Adaptive Control of a Class of Nonlinear Systems with Fuzzy Logic

Chun-Yi Su and Yury Stepanenko

***Abstract*— An adaptive tracking control architecture is proposed for a class of continuous-time nonlinear dynamic systems, for which an explicit linear parameterization of the uncertainty in the dynamics is either unknown or impossible. The architecture employs fuzzy systems, which are expressed as a series expansion of basis functions, to adaptively compensate for the plant nonlinearities. Global asymptotic stability of the algorithm is established in the Lyapunov sense, with tracking errors converging to a neighborhood of zero. Simulation results for an unstable nonlinear plant are included to demonstrate that incorporating the linguistic fuzzy information from human experts results in superior tracking performance.**

I. Introduction

FUZZY LOGIC CONTROL, as one of the most useful approaches for utilizing expert knowledge, has been a subject of intense research in the past decade. The interested readers are referred to [12] for a recent review. Fuzzy logic control is generally applicable to plants that are mathematically poorly modeled and where experienced operators are available for providing qualitative guiding. Although achieving much practical success, fuzzy control has not been viewed as a rigorous science, due to a lack of formal synthesis techniques which guarantee the very basic requirements of global stability and acceptable performance [25]. In the stability analysis [8], [11], it is commonly assumed that the mathematical model of the plant is known. This assumption contradicts the very basic premise of fuzzy control systems, i.e., to control processes that are poorly modeled from a mathematical point of view.

The design of the globally stable fuzzy control system was an open problem until recent efforts presented in [25]. Based on the result that fuzzy systems are capable of approximating, with arbitrary accuracy, any real continuous function on a compact set, a global stable adaptive fuzzy controller is firstly synthesized from a collection of fuzzy IF-THEN rules [24]. The fuzzy system, used to approximate an optimal controller, is adjusted by an adaptive law based on a Lyapunov synthesis approach.

The goal of the research described in this paper is the development of a globally stable adaptive controller for a class of continuous-time nonlinear dynamic systems for which an explicit linear parameterization of the uncertainty in the dynamics is either unknown or impossible. The controller is designed by using fuzzy IF-THEN rules from human experts, and some additional rules. Unlike [25], the fuzzy system is used to model the plant and the controller is constructed based on this fuzzy model so that fuzzy IF-THEN rules describing the plant can be incorporated into the adaptive fuzzy controller. The model used for the controller design belongs to the category of fuzzy modeling. As indicated by [21], though the term *fuzzy modeling* has not been used often, fuzzy modeling is the most important issue in fuzzy theory since the linguistic fuzzy IF-THEN rules can often be obtained from human experts who are familiar with the system under consideration. These linguistic rules are very important and often contain information about how the system behaves. Such information is not contained in the input-out pairs obtained by measuring the outputs of a system for certain text input, because the test inputs may not be rich enough to excite all the modes of the system.

Inspired by the work in [24], we express a fuzzy model for plants as a series expansion of basis functions that are named fuzzy basis functions. As indicated in [24], the most important advantage of a fuzzy basis function expansion is the provision of a natural framework for combining numerical and linguistic information in a uniform fashion. In this paper the fuzzy basis function expansion is used to approximate a plant and the adaptive control law is therefore designed through the following three steps: first, we define some fuzzy sets whose membership functions cover the state space; then, we use fuzzy IF-THEN rules from human experts and some arbitrary rules to construct an initial model of the plant in which some parameters are free to change. Finally, we develop an adaptive law to adjust free parameters based on a Lyapunov synthesis approach. Although the fuzzy description are not precise and may not be sufficient for achieving the accurate approximation, in our design the modeling error is not required to be necessarily small; the designed control law can adaptively compensate for the modeling error. If the fuzzy IF-THEN rules from human experts provide a good fuzzy model, then the compensating term is relatively minor. On the other hand, if the linguistic rules from human experts are poor, then the compensating term is dominant and our adaptive fuzzy controller becomes a regular robust nonlinear controller, similar to the case of the radial basis function adaptive controller [18].

The arrangement of this letter is as follows. In Section II, the mathematical structure of the control problem and the form of conventional adaptive solutions are examined. Section III is devoted to the general problem of function approximation using fuzzy IF-THEN rules from human experts. By expressing fuzzy systems as series expansions of fuzzy basis functions which are algebraic superpositions of fuzzy membership functions, we show that such a fuzzy system is capable of approximating, with arbitrary accuracy, any real continuous function on a compact set. Since the fuzzy system is assumed to consist of only a finite number of IF-THEN rules, its approximation capabilities can be guaranteed only

Manuscript received November 10, 1993; revised March 3, 1994.
The authors are with the Department of Mechanical Engineering, University of Victoria, Victoria, British Columbia, Canada V8W 3P6.
IEEE Log Number 9403187.

1063-6706/94$04.00 © 1994 IEEE

on a subset of the entire plant state space; beyond this subset the approximating error may not be in a chosen tolerance. Section IV exploits the fuzzy systems to develop adaptive control algorithms for the systems considered in Section II. Due to the approximating errors, bounded only on a subset, an additional, nonadaptive component in the algorithm is necessary. Combining these components into a single control law, it is then demonstrated that these mechanisms ensure asymptotic convergence of the tracking errors. In Section V, the adaptive fuzzy controller is used to control an unstable system. Section VI concludes the paper.

II. Problem Statement

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

$$x^{(n)}(t) + f\big(x(t), \dot{x}(t), \ldots, x^{(n-1)}(t)\big) = bu(t) \qquad (1)$$

where $u(t)$ is the control input, f is an unknown linear or nonlinear function, and b is the control gain. It should be noted that more general classes of nonlinear control problems can be transformed into this structure [6].

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

One approach to this problem is firstly to define an error metric as

$$s(t) = \left(\frac{d}{dt} + \lambda\right)^{(n-1)} \tilde{x}(t) \qquad \text{with} \qquad \lambda > 0 \qquad (2)$$

which can be rewritten as $s(t) = \Lambda^T \tilde{\mathbf{x}}(t)$ with $\Lambda^T = [\lambda^{(n-1)}, (n-1)\lambda^{(n-2)}, \ldots, 1]$. 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 [20]. The control input is then taken as $u(t) = -k_d s(t) + u_{fd}(t) + u_{fu}(t)$ where $u_{fd}(t) = x_d^{(n)}(t) - \Lambda_v^T \tilde{\mathbf{x}}(t)$ with $\Lambda_v^T = [0, \lambda^{(n-1)}, (n-1)\lambda^{(n-2)}, \ldots, (n-1)\lambda]$ is a linear combination of tracking error states, $u_{fu}(t)$ is an adaptive control law which will attempt to recover and cancel the unknown function $f(\mathbf{x}(t))$, k_d is a constant. With use of this control law, the time derivative of the error metric (2) can be written as:

$$\begin{aligned} \dot{s}(t) &= -x_d^{(n)}(t) + \Lambda_v^T \tilde{\mathbf{x}}(t) - f\big(\mathbf{x}(t)\big) + u(t) \\ &= -k_d s(t) + u_{fu}(t) - f(\mathbf{x}(t)) \end{aligned} \qquad (3)$$

In a typical adaptive approach one states *a priori* that $f(\mathbf{x}(t))$ lies in the span of a set of continuous, known (linear or nonlinear) *basis functions*, $Y_i(\mathbf{x})$, *i.e.*:

$$f(\mathbf{x}(t)) = \sum_{i=1}^{N} \theta_i Y_i(\mathbf{x}) \qquad (4)$$

then the adaptive controller can be chosen as

$$u_{fu}(t) = \sum_{i=1}^{N} \hat{\theta}_i Y_i(\mathbf{x}) \qquad (5)$$

where $\hat{\theta}_i$ is an approximation to the ith coefficient in the expansion of $f(\mathbf{x}(t))$. Hence,

$$u_{fu}(t) - f\big(\mathbf{x}(t)\big) = \sum_{i=1}^{N} \tilde{\theta}_i Y_i(\mathbf{x}) \qquad (6)$$

where $\tilde{\theta}_i = \hat{\theta}_i - \theta_i$. The tracking problem can thus be solved if a law for adjusting $\hat{\theta}_i$ can be created which simultaneously ensures that the parameter estimates and tracking errors remain bounded, and that $\tilde{\mathbf{x}} \to \mathbf{0}$ according to the first order differential equation of the form:

$$\dot{s}(t) = -k_d s(t) + \sum_{i=1}^{N} \tilde{\theta}_i Y_i(\mathbf{x}). \qquad (7)$$

These facts form the basis for most of the stable adaptive tracking controllers. The drawback is the relatively large amount of prior information which must be in the form of the exact structure of the basis function required for the computation of $u_{fu}(t)$. In the following section, it will be shown that, with linguistic fuzzy description, these functions can be *approximated* using a class of fuzzy systems whose mathematical structure strongly resembles (4). These approximations can then be stably tuned, using methods inspired by those described above, to produce an effective tracking control architecture.

III. Fuzzy Systems and Function Approximation

If the exact basis functions, $Y_i(\mathbf{x})$, are unknown, it may be possible to estimate the required control input using a large collection of simple elementary function. Memory based methods and their variants [1], partition the state space R^n into small hypercubes, and attempt to estimate the values assumed by f in each hypercube. Neural network approaches [14], [17] attempt to reconstruct f from compositions and superpositions of simple nonlinearities. Fuzzy systems offer yet a third approach, attempting to recover f by using a linguistic fuzzy IF-THEN rule from human experts [10], [24].

In order to develop stable control laws, it is necessary to quantify the capability of the chosen representation to approximate the required functions. For the fuzzy systems, the theoretical ability to uniformly approximate continuous functions to a specified degree of accuracy has recently been demonstrated in [10] [24] by using fuzzy IF-THEN rules, which describe the behavior of an unknown plant.

A. Fuzzy Systems

In this section, we consider a fuzzy system whose basic configuration is shown in Fig. 1 [12], [24]. There are four principal elements in such a fuzzy system: fuzzifier, fuzzy rule base, fuzzy inference engine, and defuzzifier. We consider multi-input, single-output fuzzy systems: $U \subset R^n \to R$,

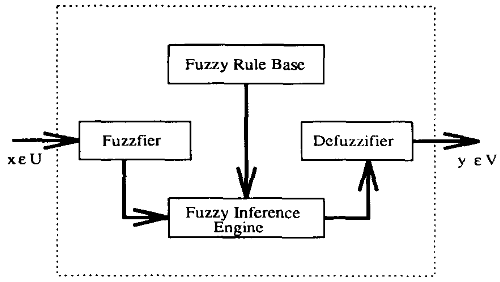

Fig. 1. Basic configuration of fuzzy logic systems.

where U is compact. A multi-output system can always be separated into a group of single-output systems.

The *fuzzifier* performs a mapping from the observed crisp input space $U \subset R^n$ to a fuzzy set defined in U, where a *fuzzy set* [26] defined in U is characterized by a membership function $\mu_F : U \rightarrow [0, 1]$, and is labelled by a linguistic term F such as "small," "medium," "large," or "very large." The most commonly used fuzzifier is the *singleton fuzzifier, i.e.*, $\mu_{Ax}(\mathbf{x}') = 1$ for $\mathbf{x}' = \mathbf{x}$ and $\mu_{Ax}(\mathbf{x}') = 0$ for all other $\mathbf{x}' \in U$ with $\mathbf{x}' \neq \mathbf{x}$.

The *Fuzzy rule base* consists of a set of linguistic rules in the form of "IF a set of conditions are satisfied, THEN a set of consequences are inferred." In this paper, we consider the case where the fuzzy **rule** base consists of N rules in the following form:

$$R_j{:}IFx_1 \text{ is } A_1^j \text{ and } x_2 \text{ is } A_2^j \text{ and } \cdots \text{ and } x_n \text{ is } A_n^j, \\ \text{THEN } z \text{ is } B^j,$$

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_i^j and B^j are linguistic term characterized by fuzzy membership functions $\mu_{A_i^j}(x_i)$ and $\mu_{B^j}(z)$, respectively. Each R_j can be viewed as a *fuzzy implication* $A_1^j \times \cdots \times A_n^j \rightarrow B^j$, which is a fuzzy set in $U \times R$ with $\mu_{A_1^j \times \cdots \times A_n^j \rightarrow B^j}(\mathbf{x}, z) = \mu_{A_1^j}(x_1) \star \cdots \star \mu_{A_1^j}(x_n) \star \mu_{B^j}(z)$ (other operations are possible, see [12]), and the most commonly used operation for "$\star$" are "product" and "min" [12].

The *fuzzy inference engine* is decision making logic which employs fuzzy rules from the fuzzy rule base, to determine a mapping from the fuzzy sets in the input space U to the fuzzy sets in the output space R. Let A_x be an arbitrary fuzzy set in U; then each R_j determines a fuzzy set $A_x \circ R_j$ in R based on the sup-star composition [12]:

$$\begin{aligned}\mu_{A_x \circ R_j}(z) &= \sup_{\mathbf{x} \in U} [\mu_{Ax}(\mathbf{x}) \star \mu_{A_1^j \times \cdots \times A_n^j \rightarrow B^j}(\mathbf{x}, z)] \\ &= \sup_{\mathbf{x} \in U} [\mu_{Ax}(\mathbf{x}) \star \mu_{A_1^j}(x_1) \star \cdots \\ &\quad \star \mu_{A_1^j}(x_n) \star \mu_{B^j}(z)]. \end{aligned} \tag{8}$$

The *defuzzifier* performs a mapping from the fuzzy sets $A_x \circ R_j$ in R to a crisp point in $z \in R$. This mapping may be chosen as weighted average *centroid defuzzifier* [24]:

$$z = \frac{\sum_{j=1}^N w_j \mu_{A_x \circ R_j}(w_j)}{\sum_{i=1}^N \mu_{A_x \circ R_j}(w_j)} \tag{9}$$

where w_j is the point in R at which $\mu_{B^j}(z)$ achieves its maximum value (usually, we assume that $\mu_{B^j}(w_j) = 1$).

Since in this paper the linguistic fuzzy IF-THEN rules are only used for the purpose of approximating the required functions, we define the defuzzifier as a weighted sum of each rule's output [5] [22]:

$$z = \sum_{j=1}^N w_j \mu_{A_x \circ R_j}(w_j). \tag{10}$$

From the above we see that a MISO fuzzy system is a complicated nonlinear system which maps a nonfuzzy $U \subset R^n$ into the nonfuzzy R. Since a fuzzy system is determined by its design parameters, then next we briefly discuss the roles of each design parameter.

The number of fuzzy sets, defined in the input and output universes of discourse, and the number of fuzzy rules in the fuzzy rule base heavily influence the complexity of a fuzzy system, where complexity includes computational complexity, i.e., the computational requirements of the fuzzy system, and space complexity, i.e., the storage requirements of the fuzzy system. These parameter can be viewed as structure parameters of a fuzzy system. In general, the larger these parameters, the more complex is the fuzzy system, and the higher the expected performance of the fuzzy system. Hence, there is always a trade-off between complexity and accuracy in the choice of these parameters; and their choice is usually quite subjective.

The membership functions of the fuzzy sets heavily influence the *smoothness* of the input-output surface determined by the fuzzy system. In general, the *sharper* the membership functions, the less smooth is the input-output surface. The choice of membership functions is also quite subjective.

The linguistic statements of the fuzzy rules are the heart of a fuzzy system in the sense that it is these linguistic statements that contain most of the information concerning the fuzzy system design; all other design parameters assist in the effective representation and use of the information. The fuzzy rules usually come from two sources: human experts, and training data. Methods of generating fuzzy rules from numerical data may refer to [9].

B. Fuzzy Systems as Fuzzy Basis Expansion

As in [24], we now consider a subset of the fuzzy systems of Fig. 1.

Definition 1: The set of a fuzzy system with *singleton fuzzifier, product inference, and Gaussian membership function* consists of all functions of the form

$$f(\mathbf{x}) = \sum_{j=1}^N w_j \left(\prod_{i=1}^n \mu_{A_i^j}(x_i) \right) \tag{11}$$

where $f : U \subset R^n \rightarrow R$, $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in U$; $\mu_{A_i^j}(x_i)$ is the *Gaussian membership function*, defined by

$$\mu_{A_i^j}(x_i) = \exp\left[-\left(\frac{x_i - \xi_i^j}{\sigma_j}\right)^2\right] \tag{12}$$

where ξ_i^j, σ_j are real-valued parameters, and w_j is the point in R at which $\mu_{B^j}(z)$ achieves its maximum value.

Clearly, (11) is obtained by substituting (8) into (10), replacing the "$\star$" with " product" (product inference), and considering the fact that by using the singleton fuzzifier and assuming that $w_j = 1$, we have $\mu_{A_x \circ R_j}(w_j) = \prod_{i=1}^n \mu_{A_i^j}(x_i)$.

If we take $(\prod_{i=1}^n \mu_{A_i^j}(x_i))$ as basis functions and w_j as constants, then $f(\mathbf{x})$ of (11) can be viewed as a linear combination of the basis functions.

Definition 2: Define *fuzzy basis functions* (FBF's) as

$$g_j\left(\frac{\|\mathbf{x} - \xi_j\|}{\sigma_j}\right) = \prod_{i=1}^n \mu_{A_i^j}(x_i), \quad j = 1, 2, \ldots, N. \tag{13}$$

where $\mu_{A_i^j}(x_i)$ are the Gaussian membership functions (12), $\xi_j = (\xi_1^j, \xi_2^j, \ldots, \xi_n^j) \in U$. Then, the fuzzy system (11) is equivalent to an *FBF expansion*:

$$f(\mathbf{x}) = \sum_{j=1}^N w_j g_j\left(\frac{\|\mathbf{x} - \xi_j\|}{\sigma_j}\right). \tag{14}$$

As in [24], we see that an FBF corresponds to a fuzzy IF-THEN rule. Specifically, an FBF for R_j can be determined as follows. First, calculate the product of all membership functions for the linguistic terms in the IF part of R_j, and call it a *pseudo*-FBF for R_j; then, after calculating the *pseudo*-FBF's for all N fuzzy IF-THEN rules, the FBF for R_j is determined by the sum of all the N pseudo-FBF's. An FBF can either be determined based on a given linguistic rule as above or generated based on a numerical input-output pair.

C. Function Approximation

We now show an important property of the FBF of (13). Let A be the set of all the FBF expansions (14) with $g_j(\mathbf{x})$ given by (13), and $d_\infty(f_1, f_2) = \sup_{\mathbf{x}\in U}(|f_1(\mathbf{x}) - f_1(\mathbf{x})|)$ be the sup-metric; then, (A, d_∞) is a metric space [16]. The following theorem shows that (A, d_∞) is dense in $(C[U], d_\infty)$, where $C[U]$ is the set of all real continuous functions defined on U.

Theorem 1: For any given real continuous function h on the compact set $U \subset R^n$ and arbitrary $\epsilon > 0$, there exists $f \in A$ such that

$$\sup_{\mathbf{x}\in U}|h(\mathbf{x}) - f(\mathbf{x})| < \epsilon. \tag{15}$$

A proof of this theorem is in the same spirit as [24]. We omit the proof here to save space. □

This theorem states that the FBF expansion (14) are *universal approximators*. Therefore, the fuzzy logic systems (14) are qualified to estimate the unknown function f. It is important to note that, since the fuzzy logic systems (14) are constructed from the fuzzy IF-THEN rules, linguistic information, describing the plant from human experts, can be directly incorporated to estimate the unknown function f. In this case, one may synthesize the controllers assuming that the fuzzy logic systems represent (approximately) the true plant.

Although the results referred to above at first sight appear attractive, they do not provide much insight into practical questions: For a prespecified accuracy on a given compact set $U \subset R^n$, how many numbers of fuzzy IF-THEN rules are required? In fact, the results were achieved by placing no restriction on the number of rules used. As a matter of fact, the membership function can be selected as other forms such as sigmoidal functions. It can easily be proven that the above theorem is still held. It is clear that the property of approximating functions arbitrarily well is not sufficient for characterizing good approximation schemes. The key property is not that of arbitrary approximation, but the property of *best approximation*, given a finite set of data points in $U \subset R^n$. It is shown in [14] that Gaussian basis functions do have the best approximation property. This is the main reason we choose the Gaussian functions as the membership functions. The sections below seek a related question: for a *given* number of fuzzy IF-THEN rules, how should the parameters w_j, ξ_j and σ_j be chosen to ensure the best approximation?

To address this question, it is helpful to first realize the functional equivalence between *Gaussian Radial Basis Function* (GRBF's) networks [14] and the fuzzy system with FBF's in (13), which was recently illustrated in [7]. Though these two models are motivated from different origins, under some minor restrictions they are functionally equivalent; the theorem on the approximations for one model can be applied to the other, and vice versa.

In theory, the GRBF network is capable of forming an arbitrarily close approximation to any continuous nonlinear mapping [4], [14]. This, from another point of view, verifies Theorem 1. Very recently, however, several major new theorems [3] have appeared regarding the theoretical approximation capability of GRBF networks, indicating that proper choices of the connection weights can result in rapid convergence of the worst case approximation error as a function of size.

In the following, we briefly discuss the new results on the approximation capabilities of a GRBF network which motivate the new adaptive fuzzy control method in this paper. Starting from a lemma due to [3] is able to prove the following: Suppose f can be expressed on a compact subset, A, of R^d by the inverse Fourier transform $f = F^{-1}(Ff)$, and Ff has compact support and is in addition $k > d/2$ times continuously differentiable. Then there exist constants w_j^*, ζ_j^*, and a constant $\kappa(f, d)$ which depends upon the function and the dimension, such that

$$\operatorname*{ess\,sup}_{\mathbf{x}}\left|f(\mathbf{x}) - \sum_{j=1}^N w_j^* g_j\left(\frac{\|\mathbf{x} - \zeta_j^*\|}{\sigma_j}\right)\right|^2 \le \frac{\kappa(f, d)}{N} \tag{16}$$

That is, it is possible to choose the centers of an FBF's expansion in such a way that bandlimited functions with sufficiently differentiable spectra can be uniformly approximated to a degree of accuracy, whose square improves linearly with the size of the fuzzy IF-THEN rule.

This linear rate of convergence result can be interpreted as stating the existence of particular parameters ξ_j^* and w_j^* which allow the indicated rate of uniform convergence for certain bandlimited functions.

The actual nonlinearity f to be approximated may not be Fourier transformable, or if transformable, the resulting transform may not be continuously differentiable. In this case, the above result may not be held. On the other hand, the particular parameters ξ_j^* and w_j^* may not easily be identified to achieve the theoretical convergence rate. Since approximation is required only on a chosen compact set, A, based on the above result, in this paper we simply assume that f on the set A satisfies the above condition and there exists a uniform bound ϵ such that

$$\sup_x \left| f(\mathbf{x}) - \sum_{j=1}^{M} w_j^* g_j\left(\frac{\|\mathbf{x}-\zeta_j\|}{\sigma_j}\right)\right| \leq \epsilon + \alpha_1(\mathbf{x}) \tag{17}$$

where $\alpha_1(\mathbf{x}) = 0$ if $\mathbf{x} \in A$.

The approximation error outside the set A represented by $\alpha_1(\mathbf{x})$ is of great importance. This error bound reflects the fact that the accuracy of the estimate implemented by the representations considered above may not be bounded by a constant and degrade drastically outside A. This possible rapid degradation of the fuzzy approximation outside of the subset for which it was designed will significantly influence the adaptive control algorithms, which the following section now presents in detail.

It should be noted that neither of explicit expressions for computation of w_j^* or ϵ are required since these functions can be learned by using the adaptive algorithm developed in the following section.

IV. Adaptive Control with Fuzzy Logic

In this section, the above discussion of the approximating power of fuzzy systems is developed into a complete specification of an adaptive tracking control architecture based upon these models, which is then proven effective using the Lyapunov stability theory. To begin combining the results of Section III, notice that if the desired trajectories are contained in a compact subset, A_d, of the state space, in principle the tracking problem posed by system (1) could be solved by an adaptive component in the control law capable of reconstructing the unknown function f everywhere on A_d. Since expansion (14) provides a fuzzy structure which can approximate unknown plant nonlinearities on a particular compact subset, the adaptive component can be taken as the output of this fuzzy system, instead of the exact basis functions $Y_i(\mathbf{x})$ used in Section II.

Suppose then that a fuzzy system, designed and analyzed with the methodology of the preceding section, is used as the adaptive component of the control architecture, with the fuzzy basis function replacing the Y_i. The error equation (7) then becomes:

$$\begin{aligned}\dot{s}(t) &= -k_d s(t) - f(\mathbf{x}) + \sum_{i=1}^{N} \hat{w}_j g_j\left(\frac{\|\mathbf{x}-\xi_j\|}{\sigma_j}\right)\\ &= -k_d s(t) + \sum_{i=1}^{N} \tilde{w}_j g_j\left(\frac{\|\mathbf{x}-\xi_j\|}{\sigma_j}\right) + d(t)\end{aligned} \tag{18}$$

where $\hat{w}_j(t)$ is an estimate of w_j^*, $\tilde{w}_j(t) = \hat{w}_j(t) - w_j^*$, and $|d(t)| \leq \epsilon + \alpha_1(\mathbf{x})$, with $\alpha_1(\mathbf{x}) = 0$ if $\mathbf{x} \in A_d$. This is almost identical to (7) except for the presence of the disturbance $d(t)$, which describes the difference between the actual function f, and the best possible fuzzy approximation to this function. Despite the similarities in the structure of the error equation, the presence of this disturbance term fundamentally changes the stability properties of classical adaptive control algorithms.

When the disturbance is uniformly bounded, one can employ a number of techniques, grouped under the heading of robust adaptive control. The disturbance will remain bounded, however, only if the plant state never leaves A_d. Unfortunately, even with the modified adaptive algorithm, it is not possible to guarantee *a priori* that the plant state will remain within any subset of its state space. Indeed, it is possible that, during the early stages of learning when the initial fuzzy system approximations may be quite poor, the tracking errors would become so large that the plant state would leave the set A_d. Further, impulsive, unmodeled disturbances might also take the plant state outside this set. From its definition, the term $\alpha_1(\mathbf{x})$ can become quite large in this region, reflecting the rapid degradation of the ability of the fuzzy system to approximate the function f outside the set for which it was designed.

Inspired by the recent work on adaptive control [18], however, it can easily be overcome by including in the control law a component, known as sliding control [23], which takes over from the adaptive component as its approximation ability begins to degrade, and which forces the state back into A_d. Since the stability of the adaptive operation can be guaranteed only as long as the disturbance effects can be uniformly bounded, the entire adaptive subsystem must be turned off if ever the state moves outside the set A_d, and remain off until the sliding controller can return the state to this region. Similarly, the sliding controller should be turned off when the state occupies regions where the fuzzy system has good approximating properties, since the former relies on crude upper bounds of the plant nonlinearities to reduce the tracking errors, and hence is likely to require large amounts of control authority when active.

The complete control law analyzed below therefore has a dual character, acting as either a sliding or an adaptive controller, depending upon the instantaneous location of the plant state vector. Thus, unlike the classical adaptive models considered in Section II, whose basis functions are sufficient to achieve a globally exactly match to the plant dynamics, the representation implemented by an adaptive fuzzy system is only approximately accurate on a subset of the entire state space. This introduces additional complexities in the design of a stable tracking control strategy whose solutions require a combination of techniques from both a robust adaptive and

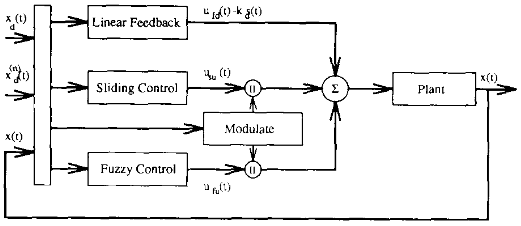

Fig. 2. Closed-loop control system.

a robust nonlinear control theory. The resulting composite controller is capable of a globally stable solution to the tracking problem posed in Section II.

A. Controller Structure

In this section, the above ideas are applied to design a control law for a system in the canonical form:

$$x^{(n)}(t) + f\big(x(t), \dot{x}(t), \ldots, x^{(n-1)}(t)\big) = u(t) \tag{19}$$

where it is assumed that a prior upper bound $D(\mathbf{x})$ is known on the magnitude of f for points outside of the set A_d, *i.e.*:

$$|f(\mathbf{x})| \le D(\mathbf{x}) \text{ when } \mathbf{x} \in A_d^c$$

Having chosen a set A_d as outlined above, let $f_A(\mathbf{x})$ be a fuzzy approximation to $f(\mathbf{x})$ designed such that $|f_A - f| \le \epsilon$ uniformly on the set A_d.

The observations of the previous section suggest a control law with the general structure:

$$u(t) = -k_d s(t) + u_{fd}(t) + \big(1 - m(t)\big)u_{fu} + m(t)u_{su}(t). \tag{20}$$

As in Section II, $s(t)$ is given in (2), $u_{fd}(t)$ is a negative feedback term including a linear combination of tracking error states given by

$$u_{fd}(t) = x_d^{(n)}(t) - \Lambda_v^T \tilde{\mathbf{x}}(t) \text{ with}$$
$$\Lambda_v^T = [0, \lambda^{(n-1)}, (n-1)\lambda^{(n-2)}, \ldots, (n-1)\lambda]. \tag{21}$$

The adaptive component of the control law is synthesized by

$$\begin{aligned} u_{fu} &= \hat{f}_A\big(t, \mathbf{x}(t)\big) - \hat{\epsilon}_d \mathrm{sgn}(s) \\ &= \sum_{j=1}^{N} \hat{w}_j g_j\left(\frac{\|\mathbf{x} - \zeta_j\|}{\sigma_j}\right) - \hat{\epsilon}_d \mathrm{sgn}(s) \end{aligned} \tag{22}$$

$$\dot{\hat{w}}_j = -\eta_1\big(1 - m(t)\big)s(t)g_j\left(\frac{\|\mathbf{x} - \zeta_j\|}{\sigma_j}\right) \tag{23}$$

$$\dot{\hat{\epsilon}}_d = \eta_2\big(1 - m(t)\big)|s(t)| \tag{24}$$

where η_1 and η_2 are constants which determine the rate of adaptation; and the sliding component is synthesized by

$$u_{su}(t) = -k_{su}(t)\mathrm{sgn}\big(s(t)\big) \tag{25}$$

where k_{su}, satisfying $k_{su}(t) \ge D(\mathbf{x})$, is the gain of the sliding controller. The function $m(t)$ is a state dependent modulation which allows the controller to transition between sliding and adaptive modes of operation, chosen so that $m(t) = 0$ when $\mathbf{x}(t) \in A_d$ and $m(t) = 1$, otherwise. A block diagram of this controller structure is shown in Fig. 2 for reference.

Remarks

1) The term $\hat{\epsilon}_d \mathrm{sgn}(s)$ actually reflects the component for compensation of the approximating error $d(t) = f_A(\mathbf{x}) - f(\mathbf{x})$ for $\mathbf{x} \in A_d$. If the linguistic rules provide good description of the unknown function f in A_d, then the approximating error $d(t)$ should be small. As a result, the adaptive control u_{fu} behaves approximately like the conventional adaptive controller when $Y_i(\mathbf{x})$ given by (4) are known. If linguistic information is poor, the adaptive compensator term $\hat{\epsilon}_d$ will rise automatically to the necessary level, ensuring the stability of the overall system.
2) Compared with the other adaptive fuzzy scheme given in [25], there are some important differences. First, based on the definition in [25], the scheme of [25] belongs to the direct adaptive fuzzy control category since the fuzzy logic systems are directly used as controllers. Our controller may be defined as the indirect adaptive fuzzy controller, because the controller uses fuzzy logic systems as the model of the plant. Second, in [25] the convergence of the tracking error depends on the the assumption that $\int_0^\infty |e(t)|^2 dt \le \infty$, where $e(t)$ is the approximation error between the actual control and the optimal control, which may not be easy to check. On the other hand, in our scheme the approximation error is not required to be necessarily small since an adaptive compensator is introduced in the controller, guaranteeing the convergence of the tracking error.

B. Stability Analysis

The time derivative of the error metric (2) can then be written as:

$$\dot{s}(t) = -x_d^{(n)}(t) + \Lambda_v^T \tilde{\mathbf{x}}(t) - f_A\big(\mathbf{x}(t)\big) + u(t) + d(t) \tag{26}$$

where the disturbance $d(t) = f_A(\mathbf{x}(t)) - f(\mathbf{x}(t))$ satisfies $|d(t)| \le \epsilon$ for all t such that $\mathbf{x}(t) \in A_d$.

Using the control law (20), the above equation can be rewritten as:

$$\begin{aligned} \dot{s}(t) &= -k_d s(t) - f_A\big(\mathbf{x}(t)\big) + \big(1 - m(t)\big)\hat{f}_A\big(t, \mathbf{x}(t)\big) \\ &\quad - \big(1 - m(t)\big)\hat{\epsilon}_d \mathrm{sgn}(s) \\ &\quad - m(t)k_{su}(t)\mathrm{sgn}\big(s(t)\big) + d(t) \\ &= -k_d s(t) + \big(1 - m(t)\big)\hat{f}_A\big(t, \mathbf{x}(t)\big) \\ &\quad - \big(1 - m(t)\big)\hat{\epsilon}_d \mathrm{sgn}(s) \\ &\quad + m(t)\big(-k_{su}(t)\mathrm{sgn}\big(s(t)\big) - f_A\big(\mathbf{x}(t)\big) + d(t)\big) \\ &\quad + \big(1 - m(t)\big)d(t) \\ &= -k_d s(t) + \big(1 - m(t)\big)\sum_{j=1}^{N} \tilde{w}_j g_j\left(\frac{\|\mathbf{x} - \zeta_j\|}{\sigma_j}\right) \\ &\quad - \big(1 - m(t)\big)\hat{\epsilon}_d \mathrm{sgn}(s) \\ &\quad + m(t)\big(-k_{su}(t)\mathrm{sgn}\big(s(t)\big) - f\big(\mathbf{x}(t)\big)\big) \\ &\quad + \big(1 - m(t)\big)d(t) \end{aligned} \tag{27}$$

where

$$\tilde{f}_A(t,\mathbf{x}(t)) = \hat{f}_A(t,\mathbf{x}(t)) - f_A(t,\mathbf{x}(t)) = \sum_{j=1}^{N} \tilde{w}_j g_j\left(\frac{\|\mathbf{x}-\zeta_j\|}{\sigma_j}\right)$$

We now present the following stability theorem for the control law (20)–(25).

Theorem 2: Consider the nonlinear plant (19) with the adaptive control law (20)–(25), then all states in the adaptive system will remain bounded; moreover, the tracking errors will asymptotically converge to zero.

Proof: Consider the Lyapunov function candidate

$$V(t) = \frac{1}{2}\left(s(t)^2 + \frac{1}{\eta_1}\sum_{j=1}^{N}\tilde{w}_j^2 + \frac{1}{\eta_2}(\hat{\epsilon}_d - \epsilon)^2\right) \tag{28}$$

where η_1 and η_2 are constants. Differentiating $V(t)$ with respect to time and using (27) one has

$$\begin{aligned}\dot{V}(t) &= s(t)\dot{s}(t) + \frac{1}{\eta_1}\sum_{j=1}^{N}\tilde{w}_j(\dot{\tilde{w}}) + \frac{1}{\eta_2}(\hat{\epsilon}_d - \epsilon)(\dot{\hat{\epsilon}}) \\ &= -k_d s^2(t) + (1-m(t))s(t)\sum_{j=1}^{N}\tilde{w}_j g_j\left(\frac{\|\mathbf{x}-\zeta_j\|}{\sigma_j}\right) \\ &\quad - (1-m(t))\hat{\epsilon}_d|s(t)| \\ &\quad + m(t)s(t)(-k_{su}(t)\mathrm{sgn}(s(t)) - f(\mathbf{x}(t))) \\ &\quad + (1-m(t))s(t)d(t) \\ &\quad + \frac{1}{\eta_1}\sum_{j=1}^{N}\tilde{w}_j(\dot{\tilde{w}}_j) + \frac{1}{\eta_2}(\hat{\epsilon}_d - \epsilon)(\dot{\hat{\epsilon}}_d) \\ &\le -k_d s^2(t) < 0 \end{aligned} \tag{29}$$

Therefore, all signals in the system are bounded. Since $s(t)$ is uniformly bounded, it is easily shown that, if $\tilde{\mathbf{x}}(0)$ is bounded, then $\tilde{\mathbf{x}}(t)$ is also bounded for all t, and since $\mathbf{x}_d(t)$ is bounded by design, $\mathbf{x}(t)$ is as well. To complete the proof and establish asymptotic convergence of the tracking error, it is necessary to show that $s \to 0$ as $t \to \infty$. This can be accomplished by applying Barbalat's Lemma [15] to the continuous, nonnegative function:

$$V_1(t) = V(t) - \int_0^t \left(\dot{V}(\tau) + k_d s^2(\tau)\right)d\tau \text{ with}$$
$$\dot{V}_1(t) = -k_d s^2(t) \tag{30}$$

It can easily be shown that every term in (27) is bounded, hence $\dot{s}$ 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 $\dot{V}_1(t) \le 0$ for all t, use of Barbalat's lemma proves that $\dot{V}_1(t) \to 0$ and hence from (30) that $s(t) \to 0$ as $t \to \infty$. □

C. Continuous Adaptive Control Law

Since the discontinuities of $m(t)$ and $\mathrm{sgn}(s)$ result in the controller (20)–(25) to discontinuously transition between sliding and adaptive modes of operation and between the sliding surface, such a control law leads to control chattering. Chattering is undesirable in practice because it involves high control activity, and further may excite unmodeled high frequency plant dynamics. This problem can be eliminated by smoothing out the discontinuous functions in a neighborhood. To do this, we firstly choose a set A containing A_d, then defining the pure adaptive operation is restricted to the interior of the set A_d, while the pure sliding operation is restricted to the exterior of the set A; in between, in the region $A - A_d$, the two modes are effectively blended using a continuous modulation function, i.e., $m(t) = 0$ when $\mathbf{x}(t) \in A_d$, $m(t) = 1$ when $\mathbf{x}(t) \in A$, and $0 < m(t) < 1$, otherwise. Although there may be many methods of choosing a modulation function, the choice of a modulation function in this paper follows that in [18], and is briefly described in the following. Define A_d and A respectively as

$$A_d = \{\mathbf{x} \mid \|\mathbf{x}-\mathbf{x}_0\|_{p,\pi} \le 1\},$$
$$A = \{\mathbf{x} \mid \|\mathbf{x}-\mathbf{x}_0\|_{p,\pi} \le 1 + \Psi\},$$

where Ψ is a positive constant representing the width of the transition region, $\mathbf{x}_0$ fixes the absolute location of the sets in the state space of the plant, and $\|\mathbf{x}\|_{p,\pi}$ is a weighted p-norm of the form:

$$\|\mathbf{x}\|_{p,\pi} = \left\{\sum_{i=1}^{n}\left(\frac{|x_i|}{\pi_i}\right)^p\right\}^{1/p}$$

for a set of strictly positive weights $\{\pi_i\}_{i=1}^n$. With these definitions, a modulation function is chosen as

$$m(t) = \max\left(0, \mathrm{sat}\left(\frac{(r(t)-1)}{\Psi}\right)\right) \tag{31}$$

where $r(t) = \|\mathbf{x}-\mathbf{x}_0\|_{p,\pi}$. When $r(t) \le 1$, meaning that $\mathbf{x} \in A_d$, the output of the saturation function is negative, hence the maximum which defines $m(t)$ is zero, as desired. When $r(t) \ge 1 + \Psi$, corresponding to $\mathbf{x} \in A$, the saturation function is unity, hence $m(t) = 1$, again as desired. In between, for $\mathbf{x} \in A - A_d$, it is easy to check that $0 < m(t) < 1$.

To avoid the discontinuity of $\mathrm{sgn}(s)$, we use the saturation function $\mathrm{sat}(s/\phi)$ to replace $\mathrm{sgn}(s)$. The constant ϕ describes the width of a boundary layer, which is used to prevent discontinuous control transitions. With the above in mind, the adaptive control law given by (20)–(25) becomes

$$u(t) = -k_d s(t) + u_{fd}(t) + (1-m(t))u_{fu} + m(t)u_{su}(t). \tag{32}$$

$$u_{fd}(t) = x_d^{(n)}(t) - \Lambda_v^T\tilde{\mathbf{x}}(t) \tag{33}$$

$$u_{fu} = \sum_{j=1}^{N}\hat{w}_j g_j\left(\frac{\|\mathbf{x}-\zeta_j\|}{\sigma_j}\right) - \hat{\epsilon}_d \mathrm{sat}(s/\phi) \tag{34}$$

$$\dot{\hat{w}}_j = -\eta_1(1-m(t))s_\phi(t)g_j\left(\frac{\|\mathbf{x}-\zeta_j\|}{\sigma_j}\right) \tag{35}$$

$$\dot{\hat{\epsilon}}_d = \eta_2(1-m(t))|s_\phi(t)| \tag{36}$$

$$u_{su}(t) = -k_{su}(t)\mathrm{sat}(s/\phi) \tag{37}$$

where $s_\phi = (s_{\phi 1} \cdots s_{\phi n})^T$ with $s_{\phi i} = s_i - \phi \mathrm{sat}(s/\phi)$ is a measurement of the algebraic distance of the current state to the boundary layer.

The following stability theorem for the control law (32)–(37) can be proposed.

Theorem 3: Consider the nonlinear plant (19) with the adaptive control law (32)–(37). If ϕ is chosen so that $\phi \geq \epsilon/k_d$, then all states in the adaptive system will remain bounded; moreover, the tracking errors will be asymptotically bounded by:

$$|\tilde{x}^{(i)}(t)| \leq 2^i \lambda^{i-n+1}\phi, \qquad i = 0, \ldots, n-1.$$

Proof: Similar to the proof of Theorem 2, consider the Lyapunov function candidate:

$$V(t) = \frac{1}{2}\left(s_\phi(t)^2 + \frac{1}{\eta_1}\sum_{j=1}^{N}\tilde{w}_j^2 + \frac{1}{\eta_2}(\hat{\epsilon}_d - \epsilon)^2\right) \tag{38}$$

While $\dot{s}_\phi$ is not defined when $|s| = \phi$, $(d/dt)s_\phi^2$ is well defined and continuous everywhere and can be written $(d/dt)s_\phi^2 = 2s_\phi\dot{s}$. Since $s_\phi \mathrm{sat}(s/\phi) = |s_\phi|$, one has

$$\begin{aligned}\dot{V}(t) &= -\left(k_d s_\phi^2 + |s_\phi| k_d \phi\right)\\ &\quad + \left(1 - m(t)\right)s_\phi \sum_{j=1}^{N}\tilde{w}_j g_j\left(\frac{\|\mathbf{x} - \zeta_j\|}{\sigma_j}\right)\\ &\quad - \left(1 - m(t)\right)\hat{\epsilon}_d |s_\phi|\\ &\quad + m(t)\Big(-k_{su}(t)|s_\phi| - f\big(\mathbf{x}(t)\big)s_\phi\Big)\\ &\quad + \left(1 - m(t)\right)s_\phi d(t)\\ &\quad + \frac{1}{\eta_1}\sum_{j=1}^{N}\tilde{w}_j(\dot{\tilde{w}}_j) + \frac{1}{\eta_2}(\hat{\epsilon}_d - \epsilon)(\dot{\hat{\epsilon}}_d)\\ &\leq -k_d s_\phi^2 + \left(1 - m(t)\right)|s_\phi|\left(|d(t)| - \epsilon - k_d\phi\right)\\ &\quad + m(t)|s_\phi|\left(|f\big(\mathbf{x}(t)\big)| - k_d\phi - k_{su}\right).\end{aligned} \tag{39}$$

Due to $|d| \leq \epsilon$ whenever $m(t) < 1$, $k_d\phi \geq \epsilon$ and $0 \leq m(t) \geq 1$ for all $t \geq 0$, the second term on the right is nonpositive. Similarly, since $|f(\mathbf{x}(t))| < k_{su}$, the third term is also nonpositive.

Thus one has $\dot{V}(t) \leq -k_d s_\phi^2$ for all $t \geq 0$. Therefore, following the argument in the proof of Theorem 2, it can be shown that $s_\phi(t) \to 0$ as $t \to \infty$. This means that the inequality $|s(t)| \leq \phi$ is obtained asymptotically, and the asymptotic tracking errors can be shown [20] to be asymptotically bounded by

$$|\tilde{x}^{(i)}(t)| \leq 2^i \lambda^{i-n+1}\phi, \qquad i = 0, \ldots, n-1.$$

□

Remarks:

1) The tracking errors depend on the choice of ϕ. If extreme tracking accuracy is required, the resulting choice of the ϕ might produce a sliding controller with a boundary layer so thin it risks exciting high frequency dynamics when this component is active. This suggests that the value of ϕ used, and hence the resulting tracking accuracy, must arise as a trade-off between the frequency range of the unmodeled dynamics and the trajectory following requirements. As suggested in [20], an effective way to implement such a trade-off is to actually let ϕ vary depending on the state location. In the current situation, since the sliding component is used mainly as a stabilizing influence during the initial learning phases, one could simply introduce the flexibility of letting ϕ vary between the small value ϵ/k_d when the fuzzy approximation is effective, and a large value $D(\mathbf{x})/k_d$ when it is not.
2) The transition between adaptive and robust operation is necessary given the limitations on the approximating abilities of the adaptive controllers examined in this paper. The above comments which apply to ϕ also apply to Ψ: the transition region should be thick enough that there is no possibility that transition between the adaptive and sliding operation might be so sudden as to excite unmodeled dynamics.

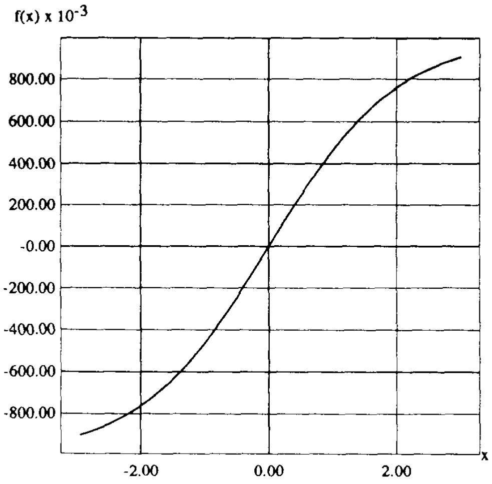

Fig. 3. Curve of $f(x)$.

V. Simulations

In this section, we apply the adaptive fuzzy controller developed in the last section to control a system given in [25].

$$\dot{x} = \frac{1 - e^{-x(t)}}{1 + e^{-x(t)}} + u(t) \tag{40}$$

Without control, i.e., $u(t) = 0$, Fig. 3 shows that the system is unstable, because of $\dot{x} = \frac{1-e^{-x(t)}}{1+e^{-x(t)}} > 0$ for $x > 0$, and $\dot{x} = \frac{1-e^{-x(t)}}{1+e^{-x(t)}} < 0$ for $x < 0$. The objective is to control the system state x to the origin; i.e., $x_d = 0$. The set A_d was chosen to be $[-3, 3]$ interval with respect to the weighted infinity norm $\|\mathbf{x}\|_{\infty,\pi} = |x|/3$. A thin transition region between the adaptive and sliding operation was chosen to a value of $\Psi = 0.05$ so that $A = \{x \mid \|\mathbf{x}\|_{\infty,\pi} \leq 1.05\}$.

To gauge the effectiveness of the adaptive fuzzy component of the control law, it is useful to compare the control performance of the closed loop system both with and without the output of the adaptive fuzzy approximation. Since the

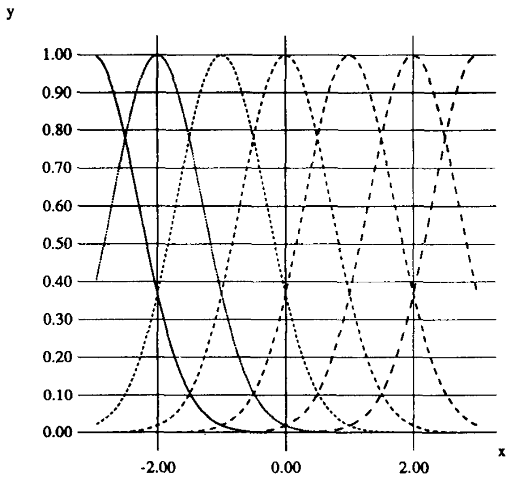

Fig. 4. Fuzzy membership functions.

use of the sliding controller is sufficient to keep the tracking error bounded, even without the adaptive contribution, these simulations can be conducted without fear of instability. We simulate two cases: 1) without linguistic descriptions about f, and 2) with the following linguistic descriptions

$$R_f^k : IFx \text{ is near } k, \text{ THEN is near } B_k$$

where *near k*, $k = -3, -2, -1, 0, 1, 2, 3$, is a fuzzy set with membership functions $\mu_k(x) = \exp\left(-(x-k)^2\right)$, which are shown in Fig. 4. B_k are obtained by evaluating f at points $x = -3, -2, -1, 0, 1, 2, 3$. The values of B_k is not required here since the exact w_j^* is not required in the control law. However, with the knowledge of B_k it will be helpful for the choice of initial $\hat{w}_j(0)$ to speed up the adaptation process. In this example, the initial $\hat{w}_j(0)$ are selected as $\hat{w}_1(0) = -0.8$, $\hat{w}_2(0) = -0.6$, $\hat{w}_4(0) = -0.4$, $\hat{w}_5(0) = 0$, $\hat{w}_6(0) = 0.4$, $\hat{w}_7(0) = 0.6$, $\hat{w}_8(0) = 0.8$.

In the simulations, the control law (32)–(37) was used taking $k_d = 10$, the parameters w_j are adjusted according to (35) with $\eta_1 = 20$ and ϵ is adjusted according to (36) with $\eta_2 = 0.5$. The initial $\epsilon(0)$ is selected as $\epsilon(0) = 1$. Since the nonlinearity f is uniformly upper bounded on A_d^c, satisfying $D(x) = 1 \geq \frac{1-e^{-x(t)}}{1+e^{-x(t)}}$, the gain $k_{su} = 1$ is used in the sliding controller with boundary $\phi = 0.05$. The initial state is chosen as $x(0) = 2$.

Two simulations were conducted, one in which the complete control law (32)–(37), including the adaptive fuzzy rules, was employed (solid curve), and one in which the adaptive fuzzy control (u_{fu}) was not used (dotted curve), where the control gains are unchanged. Fig. 5 shows the tracking performance and Fig. 6 shows the control input. We see from Fig. 5 that 1) the control without adaptive fuzzy approximation could regulate the plant to the origin and 2) by incorporating the fuzzy rules, the speed of convergence becomes much faster. We also simulated for other initial conditions, and the results were very similar; we do not show them in order to keep clear the figures and make comparison easier. Therefore,

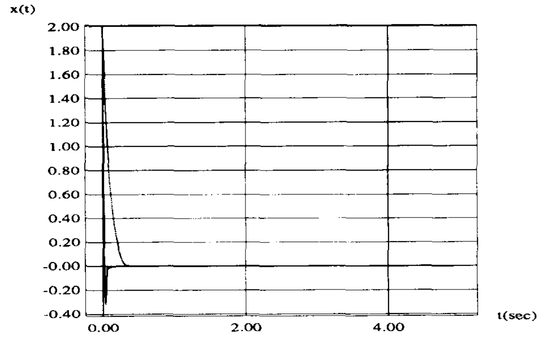

Fig. 5. Closed-loop state $x(t)$ using the control law (32)–(37) with adaptive fuzzy approximation u_{fu} (solid line) and without adaptive fuzzy approximation u_{fu} (dotted line).

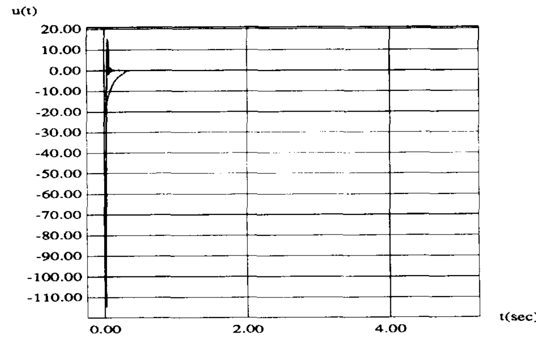

Fig. 6. Control $u(t)$ using the control law (32)–(37) (solid line) with adaptive fuzzy approximation u_{fu} (solid line) and without adaptive fuzzy approximation u_{fu} (dotted line).

incorporating the linguistic fuzzy information clearly results in superior tracking performance.

VI. Conclusion

In this paper, an adaptive tracking control architecture is proposed for a class of continuous-time nonlinear dynamic systems for which an explicit linear parameterization of the uncertainty in the dynamics is either unknown or impossible. The developed controller is capable of incorporating fuzzy IF-THEN rules into the controller and guarantees the global stability of the resulting closed-loop system in the sense that all signals involved are uniformly bounded. Simulation results show that incorporating the linguistic fuzzy information into controllers clearly results in superior tracking performance. Therefore, the proposed control method provides a tool for making use of the fuzzy information in a systematic and efficient manner.

References

[1] C. G. Atkeson, "Memory-based approaches to approximating continuous functions," in *Proc. 6th Yale Workshop on Adaptive and Learning Systems*, Yale University, New Haven, CT, 1990, pp. 212–217.

[2] A. G. Barto, R. S. Sutton, and C. W. Anderson, "Neuronlike adaptive elements that can solve difficult learning control problems," *IEEE Trans. Syst., Man, Cybern.*, vol. 13, pp. 834–846, 1983.

[3] F. Girosi and G. Anzelloti, "Rates of convergence of a approximation by translates," *Artific. Intellig. Lab. Memo. No. 1288*, MIT, Cambridge, MA, Feb., 1992.

[4] E. J. Hartman, J. D. Keeler, and J. M. Kowalski, "Layered neural networks with Gaussian hidden units as universal approximations," *Neural Computat.*, vol. 2, pp. 210–215, 1990.

[5] S. Horiawa, T. Furuhashi, S. Ouma, and Y. Uchikawa, "A fuzzy controller using a neural network and its capability to learn expert's control rules," in *Proc. Int. Conf. Fuzzy Logic and Neural Networks*, Japan, 1990, pp. 103–106.

[6] A. Isidori, *Nonlinear Control Systems: An Introduction*, 2nd ed. Berlin: Springer-Verlag, 1989.

[7] J.-S. R. Jang and C.-T. Sun, "Functional Equivalence between radial basis function networks and fuzzy inference systems," *IEEE Trans. Neural Networks*, vol. 4, pp. 156–159, 1993.

[8] J. Kiszka, M. Gupta, and P. Nikiforuk, "Eergetistic stability of fuzzy dynamic systems" *IEEE Trans. Syst., Man, Cybern.*, vol. 15, pp. 783–792, 1985.

[9] B. Kosko, *Neural Networks and Fuzzy Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1991.

[10] ——, "Fuzzy systems as universal approximators," in *Proc. IEEE Int. Conf. on Fuzzy Systems*, San Diego, CA, March, 1992, pp. 1153–1162.

[11] G. Langari and M. Tomizuka, "Stability of fuzzy linguistic control systems," in *Proc. IEEE Conf. on Decision and Contr.*, 1990, pp. 2185–2190.

[12] C. C. Lee, "Fuzzy logic in control systems: Fuzzy logic controller, Part I, part II," *IEEE Trans. Syst., Man, Cybern.*, vol. 20, pp. 404–435, 1990.

[13] R. P. Lippman, "Pattern classification using neural networks," *IEEE Commun. Mag.*, vol. 27, pp. 47–64, 1989.

[14] T. Poggio and F. Girosi, "Networks for approximation and learning," *Proc. IEEE*, vol. 78, pp. 1481–1479, 1990.

[15] V. Popov, *Hyperstability of control system*. Berlin: Springer-Verlag, 1973.

[16] W. Rudin, *Principles of Mathematical Analysis*. New York: McGraw-Hill, 1976.

[17] D. E. Rumelhart and J. L. McClelland, *Parallel Distributed Processing, Explorations in the Microstructure of Cognition*. Cambridge, MA: MIT Press, 1986, vol. 1.

[18] R. M. Sanner and J. J. E. Slotine, "Gaussian networks for direct adaptive control," *IEEE Trans. Neural Networks*, vol. 3, pp. 837–863, 1992.

[19] R. M. Sanner and J. J. E. Slotine, "Irregular sampling theory and scattered-center radial Gaussian networks," in *Proc. 31st Conf. on Decision and Contr.*, 1992, pp. 13–19.

[20] J.-J. E. Slotine and J. A. Coetsee, "Adaptive sliding controller synthesis for nonlinear systems," *Int. J. Contr.*, vol. 43, pp. 1631–1651, 1986.

[21] M. Sugeno and T. Yasukawa, "A fuzzy-logic-based approach to qualitative modeling," *IEEE Trans. Fuzzy Syst.*, vol. 1, pp. 7–31, 1993.

[22] T. Takagi and I. Hayashi, "NN-deriven fuzzy reasoning," *Int. J. Approx. Reason.*, vol. 5, pp. 191–212, 1991.

[23] V. I. Utkin, "Variable structure systems with sliding mode: A survey," *IEEE Trans. Automatic Contr.*, vol. 22, p. 212, 1977.

[24] L.-X. Wang and J. M. Mendel, "Fuzzy basis functions, universal approximation, and orthogonal least-squares learning," *IEEE Trans. Neural Networks*, vol. 3, pp. 807–814, 1992.

[25] L.-X. Wang, "Stable adaptive fuzzy control of nonlinear system," *IEEE Trans. Fuzzy Syst.*, vol. 1, pp. 146–155, 1993.

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