x}) \in C_B$ on the compact set Ω, then it is bounded on Ω. Therefore, though the control representative value vector U is unknown in applications, $\|U\| \le U_{\max}$ for some known bound U_{max}. ■

4 Universal Adaptive Fuzzy Logic Controller (AFLC)

The structure and design methodology for the AFLC are now given. The final form of the fuzzy controller to be defined is shown in Figure 2. The FLC has two control loops, an outer proportional-plus-derivative (PD) tracking loop plus an extra term generated by a fuzzy logic system in an inner control loop.

The unknown system (2.1) is now approximately feedback linearized by choosing the control action

$$u = -\hat{f}(\mathbf{x}) + v \tag{4.1}$$

where the estimate of $f(.)$ is given by a fuzzy system as

$$\hat{f}(\mathbf{x}) = \hat{U}^T P(\mathbf{x}) \tag{4.2}$$

and the auxiliary control signal is

$$v = -K_v r - Y_d \tag{4.3}$$

Here $\hat{f}(\mathbf{x})$ will be constructed by using the fuzzy system to form an appropriate set of basis functions, namely, the participation vector $P(\mathbf{x})$ which is constructed from the selected membership functions using Algorithm 3.1. Then, $\hat{U}$, the current estimate of the control representative values U, is provided by the adaptive tuning law to be subsequently given.

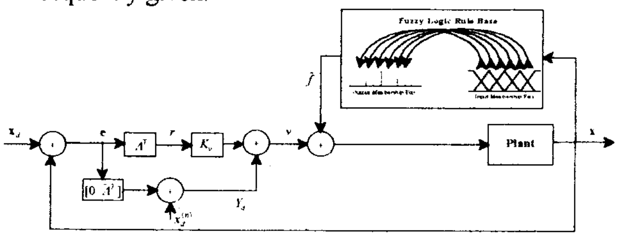

Figure 2: Adaptive Fuzzy Logic Universal Control Structure.

Note that with most adaptive control approaches, one must find a regression matrix by tedious preliminary analysis by analysis of the system (2.1) using a linear-in-the-parameters assumption. That is, the regression matrix provides a set of basis vectors that depends on the unknown plant. By contrast, in FL control, the participation vector $P(\mathbf{x})$ depends on the fuzzy system design, and is good for any plant in the class (2.1), regardless of the specific form of $f(.)$ as long as it is smooth. That is, no regression matrix, as needed in adaptive control, need be found. This has the effect of making Figure 2 a universal fuzzy controller for any system in class (2.1). The universal controller property of FLC accounts for their success in the literature despite the lack of formal design algorithms or formal stability proofs.

By our discussion in the previous section we know that the ideal value for U exists so that (3.23) holds and it is bounded so that

$$\|U\| \le U_{\max} \tag{4.4}$$

for some known $U_{\max}$. Moreover, $\|\varepsilon\| < \varepsilon_N$ a known approximation error bound. Define the control representative value estimation error as

$$\tilde{U} = U - \hat{U}. \tag{4.5}$$

Combining (4.1)-(4.3), we have the control input

$$\begin{aligned} u &= -\hat{f}(\mathbf{x}) - K_v r - Y_d \\ &= -\hat{U}^T P(\mathbf{x}) - K_v r - Y_d \end{aligned} \tag{4.6}$$

where $P(\mathbf{x})$ is given by (3.24), and (3.16), under the ordering (3.25). Then, substituting (4.6) into (2.5) the closed-loop filtered error dynamics become

$$\begin{aligned} \dot{r} &= U^T P(\mathbf{x}) + \varepsilon - \hat{U}^T P(\mathbf{x}) - K_v r - Y_d + d + Y_d \\ &= -K_v r + \tilde{U}^T P(\mathbf{x}) + (d + \varepsilon) \equiv -K_v r + \zeta_1 . \end{aligned} \tag{4.7}$$

In adaptive control, though it is often straightforward to show that the tracking error is small, it is usually difficult to demonstrate that the adapted parameter vector remains bounded without a stringent persistence of excitation (PE) condition. This is known as the problem of 'parameter drift'. Alternatives that avoid the need for PE include σ-modification [11], e-modification [10], or a dead-zone approach. In this chapter, this issue manifests itself in proving that the estimate $\hat{U}$ for the control representative value vector U is bounded despite disturbances. We resolve the problem by selecting an e-modification sort of approach to adapt the fuzzy system parameters.

Theorem 4.1: Let the desired trajectory $\mathbf{x}_d(t)$ be bounded as in Assumption 2.1. Take the control input for (2.1) as (4.6), and let the fuzzy system control representative values be tuned on-line by

$$\dot{\hat{U}} = FP(\mathbf{x})r - \kappa F\|r\|\hat{U}, \tag{4.8}$$

with $F = F^T > 0$ a constant matrix, normally chosen diagonal, and $\kappa > 0$ a design parameter. Then, for high enough gains K_v, the filtered tracking error $r(t)$ and the control representative value estimates $\hat{U}(t)$ are UUB, with practical bounds given specifically by the right-hand sides of (4.9) and (4.10). Moreover, $\|r(t)\|$ can be made as small as desired by increasing the control gain K_v.

Proof: The practical bounds on the filtered tracking error $r(t)$ and control representative value estimation error $\tilde{U}(t)$ respectively are simply stated as

$$|r| > \frac{\kappa U_{\max}^2/4 + (\varepsilon_N + b_d)}{K_{v\min}} = b_r \tag{4.9}$$

and

$$\|\tilde{U}\| > U_{\max}^2/2 + \sqrt{U_{\max}^2/4 + (\varepsilon_N + b_d)/\kappa} = b_U, \tag{4.10}$$

while the entire proof can be found in [13]. Since any values above these bounds cause the Lyapunov function to decrease, thereby decreasing $r(t)$ and/or $\tilde{U}(t)$. It can be seen from (4.9) that the tracking error bound can be made arbitrarily small by increasing the outer-tracking-loop gain $K_{v\min}$. ■

The adaptation law (4.8) for the fuzzy system parameters is worth discussing. The first term is a gradient-descent-type term for adjusting the parameters. According to the theorem, this term by itself cannot be shown to yield closed-loop stability. Indeed, the second term is required; this is a term well-known in adaptive control known as the e-modification term [10]. It is required to guarantee robustness of the controller, including bounded control signals, despite unknown disturbances.

5 Simulations

As an example consider the nonlinear system

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= (1-2e^{-(x_1^2+x_2^2)})x_2 + x_1 + 2 + u \end{aligned} \tag{5.1}$$

which is in the controllability canonical form (2.1) and unstable (i.e. the linearization of the system at the origin is unstable). For the fuzzy system membership functions, the states are partitioned uniformly in both x_1 and x_2 and

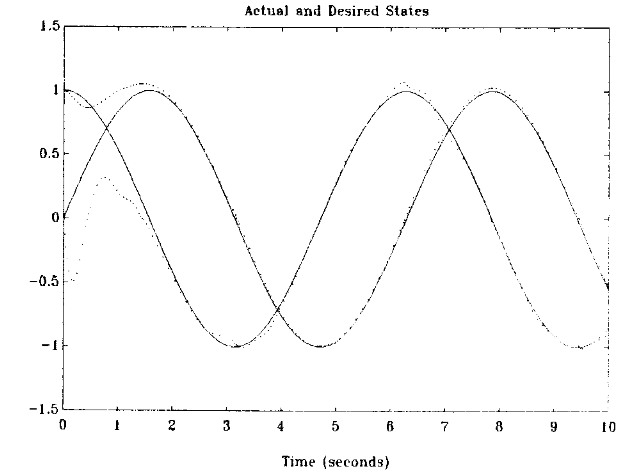

Figure 3: Actual (dotted) and desired (solid) states with $K_v = 10$, $x_1(0) = 1$, and $x_2(0) = 0$.

since the signed magnitude of the desired trajectory $y_d(t) = \sin(t)$ is symmetric about the origin, we construct the partitions symmetric about and including zero. For this example the partition refinement was chosen as $N_1 = N_2 = 11$ although performance did not degrade appreciably for values of $N_i \geq 7$. For the first simulation, design parameters are set to $K_v = 10,\ \lambda_1 = 2,\ \kappa = 0.5,\ F = 20 \cdot I_{121}$. Initial conditions are $\hat{U}^T(0) = 0$, $x_1(0) = 1$, and $x_2(0) = 0$. The desired tra-

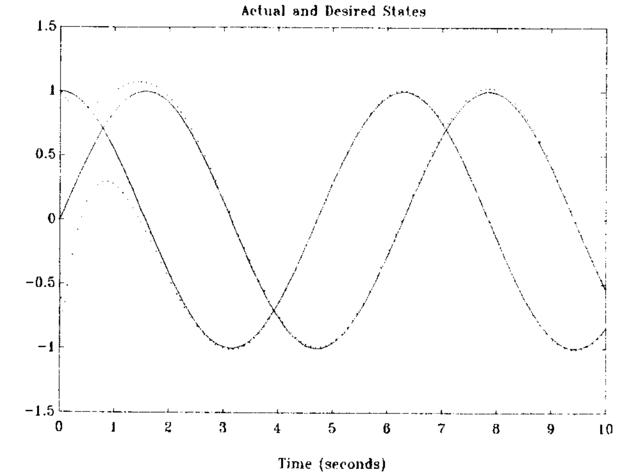

Figure 4: Actual (dotted) and desired (solid) states with $K_v = 50$, $x_1(0) = 1$, and $x_2(0) = 0$.

jectory is selected as $y_d(t) = \sin(t)$. Using the FLC in Figure 2, the actual and desired states are plotted in Figure 3. Note that good tracking is obtained considering that the nonlinear plant is unknown (i.e. coefficients inside $f(\mathbf{x})$).

To show the effects of increasing the outer-loop control gains, a second simulation was done with design parameters $K_v = 50,\ \lambda_1 = 2,\ \kappa = 0.5,\ F = 20 \cdot I_{121}$ while keeping all initial conditions the same. That is, the outer-loop control gain is increased. The FLC results are given in Figure 4, where the actual states (dotted) almost perfectly track desired states (solid) in less than 2 seconds.

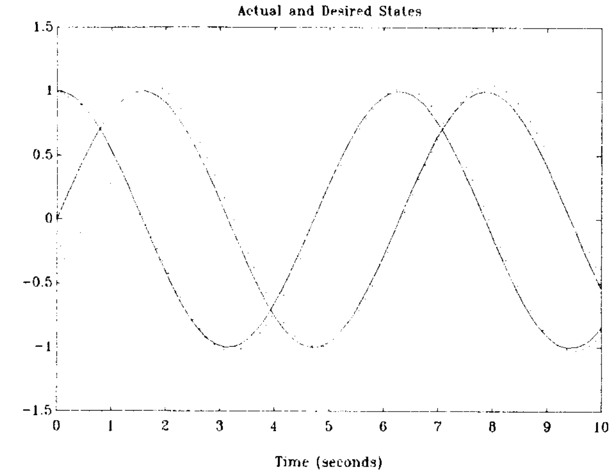

Figure 5: Response of Adaptive Controller with full regression matrix (dotted) where the desired states are (solid). $K_v = 6$, $K_p = 10$.

For comparison, a robust adaptive controller [4] is given by

$$u = \ddot{e} + K_v \dot{e} + K_p e + W(x_1, x_2, \dot{x}_2)\hat{\varphi} \tag{5.2}$$

$$\dot{\hat{\varphi}} = \Gamma W^T(\cdot) B^T P [e \quad \dot{e}]^T \tag{5.3}$$

with $\Gamma^T = \Gamma > 0$, K_p, and K_v design parameters, $W(x_1, x_2, \dot{x}_2)$ regression matrix of functions that must be explicitly derived from the specific system dynamics, and φ a vector of unknown parameters. In this case φ is simply the constant coefficients of all terms besides u in the second equation of (5.1), and the regression matrix is

$$W = \begin{bmatrix} x_2 & \exp(-x_1^2 - x_2^2)x_2 & x_1 & 1 \end{bmatrix}. \tag{5.4}$$

Transforming system (6.1) as in [4] yields

$$A = \begin{bmatrix} 0 & 1 \\ -K_p & -K_v \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

In (5.3) P satisfies the Lyapunov equation $A^T P + PA = -Q$ where Q is chosen as the identity matrix. The system parameters are $\varphi^T = [1 \ \ -2 \ \ 1 \ \ 2]$, and we selected the same desired trajectory as above, $\Gamma = diag(0.2, 0.2, 0.2, 0.2)$, $K_p = 10$, $K_v = 6$. The response using this controller with $\hat{\varphi}(0) = [0.5 \ \ -1.9 \ \ .5 \ \ 1.9]$, $x_1(0) = 1$, and $x_2(0) = 0$ appears (dotted) in Figure 5. Note the good behavior which results with four unknown parameters since the regression matrix is complete. To demonstrate the deleterious effects of the unmodeled dynamics in adaptive control, the (1, 4) entry of the matrix

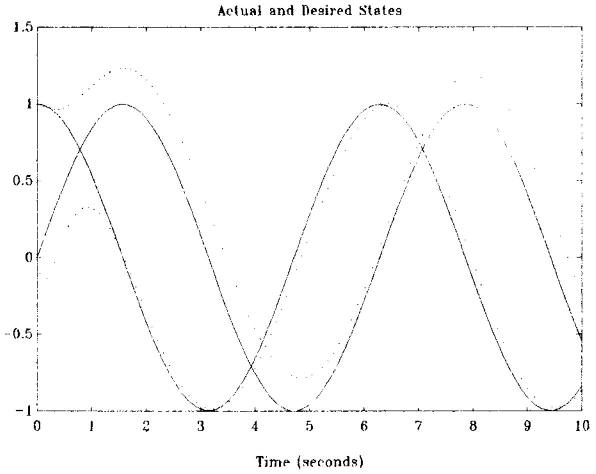

Figure 6: Response of Adaptive Controller without full regression matrix (dotted) where the desired states are (solid). $K_v = 6$, $K_p = 10$.

W was replaced with zero in the controller. Figure 6 shows moderate tracking of x_2 and poor tracking of x_1 (both dot-dashed). It is emphasized that all the dynamics are unmodeled in the universal adaptive FLC in Figures 3 and 4.

6 Conclusions

A universal adaptive fuzzy logic controller is proposed that guarantees prescribed performance for the general class of feedback linearizable nonlinear systems (2.1) with mild assumptions. This controller has a multi-loop structure with a fuzzy logic approximate linearization loop, a robustifying loop, and a unity gain outer tracking loop. The structure of the fuzzy logic controller and adaptation law are chosen such that a Lyapunov-based proof of uniform ultimate boundedness is obtained. The main benefit of this adaptive controller is that it is model-free and requires no regression matrix, unlike adaptive control, because the fuzzy system forms a set of universal basis functions. That is, $P(x)$, the participation vector constructed algorithmically herein, depends on the membership functions chosen for the fuzzy system. This method for controller synthesis is systematic and the proposed adaptive fuzzy-logic controller performs remarkably well as indicated by the simulation results. Furthermore, with the tools presented in the third section, it is evident that problems with high dimensionality, while more mathematically and computationally intensive, are nonetheless manageable.

7 References

[1] J.J. Buckley, ``Universal fuzzy controllers," *Automatica*, vol. 28, no. 6, pp. 1245-1248, 1992.

[2] G. Chen and H. Ying, ``On the stability of fuzzy PI control systems," *Proc. Int. Conf. Industrial Fuzzy Control and Intelligent Sys.*, Dec.,1993.

[3] S. Chiu and S. Chand, ``Fuzzy controller design and stability analysis for an aircraft model," *Proc. Fuzzy Control Workshop for Industrial App.*, Texas A&M Univ., 1991.

[4] J. Craig, *Adaptive Control of Mechanical Manipulators*, Reading, MA, Addison-Wesley, 1985.

[[5] R. Langari and M. Tomizuka, "Analysis of stability of a class of fuzzy linguistic controllers with internal dynamics," *Proc. ASME Winter Annual Meeting*, paper 91-WA-DSC-13, Dec. 1991.

[6] P.M. Larsen, "Industrial Applications of Fuzzy Logic Control," Mamdani and G.R. Gaines eds.,*Fuzzy Reasoning and Its Applications*, London, New York, Toronto, pp. 335-343, 1981.

[7] F.L. Lewis and K. Liu, "Toward a paradigm for fuzzy logic control," submitted, 1994.

[8] F.L. Lewis, K. Liu, and A. Yesildirek, "Neural net robot controller with guaranteed tracking performance," Proc. of the 1993 Int. Symp. on Intelligent Control, pp. 225-231, 1993.

[9] F.L. Lewis, S.Q. Zhu, and K. Liu, " Function approximation by fuzzy systems," submitted to *IEEE Trans. Fuzzy Systems*, 1994.

[10] K.S. Narendra and A.M. Annaswamy, "A new adaptive law for robust adaptation without persistent excitation," *IEEE Trans. Automatic Control*, vol. AC-32, no. 2, pp. 134-145, Feb. 1987.

[11] M.M. Polycarpou and P.A. Ioannu, ``Identification and control using neural network models: design and stability analysis," Tech. Report 91-09-01,Dept. Elect. Eng. Sys., Univ. S. Cal., Sept. 1991.

[12] J.J. Slotine and Li, *Applied Nonlinear Control*, New Jersey: Prentice-Hall, 1990.

[13] M.W. Vandegrift, F.L. Lewis, S. Zhu, and K. Liu, "Fuzzy logic control of a class of feedback ...," submitted to *IEEE Trans. on Fuzzy Systems*, February 1995.

[14] L.-X. Wang and J.M. Mendel, "Fuzzy basis functions, universal approximators, and orthogonal least-squares learning," IEEE Trans. Neural Networks, vol. 3, no. 5, pp. 807-814, Sept. 1992.

[15] T.K. Yin, C.S.G. Lee, "Fuzzy model-reference adaptive control," in review for *IEEE Systems Man and Cybernetics*, 1994.

[16] H. Ying, "Sufficient conditions on general fuzzy systems as function approximators," *Automatica*, vol. 30 no. 3, pp. 521-525, 1994.

[17] H. Ying, "General analytical structure of typical fuzzy controllers and their limiting structure theorems," *Automatica*, vol. 29, no. 4, pp. 1139-1143, 1993.

[18] L.A. Zadeh, "Fuzzy sets," *Informat. Contr.*, vol. 8, pp. 338-353, 1965.

[19] X.-J. Zeng and M.G. Singh, "Approximation theory of fuzzy systems - SISO case," *IEEE Trans. Fuzzy Systems*, vol. 2, no. 2, pp. 162-176, May 1994.