Stable Adaptive Control Using Fuzzy Systems and Neural Networks

Jeffrey T. Spooner and Kevin M. Passino, *Senior Member, IEEE*

***Abstract*— Stable direct and indirect adaptive controllers are presented which use Takagi–Sugeno fuzzy systems, conventional fuzzy systems, or a class of neural networks to provide asymptotic tracking of a reference signal for a class of continuous-time nonlinear plants with poorly understood dynamics. The indirect adaptive scheme allows for the inclusion of *a priori* knowledge about the plant dynamics in terms of exact mathematical equations or linguistics while the direct adaptive scheme allows for the incorporation of such *a priori* knowledge in specifying the controller. We prove that with or without such knowledge both adaptive schemes can "learn" how to control the plant, provide for bounded internal signals, and achieve asymptotically stable tracking of a reference input. In addition, for the direct adaptive scheme a technique is presented in which linguistic knowledge of the inverse dynamics of the plant may be used to accelerate adaptation. The performance of the indirect and direct adaptive schemes is demonstrated through the longitudinal control of an automobile within an automated lane.**

I. Introduction

FUZZY controllers have stirred a great deal of excitement in some circles since they allow for the simple inclusion of heuristic knowledge about how to control a plant rather than requiring exact mathematical models. This can sometimes lead to good controller designs in a very short period of time. In situations where heuristics do not provide enough information to specify all the parameters of the fuzzy controller *a priori*, researchers have introduced adaptive schemes that use data gathered during the on-line operation of the controller, and special adaptation heuristics, to automatically learn these parameters (see e.g., [1]–[13] and the references therein). To date, stability conditions have not been provided for any of the approaches in [1]–[13], but Langari and Tomizuka [14] and others have developed stability conditions for (nonadaptive) fuzzy controllers and recently several stable adaptive fuzzy control schemes have been introduced [15]–[18]. Moreover, closely related neural control approaches have been studied [19]–[24].

In this paper, we seek to introduce adaptive fuzzy or neural control approaches that are guaranteed to operate properly under less restrictive assumptions and for more general continuous-time nonlinear systems. In particular, we first introduce an "indirect adaptive controller"[1] in which fuzzy systems or neural networks are used to estimate the plant dynamics, and then use these estimates to generate controls that achieve asymptotic tracking of a reference input. Work on the use of fuzzy systems and neural networks for identification has been performed in [15], [19], and [26]. Indirect adaptive controllers based on neural network radial-basis functions and standard fuzzy systems have both been shown to provide asymptotic tracking of a reference signal for a class of continuous time nonlinear plants with no zero dynamics, provided that the error in representing the nonlinear plant dynamics with neural networks or fuzzy systems converges to zero [15], [20]. A scheme was presented in [22] for a similar class of plants which uses modified Hebbian learning rules. In theory, it is possible to exactly represent the dynamics of a large class of nonlinear plants using standard fuzzy systems or radial-basis functions. Unfortunately, this may require the use of a very large, or infinite, number of rules (for fuzzy systems) or nodes (for neural networks), limiting the applicability of the techniques in [15] and [20]. In particular, in [15] and [20], the authors represent the error between the actual plant dynamics and the fuzzy estimation by a term $w(t)$. Convergence of the tracking error to zero is guaranteed by assuming that $w(t)$ is square integrable. This, however, is difficult to show for any given plant (indeed, even for some very simple nonlinear plants such as a tank-level control problem, the assumption fails to hold). In addition, this calculation may require an exact model of the plant, which defeats the purpose of using a "model-free" technique. Within this work, we simply require knowledge of the plant relative degree and bounds on the plant dynamics.

For our indirect adaptive controller, we take advantage of robustness properties associated with sliding mode techniques [27] to ensure that the tracking errors will asymptotically converge to zero even if there are approximation errors between the identifier model and plant. We also show that the control signal may be smoothed, allowing for stable operation and tracking convergence to an ϵ-neighborhood of zero. Our indirect adaptive scheme allows for the use of a combination of standard fuzzy systems, Takagi–Sugeno fuzzy systems, and

Manuscript received February 15, 1995; revised March 12, 1996. J. T. Spooner was supported by a CITR-OSU Fellowship. This work was supported in part by The Center for Intelligent Transportation Research (CITR) at The Ohio State University and National Science Foundation Grants IRI-9210332 and EEC-9315257.

J. T. Spooner was with the Department of Electrical Engineering, Ohio State University, Columbus, OH 43210 USA. He is currently with Sandia National Laboratories, Albuquerque, NM 87185 USA.

K. M. Passino is with the Department of Electrical Engineering, The Ohio State University, Columbus, OH 43210 USA.

Publisher Item Identifier S 1063-6706(96)05627-5.

[1] Indirect adaptive control uses an "identifier" to synthesize a model of the plant dynamics and then information from this model is used to tune a controller (we say that the controller was tuned "indirectly" by first identifying a model of the plant). For "direct adaptive control," an identifier is not used for the plant; the parameters of the controller are tuned directly (some think of the direct adaptive controller as a "controller identifier"). For more details see, e.g., [25].

neural networks. If knowledge of the plant is available either in the form of linguistics or mathematical formulations, this information may be incorporated into the indirect adaptive scheme to accelerate tracking convergence. The combination of Takagi–Sugeno fuzzy systems and the ability to incorporate knowledge of the plant dynamics provides a great deal of design flexibility.

Though this paper is developed through the same general philosophy as many of the cited papers, our indirect adaptive controller does possess the following distinctions: i) the results of the stability theory may be applied to a class of plants with zero dynamics (this was not done in [15], [20], [22]), ii) this paper ensures asymptotic tracking convergence using a larger class of fuzzy systems (i.e., Takagi–Sugeno fuzzy systems) than in [15] and a larger class of neural networks (i.e., those with a second hidden layer) than in [20], iii) within this paper we allow for the direct inclusion of a mathematical description of the known part of the plant dynamics (this was not done in [15], [20], and [22]), iv) unlike [15], [20], and [22], this paper uses a manifold to develop an error measurement which allows for asymptotic stability of the output error even if the modeling error does not go to zero, and v) tracking convergence is guaranteed to a boundary layer of zero using the smoothed version of the control law.

A direct adaptive controller is then introduced which attempts to directly adjust the parameters of a fuzzy or neural controller to achieve asymptotic tracking of a reference input. Within [16] a stable direct adaptive control scheme based on standard fuzzy systems was presented for a class of plants with constant input gain and no zero dynamics. Asymptotic tracking convergence was proven for this scheme if a certain approximation error is square integrable. The direct adaptive scheme in [16] thus has the same deficiencies as in [15] and [20] by requiring square integrability of the approximation error. It was shown that radial basis neural networks and standard fuzzy systems may provide asymptotic tracking of a reference signal for a class of nonlinear plants, even if the estimation error is not square integrable in [17] and [21]. Within this paper, we present a direct adaptive scheme which uses standard fuzzy systems, Takagi–Sugeno fuzzy systems, or neural networks to achieve stable tracking of a reference input for a class of plants with zero dynamics and a state-dependent input gain. If knowledge of how to design the controller is available either in the form of linguistics or mathematical equations, this information may be incorporated into the direct adaptive scheme to accelerate convergence. Our direct adaptive scheme also allows for the inclusion of linguistic knowledge of the plant inverse dynamics to accelerate tracking convergence, and a control smoothing scheme may be used to reduce the control action while maintaining closed-loop stability.

As with the indirect adaptive scheme, our direct adaptive scheme has many differences from the existing techniques (i.e., those presented in [16], [17], and [21]). Particularly, i) the stability results presented here may be applied to systems with a state-dependent input gain, whereas [16] and [21] consider a class of nonlinear plants with constant input gain, and [17] only considers the special case of unity gain, ii) none of the results in [16], [17], and [21], considered systems containing zero dynamics, iii) unlike [16], our direct adaptive algorithm ensures that even if the approximation error is not square integrable, then the tracking error will go to zero (or to an ϵ-boundary layer of zero for the smoothed control version), iv) our direct adaptive controller allows for Takagi–Sugeno fuzzy systems, standard fuzzy systems, or neural networks, v) the direct adaptive technique presented here allows for the inclusion of a known controller u_k so that it may be used to either enhance the performance of some prespecified controller or stand alone as a stable adaptive controller, and vi) furthermore, our approach allows for the incorporation of heuristics about the inverse plant dynamics to speed adaptation. We illustrate the design of both indirect and direct adaptive controllers for the longitudinal control of a vehicle within an automated lane (it seems that adaptive fuzzy control has not yet been used for this application).

It should be mentioned that other work has been completed in combining conventional stable adaptive control and intelligent control. Within [18] a nonlinear discrete-time plant is represented by a linear regression form using Takagi–Sugeno fuzzy systems to provide global stability. A discrete time adaptive routine is presented in [23] which uses layered neural networks to provide stable adaptive tracking provided some initialization conditions are satisfied. Finally, in [24], a new adaptive routine using dynamic neural networks is presented with stability investigated using a singular perturbation model of the plant [28].

This paper is organized as follows. In Section II, we define a class of Takagi–Sugeno fuzzy systems and show that a large class of fuzzy systems and neural networks may be represented using the same functional form. Sections III and IV present the indirect and direct adaptive schemes and the stability proofs. In Section V, we illustrate the concepts on the longitudinal control of a vehicle in an automated lane. Section VI contains the concluding remarks where we discuss both the advantages and disadvantages of the adaptive schemes. Note that this paper expands on the work done in [29]–[32].

II. Fuzzy Systems and Neural Networks

In this section, we define the Takagi–Sugeno fuzzy system and show that a class of standard fuzzy systems[2] and some neural networks are a special case of this model.

A. Takagi–Sugeno Fuzzy Systems

A multiple-input single-output (MISO) fuzzy system is a nonlinear mapping from an input vector $X = [x_1, x_2, \cdots, x_n]^T \in \Re^n$ (T denotes transpose) to an output $\tilde{y} = \tilde{f}(X) \in \Re$ (note that we use X as a general-vector input to the fuzzy system; it may or may not be the same as the "state" that is used in all the later sections). Using the Takagi–Sugeno model [34], the fuzzy system is characterized by a set of p **If-Then** rules stored in a rule-base

[2] It is assumed that the reader has some familiarity with fuzzy systems. For an introduction, see [5], [15], and [33]

TABLE I
SOME STANDARD MEMBERSHIP FUNCTIONS

	Triangular	Gaussian
Left	$\mu(x) = \begin{cases} 1 & \text{if } x \le c \\ \max\left(0, 1 + \frac{c-x}{w}\right) & \text{otherwise} \end{cases}$	$\mu(x) = \begin{cases} 1 & \text{if } x \le c \\ \exp\left(-\left(\frac{x-c}{\sigma}\right)^2\right) & \text{otherwise} \end{cases}$
Centers	$\mu(x) = \begin{cases} \max\left(0, 1 + \frac{x-c}{w}\right) & \text{if } x \le c \\ \max\left(0, 1 + \frac{c-x}{w}\right) & \text{otherwise} \end{cases}$	$\mu(x) = \exp\left(-\left(\frac{x-c}{\sigma}\right)^2\right)$
Right	$\mu(x) = \begin{cases} \max\left(0, 1 + \frac{x-c}{w}\right) & \text{if } x \le c \\ 1 & \text{otherwise} \end{cases}$	$\mu(x) = \begin{cases} \exp\left(-\left(\frac{x-c}{\sigma}\right)^2\right) & \text{if } x \le c \\ 1 & \text{otherwise} \end{cases}$

and expressed as

$$\begin{array}{lll} R_1: & \textbf{If} & (\tilde{x}_1 \text{ is } \tilde{F}_1^i \textbf{ and } \cdots \textbf{ and } \tilde{x}_n \text{ is } \tilde{F}_n^j) \\ & \textbf{Then} & c_1 = g_1(X) \\ \vdots & & \vdots \\ R_p: & \textbf{If} & (\tilde{x}_1 \text{ is } \tilde{F}_1^k \textbf{ and } \cdots \textbf{ and } \tilde{x}_n \text{ is } \tilde{F}_n^l) \\ & \textbf{Then} & c_p = g_p(X). \end{array}$$

Here, $\tilde{F}_b^a$ is the ath linguistic value associated with the linguistic variable $\tilde{x}_b$ that describes input x_b, and $c_q = g_q(X)$ is the consequence of the qth rule and g_q: $\Re^n \to \Re$. Using fuzzy set theory, the rule-base is expressed as

$$\begin{array}{lll} R_1: & \textbf{If} & (F_1^i \textbf{ and } \cdots \textbf{ and } F_n^j) \\ & \textbf{Then} & c_1 = g_1(X) \\ \vdots & & \vdots \\ R_p: & \textbf{If} & (F_1^k \textbf{ and } \cdots \textbf{ and } F_n^l) \\ & \textbf{Then} & c_p = g_p(X) \end{array}$$

where F_b^a is a fuzzy set defined by

$$F_b^a := \{(x_b, \mu_{F_b^a}(x_b)):\ x_b \in \Re\}. \tag{1}$$

The membership function $\mu_{F_b^a} \in [0, 1]$ quantifies how well the linguistic variable $\tilde{x}_b$ that represents x_b is described by the linguistic value $\tilde{F}_b^a$. There are many ways to define membership functions [15]. For instance, Table I specifies triangular membership functions with "center" c and "width" w, and it specifies Gaussian membership functions with "center" c and "width" σ (see Figs. 6 and 10 in Section V for graphical representations).

The antecedent fuzzy set $F_1 \times F_2 \times \cdots \times F_n$ (fuzzy Cartesian product), of each rule is quantified by the "t-norm" [15] which may be defined by, for example, the min-operator or the product-operator

$$\mu_{F_1 \times \cdots \times F_n}(x_1, \cdots, x_n) := \min\{\mu_{F_1}(x_1), \cdots, \mu_{F_n}(x_n)\} \tag{2}$$

or

$$\mu_{F_1 \times \cdots \times F_n}(x_1, \cdots, x_n) := \mu_{F_1}(x_1) \cdot \cdots \cdot \mu_{F_n}(x_n) \tag{3}$$

respectively (notice that for convenience, we have removed the superscripts from the F_b^a). Using singleton fuzzification, defuzzification may be obtained using

$$\tilde{y} = \tilde{f}(X) = \frac{\sum_{i=1}^{p} c_i \mu_i}{\sum_{i=1}^{p} \mu_i} \tag{4}$$

where $\mu_i := \mu_{F_1 \times \cdots \times F_n}(x_1, \cdots, x_n)$ is the value that the membership function [defined via (2) or (3)] for the antecedent of the ith rule takes on at $X = [x_1, \cdots, x_n]^T$. It is assumed that the fuzzy system is defined so that for all $X \in \Re^n$, we have $\sum_{i=1}^{p} \mu_i \neq 0$. We may express (4) equivalently as

$$\tilde{y} = c^T \zeta = \tilde{f}(X) \tag{5}$$

where $c^T := [c_1 \cdots c_p]$ and $\zeta^T := [\mu_1 \cdots \mu_p]/[\sum_{i=1}^{p} \mu_i]$. We assume that $\tilde{f}$, the mapping produced by the fuzzy system, is Lipschitz continuous [25].

In this paper, the output consequences for each rule are taken as a linear combination of a set of Lipschitz continuous functions $\theta_k(X) \in \Re$, $k = 1, 2, \cdots, m-1$, so that

$$\begin{aligned} c_i &= g_i(X) \\ &:= a_{i,0} + a_{i,1}\theta_1(X) + \cdots \\ &\quad + a_{i,m-2}\theta_{m-2}(X) + a_{i,m-1}\theta_{m-1}(X) \end{aligned} \tag{6}$$

$i = 1, \cdots, p$. Define the following:

$$z := \begin{bmatrix} 1 \\ \theta_1(X) \\ \vdots \\ \theta_{m-1}(X) \end{bmatrix} \in \Re^m \tag{7}$$

and

$$A^T := \begin{bmatrix} a_{1,0} & a_{1,1} & \cdots & a_{1,m-1} \\ a_{2,0} & a_{2,1} & \cdots & a_{2,m-1} \\ \vdots & \vdots & \ddots & \vdots \\ a_{p,0} & a_{p,1} & \cdots & a_{p,m-1} \end{bmatrix}. \tag{8}$$

The consequence vector associated with the fuzzy rules is now given by $c = A^T z$, so that the output of the fuzzy system may now be expressed as

$$\tilde{y} = z^T A \zeta. \tag{9}$$

Clearly, (9) is a special form of a Takagi–Sugeno fuzzy system.

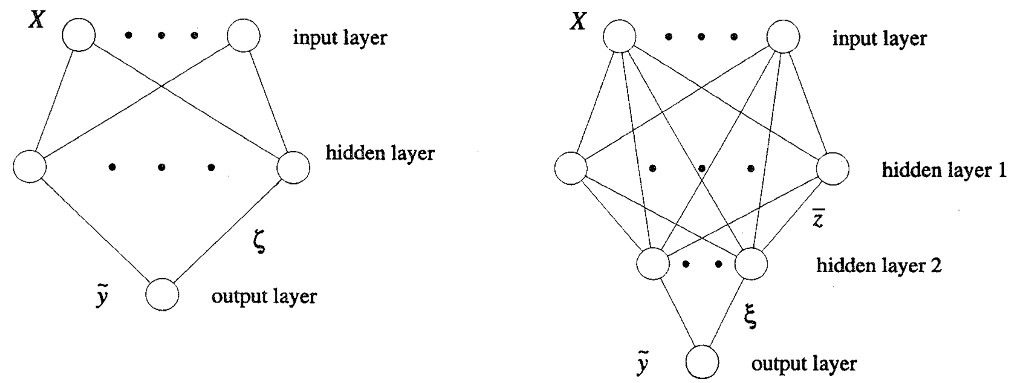


Fig. 1. Two types of neural networks which may be used with the adaptive techniques.

B. Standard Fuzzy Systems

Standard fuzzy systems naturally allow for the inclusion of heuristics into controller design. In standard fuzzy control, the output of a fuzzy system may be found using the center of gravity operation, which for a wide class of fuzzy systems is expressed as

$$\tilde{y} = \frac{\sum_{i=1}^{p} c_i \xi_i}{\sum_{i=1}^{p} \xi_i} \tag{10}$$

where c_i is the center of the output membership function associated with the ith rule, and ξ_i is the area of the implied membership function associated with the ith rule (i.e., ξ_i is the area of the output membership function that is modified via the fuzzy implication that represents the ith rule). This fits the form of (9) with $z = [1]$, $A = [c_1 \ \cdots \ c_p]$, and $\zeta_i = \xi_i / \sum_{i=1}^{p} \xi_i$ so that this standard fuzzy system is a special case of the Takagi–Sugeno fuzzy system defined by (9). Other standard fuzzy systems such as those that use centroid defuzzification will also fit the form of (9).

C. Neural Networks

Our framework allows for the use of neural networks in which a single hidden layer of radial-basis functions are used or if a special form of two hidden layers is used. Fig. 1 demonstrates these two cases. With a single hidden layer of radial basis functions the output of neural network is given by

$$\tilde{y} = c^T \zeta \tag{11}$$

where $\zeta \in \Re^p$ are (possibly normalized) radial-basis functions (e.g., squashing functions characterized by Gaussian functions [35]) and c^T is a vector of adapting weights. This type of system may be described by (9) with $z = [1]$ and $A = c^T$. As it is well-recognized in the literature, this is exactly the same representation as used with standard fuzzy systems [15].

A second type of neural network considered in this paper is one in which there are two hidden layers with the second hidden layer of a special form. The output of the first hidden layer produces a vector of functions

$$\overline{z} = [\theta_1 \ \cdots \ \theta_m]. \tag{12}$$

The nodes which make up the first hidden layer may be normalized radial-basis functions, squashing functions or any other standard neural-basis function [35]. Here, we allow both the output of the first hidden layer and the original input to be passed to the second hidden layer (see Fig. 1). The output of the ith node of the second hidden layer is given by

$$\xi_i = \zeta_i(\overline{z}, X) \left(b_{i,0} + \sum_{j=1}^{m} b_{i,j}\theta_j + \sum_{j=1}^{n} b_{i,j+m} x_j \right) \tag{13}$$

where $\zeta_i(\overline{z}, X)$ are squashing functions or radial-basis functions (which may be normalized) and $b_{i,0}$ is the bias for ith node. The output of the neural network is taken as a linear combination of the outputs of the second hidden layer; that is

$$\tilde{y} = \sum_{j=1}^{p} c_i \xi_i. \tag{14}$$

We may combine (13) and (14) to obtain

$$\tilde{y} = \sum_{j=1}^{p} \zeta_i(z, X) \left(a_{i,0} + \sum_{j=1}^{m} a_{i,j}\theta_j + \sum_{j=1}^{n} a_{i,j+m} x_j \right) \tag{15}$$

which may be expressed in the form of (9) with $z = [1 \ \theta_1 \ \cdots \ \theta_m x_1 \ \cdots \ x_n]^T$, and $A = [a_{i,j}]$ with $a_{i,j} = c_i b_{i,j}$. Note that z may or may not include any θ_i or x_i.

Within the adaptive framework to follow, we shall typically refer to Takagi–Sugeno fuzzy systems within our discussion. However, any of the above fuzzy or neural network systems apply.

III. Indirect Adaptive Control

Our objective is to design a control system which will cause the output of a relative degree r plant, y_p, to track a desired output trajectory, y_m (a relative degree r plant is one in which the plant input appears in the output dynamics after r differentiations of the output). The desired output trajectory may be defined by a signal external to the control system so

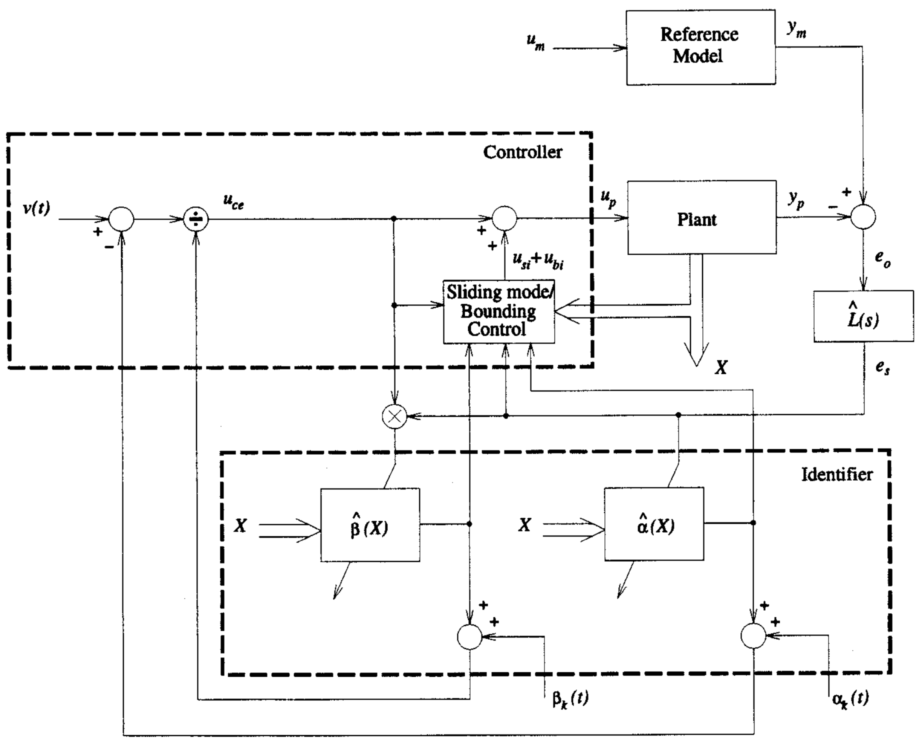


Fig. 2. An indirect adaptive fuzzy control system with a reference model.

that the first r derivatives of y_m may be measured, or by a reference model, with relative degree greater than or equal to r which characterizes the desired performance (see Fig. 2). With these considerations, we make the following assumption about the reference signal [let $y_m^{(r)}$ denote the rth derivative of y_m with respect to time].

R1) Reference Input Assumption: The desired output trajectory and its derivatives $y_m, \cdots, y_m^{(r)}$ are measurable and bounded.

Within this section, we use an "output error indirect adaptive controller," as shown in Fig. 2 (using the terminology from [25]) where an identifier seeks to approximate the plant dynamics and uses this to tune the parameters of a controller so that y_p follows y_m, and hence, $e_o = y_m - y_p \to 0$. Next, we describe each component of Fig. 2.

Here, we consider the SISO plant

$$\dot{X} = f(X) + g(X)u_p \tag{16}$$

$$y_p = h(X) \tag{17}$$

where $X \in \Re^n$ is the state vector, $u_p \in \Re$ is the input, $y_p \in \Re$ is the output of the plant and functions $f(X), g(X) \in \Re^n$, and $h(X) \in \Re$ are smooth. If the system has "strong relative degree" r then

$$\begin{aligned} \dot{\xi}_1 &= \xi_2 = L_f h(X) \\ &\vdots \\ \dot{\xi}_{r-1} &= \xi_r = L_f^{r-1} h(X) \\ \dot{\xi}_r &= L_f^r h(X) + L_g L_f^{r-1} h(X) u_p \end{aligned} \tag{18}$$

with $\xi_1 = y_p$, which may be rewritten as

$$y_p^{(r)} = [\alpha_k(t) + \alpha(X)] + [\beta_k(t) + \beta(X)]u_p \tag{19}$$

where $L_g^r h(X)$ is the rth Lie derivative of $h(X)$ with respect to g $\{L_g h(X) = (\partial h/\partial X) g(X)$ and, e.g., $L_g^2 h(X) = L_g[L_g h(X)]\}$; and it is assumed that for some $\beta_0 > 0$, we have $|\beta_k(t) + \beta(X)| \geq \beta_0$ so that it is bounded away from zero (for convenience we assume that $\beta_k(t) + \beta(X) > 0$, however, the following analysis may easily be modified for systems which are defined with $\beta_k(t) + \beta(X) < 0$). We will assume that $\alpha_k(t)$ and $\beta_k(t)$ are known components of the dynamics of the plant (that may depend on the state) or known exogenous time dependent signals and that $\alpha(X)$ and $\beta(X)$ represent nonlinear dynamics of the plant that are unknown. It is assumed that if X is a bounded state vector, then $\alpha_k(t)$ and $\beta_k(t)$ are bounded signals. Throughout the analysis to follow, both $\alpha_k(t)$ and $\beta_k(t)$ may be set to zero for all $t \geq 0$.

We shall approximate the functions $\alpha(X)$ and $\beta(X)$ with fuzzy systems (neural networks) $\tilde{y}_\alpha = \tilde{f}_\alpha(X) = z_\alpha^T A_\alpha \zeta_\alpha$ and $\tilde{y}_\beta = \tilde{f}_\beta(X) = z_\beta^T A_\beta \zeta_\beta$ by adjusting the A_α and A_β. The parameter matrices A_α and A_β are assumed to be defined within the compact parameter sets Ω_α and Ω_β, respectively. In addition, we define the subspace $S_x \subseteq \Re^n$ as the space through which the state trajectory may travel under closed-loop control (we are making no *a priori* assumptions here about the size of S_x; later, we will specify a control law that will place an explicit bound on S_x). Notice that

$$\alpha(X) = z_\alpha^T A_\alpha^* \zeta_\alpha + d_\alpha(X) \tag{20}$$

$$\beta(X) = z_\beta^T A_\beta^* \zeta_\beta + d_\beta(X) \tag{21}$$

where

$$A^*_\alpha \in \Re^{m_\alpha \times p_\alpha}$$
$$A^*_\alpha := \arg \min_{A_\alpha \in \Omega_\alpha} \left[\sup_{X \in S_x} |z^T_\alpha A_\alpha \zeta_\alpha - \alpha(X)| \right] \quad (22)$$
$$A^*_\beta \in \Re^{m_\beta \times p_\beta}$$
$$A^*_\beta := \arg \min_{A_\alpha \in \Omega_\beta} \left[\sup_{X \in S_x} |z^T_\beta A_\beta \zeta_\beta - \beta(X)| \right] \quad (23)$$

so that $d_\alpha(X)$ and $d_\beta(X)$ are approximation errors which arise when $\alpha(X)$ and $\beta(X)$ are represented by fuzzy systems. We assume that $D_\alpha(X) \geq |d_\alpha(X)|$, and $D_\beta(X) \geq |d_\beta(X)|$ where $D_\alpha(X)$ and $D_\beta(X)$ are known bounds on the error in representing the actual system with fuzzy systems. Since fuzzy systems are "universal approximators" (see [15]) both $|d_\alpha(X)|$ and $|d_\beta(X)|$ may be made arbitrarily small by a proper choice of the fuzzy system if $\alpha(X)$ and $\beta(X)$ are smooth (of course this may require an arbitrarily large number of rules). It is important to keep in mind that $D_\alpha(X)$ and $D_\beta(X)$ represent the magnitude of error between the actual nonlinear functions describing the system dynamics and the fuzzy systems when the "best" parameters are used within the fuzzy systems.

The fuzzy system approximations of $\alpha(X)$ and $\beta(X)$ of the actual system are

$$\hat{\alpha}(X) = z^T_\alpha A_\alpha \zeta_\alpha \quad (24)$$
$$\hat{\beta}(X) = z^T_\beta A_\beta \zeta_\beta \quad (25)$$

where the matrices $A_\alpha(t)$ and $A_\beta(t)$ are updated on line as shown in Fig. 2. The parameter error matrices

$$\Phi_\alpha(t) = A_\alpha(t) - A^*_\alpha \quad (26)$$
$$\Phi_\beta(t) = A_\beta(t) - A^*_\beta \quad (27)$$

are used to define the difference between the current estimate of the parameters and the best values of the parameters defined by (22) and (23).

Consider the indirect adaptive control law

$$u_p = u_{ce} + u_{si} + u_{bi}. \quad (28)$$

The control law is comprised of a "bounding control" term, u_{bi}, a "sliding mode" term, u_{si}, and a "certainty equivalence" [25] control term u_{ce}.

A. Certainty Equivalence Control Term

The certainty equivalence control term [36] is defined as

$$u_{ce} = \frac{1}{\beta_k(t) + \hat{\beta}(X)} \{-[\alpha_k(t) + \hat{\alpha}(X)] + \nu(t)\} \quad (29)$$

where $\nu(t) := y^{(r)}_m + \eta e_s + \overline{e}_s$, with $\overline{e}_s := \dot{e}_s - e^{(r)}_o$ and $\eta > 0$. For now we assume that $\beta_k(t) + \hat{\beta}(X)$ is bounded away from zero so that (29) is well-defined, however, we shall later show how to ensure that this is the case. The tracking error is defined as $e_s := k^T e$ where $e := [e_o \quad \dot{e}_o \quad \cdots \quad e^{(r-1)}_o]^T$, $k := [k_0 \quad \cdots \quad k_{r-2} \quad 1]^T$, and $e_o := y_m - y_p$, thus, $\overline{e}_s = [\dot{e}_o \quad \cdots \quad e^{(r-1)}_o][k_0 \quad \cdots \quad k_{r-2}]^T$. We pick the elements of k such that $\hat{L}(s) := s^{r-1} + k_{r-2}s^{r-2} + \cdots + k_1 s + k_0$ has its roots in the open left half plane. The goal of the adaptive algorithm is to "learn" how to control the plant to drive e_s to zero. Thus, e_s is a measure of the tracking error. The term "certainty equivalence" is used to describe u_{ce} since this control term is obtained by assuming that the current estimates of the plant parameters are close to the actual plant parameters, so that a "feedback linearizing controller" may be obtained [25]. Even though the current estimates may not be close to the actual plant parameters, the certainty equivalence control term may be used to later manipulate the system dynamics into a special form.

Using the control (28), the rth derivative of the output error becomes $e^{(r)}_o = y^{(r)}_m - y^{(r)}_p$ so

$$\begin{aligned} e^{(r)}_o = & y^{(r)}_m - [\alpha_k(t) + \alpha(X)] - \frac{\beta_k(t) + \beta(X)}{\beta_k(t) + \hat{\beta}(X)} \\ & \cdot \{-[\alpha_k(t) + \hat{\alpha}(X)] + \nu(t)\} \\ & - [\beta_k(t) + \beta(X)](u_{si} + u_{bi}). \end{aligned} \quad (30)$$

We may rearrange terms so that

$$\begin{aligned} e^{(r)}_o = & \left[1 - \frac{\beta_k(t) + \beta(X)}{\beta_k(t) + \hat{\beta}(X)} \right] \\ & \cdot \{-[\alpha_k(t) + \hat{\alpha}(X)] + \nu(t)\} - \alpha(X) + \hat{\alpha}(X) \\ & - \eta e_s - \overline{e}_s - [\beta_k(t) + \beta(X)](u_{si} + u_{bi}) \quad (31) \\ = & [\hat{\alpha}(X) - \alpha(X)] + [\hat{\beta}(X) - \beta(X)]u_{ce} \\ & - \eta e_s - \overline{e}_s - [\beta_k(t) + \beta(X)](u_{si} + u_{bi}). \quad (32) \end{aligned}$$

We may express (32) as

$$\begin{aligned} \dot{e}_s + \eta e_s = & [\hat{\alpha}(X) - \alpha(X)] + [\hat{\beta}(X) - \beta(X)]u_{ce} \\ & - [\beta_k(t) + \beta(X)](u_{si} + u_{bi}). \end{aligned} \quad (33)$$

With this representation, we next define the bounding and sliding mode control terms in (28).

B. Bounding Control Term

Later, we will show that if the plant states are bounded then an indirect adaptive fuzzy controller may be used to provide stable, asymptotic tracking of the output. At this point, however, we need to define a "bounding control" u_{bi} to ensure that the output and states are bounded. Consider

$$v_{bi} = \tfrac{1}{2} e^2_s. \quad (34)$$

Using (33) and (34), and the fact the $\beta_k(t) + \hat{\beta}(X) \geq \beta_0 > 0$, for some β_0 we obtain

$$\begin{aligned} \dot{v}_{bi} = & -\eta e^2_s + e_s \{[\hat{\alpha}(X) - \alpha(X)] + [\hat{\beta}(X) - \beta(X)]u_{ce} \\ & - [\beta_k(t) + \beta(X)](u_{si} + u_{bi})\} \quad (35) \\ \leq & -\eta e^2_s + |e_s|\{|\hat{\alpha}(X)| + |\alpha(X)| \\ & + [|\hat{\beta}(X)| + |\beta(X)|]|u_{ce}|\} \\ & + |e_s|\{[\beta_k(t) + \beta(X)]|u_{si}|\} \\ & - e_s[\beta_k(t) + \beta(X)]u_{bi}. \quad (36) \end{aligned}$$

Let ϵ_M and M_e be fixed parameters such that $0 < \epsilon_M \leq M_e$. We choose the bounding control to be

$$u_{bi} = \Pi(t) k_{bi}(t) \operatorname{sgn}(e_s) \quad (37)$$

where

$$\Pi(t) = \begin{cases} 1, & \text{if } M_e \le |e_s| \\ \dfrac{|e_s| + \epsilon_M - M_e}{\epsilon_M}, & \text{if } M_e - \epsilon_M \le |e_s| < M_e \\ 0, & \text{otherwise} \end{cases} \tag{38}$$

and

$$\operatorname{sgn}(x) := \begin{cases} 1 & x > 0 \\ -1 & x < 0. \end{cases} \tag{39}$$

The bounding control is continuous and defined so that it is always used when $|e_s| \ge M_e$. We require that there are known bounds $\beta_1(X) \ge |\beta(X)|$ and $\alpha_1(X) \ge |\alpha(X)|$ when $|e_s| \ge M_e$ with $\alpha_1(X)$ and $\beta_1(X)$ continuous in x. Using these state dependent bounds, the following gain is used

$$k_{bi}(t) = \frac{1}{\beta_0}\{|\hat{\alpha}(X)| + \alpha_1(X) + [|\hat{\beta}(X)| + \beta_1(X)]|u_{ce}|\} + |u_{si}|. \tag{40}$$

Using (36) with (40), we obtain

$$\dot{v}_{bi} \le -\eta e_s^2, \quad \text{if} \quad |e_s| \ge M_e. \tag{41}$$

Thus, we are ensured that if there exists a time t' such that $|e_s(t')| > M_e$, then for $t > t'$, $|e_s(t)|$ will decrease exponentially until $|e_s| \le M_e$.

At this point, it is convenient to define transfer functions

$$\hat{G}_i(s) := \frac{s^i}{\hat{L}(s)}, \quad i = 0, \cdots, r-1 \tag{42}$$

which each are stable since $\hat{L}(s)$ has its poles in the open left half plane. Since $e_o^{(i)} = \hat{G}_i(s)e_s$ with e_s bounded, then $e_o^{(i)} \in \mathcal{L}_\infty$ ($\mathcal{L}_\infty = \{z(t)\colon \sup_t |z(t)| < \infty\}$). This is shown for the case $e_s = \dot{e}_o + k_o e_o$ in Fig. 3 where if $|e_s| \le M_e$ then e_o and $\dot{e}_o$ stay in the shaded region (i.e., $|e_o| \le M_e/k_o$ and $|\dot{e}_o| \le 2M_e$). This may be extended to higher dimensional systems as

$$|e_o^{(i)}| \le M_e\|\hat{G}_i(s)\|_1, \quad i = 0, \cdots, r-2 \tag{43}$$

and since $e_o^{(r-1)} = e_s - \sum_{i=0}^{r-2} k_i e_o^{(i)}$ the triangular inequality may be used to show that

$$|e_o^{(r-1)}| \le M_e + M_e \sum_{i=0}^{r-2} k_i \|\hat{G}_i(s)\|_1 \tag{44}$$

for all time if $|e_s| \le M_e$ and the initial conditions are such that $|e_o^{(i)}(0)| \le M_e\|\hat{G}_i(s)\|_1$, $i = 0, \cdots, r-2$. The transfer function 1-norm is defined as $\|\hat{G}_i(s)\|_1 := \int_{-\infty}^{\infty} |g_i(\tau)| d\tau$, where $g_i(t)$ is the impulse response of $\hat{G}_i(s)$. Using the example of $e_s = \dot{e}_o + k_0 e_o$, we obtain $\hat{G}_0(s) = 1/(s + k_0)$ which has an impulse response function of $g_0(t) = e^{-k_0 t}$ with the 1-norm $\|\hat{G}_0(s)\|_1 = 1/k_0$. Using (43) and (44), we obtain the bounds $|e_o| \le M_e/k_0$ and $|\dot{e}_o| \le 2M_e$, as shown in Fig. 3. Overall, we see that (43) and (44) provide explicit bounds on the output error when the bounding control u_{bi} is used.

Up to this point, we have shown output-error boundedness. Next, we show that for some plants state boundedness is also guaranteed. The dynamics for a relative degree r plant described by (16) may be written in normal form as

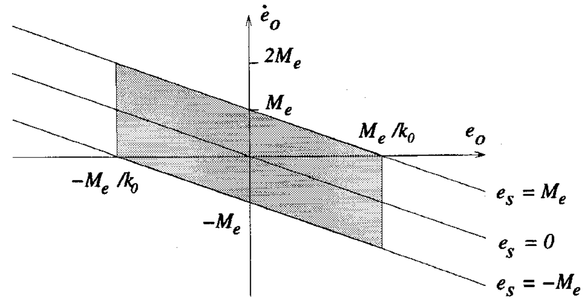


Fig. 3. Boundedness around the manifold $e_s = \dot{e}_o + k_o e_o = 0$.

$$\dot{\xi}_1 = \xi_2 \tag{45}$$

$$\vdots \quad \vdots \tag{46}$$

$$\dot{\xi}_{r-1} = \xi_r \tag{47}$$

$$\dot{\xi}_r = \alpha(\xi, \pi) + \beta(\xi, \pi)u_p \tag{48}$$

$$\dot{\pi} = \Psi(\xi, \pi) \tag{49}$$

with $\pi \in \Re^{n-r}$ and $y_p = \xi_1$. The "zero dynamics" of the system are given as

$$\dot{\pi} = \Psi(0, \pi). \tag{50}$$

We may now consider the adaptive control of plants with no zero dynamics, or plants which have exponential attractivity of the zero dynamics (i.e., plants where (50) is exponentially stable when the states π move outside a ball $|\pi| > B$). The two plant types are characterized by the following assumptions.

P1) Plant Assumption: The plant is of relative degree $r = n$ (i.e., no zero dynamics) such that

$$\frac{d}{dt}x_i = x_{i+1}, \quad i = 1, \cdots, n-1$$

$$\frac{d}{dt}x_n = \alpha(X) + \alpha_k(t) + [\beta(X) + \beta_k(t)]u_p$$

where $y_p = x_1$, with $\alpha_k(t)$ and $\beta_k(t)$ known functions. Here, it is assumed that there exists $\beta_0 > 0$ such that $\beta(X) + \beta_k(t) \ge \beta_0$, and that $x_1, \cdots, x_n$ are measurable.

P2) Plant Assumption: The plant is of relative degree r, $1 \le r < n$ with the zero dynamics exponentially attractive and there exists $\beta_0 > 0$ such that $\beta(X) + \beta_k(t) \ge \beta_0$. The outputs $y_p, \cdots, y_p^{(r-1)}$ are measurable.

Clearly, plants satisfying P1) have bounded states if the reference input, y_m, and its derivatives are bounded with the output error e_o and its derivatives bounded. We may use Lipschitz properties of $\Psi(\xi, \pi)$ to see that plants satisfying P2) have bounded states if the output is bounded in the following manner [25]. For some positive constants γ_1, γ_2, γ_3, γ_4, and B and function v_1 we have

$$\gamma_1|\pi|^2 \le v_1(\pi) \le \gamma_2|\pi|^2 \tag{51}$$

$$\frac{dv_1}{d\pi}\Psi(0, \pi) \le -\gamma_3|\pi|^2, \quad \text{if} \quad |\pi| > B \tag{52}$$

$$\left|\frac{dv_1}{d\pi}\right| \le \gamma_4|\pi| \tag{53}$$

if the zero dynamics are exponentially attractive. Since we have e_o bounded and bounded reference signals, by *R1*, $|\xi| \le k_1$ where k_1 is some positive constant. Using (52), we have

$$\dot{v}_1 = \frac{dv_1}{d\pi}\,\Psi(\xi,\,\pi) \tag{54}$$

$$\le -\gamma_3|\pi|^2 + \frac{dv_1}{d\pi}\,[\Psi(\xi,\,\pi) - \Psi(0,\,\pi)] \quad \text{if} \quad |\pi| > B. \tag{55}$$

If $\Psi(\xi,\,\pi)$ is Lipschitz in ξ, then $|\Psi(\xi,\,\pi) - \Psi(0,\,\pi)| \le k_2|\xi|$ some positive k_2. Using this, if $|\pi| > B$ we now have

$$\dot{v}_1 \le -\gamma_3|\pi|^2 + \left|\frac{dv_1}{d\pi}\right| |[\Psi(\xi,\,\pi) - \Psi(0,\,\pi)]| \tag{56}$$

$$\le -\gamma_3|\pi|^2 + \gamma_4 k_2|\xi||\pi| \tag{57}$$

$$\le -\gamma_3|\pi|^2 + \gamma_4 k_1 k_2|\pi|. \tag{58}$$

Therefore, $\dot{v}_1 \le 0$ if $|\pi| \ge \max(B,\,\gamma_4 k_1 k_2/\gamma_3)$. This ensures boundedness of ξ and π, therefore the system states are bounded.

Since the fuzzy systems are used to approximate $\alpha(X)$ and $\beta(X)$, we require that the plant be described by either P1) or P2), ensuring state boundedness so that the fuzzy system input-membership functions do not need to cover all $\Re^n$. The subspace through which the plant state trajectory may travel S_x is determined by first finding the range of the reference signal and the output error from (43) and (44). Then, the range of the states may be determined from the particular application and choice of state representation.

C. Adaptation Algorithm

Consider the following Lyapunov function candidate

$$V_i = \tfrac{1}{2}e_s^2 + \tfrac{1}{2}\mathrm{tr}\,(\Phi_\alpha^T Q_\alpha \Phi_\alpha) + \tfrac{1}{2}\mathrm{tr}\,(\Phi_\beta^T Q_\beta \Phi_\beta) \tag{59}$$

where $\mathrm{tr}\,(\cdot)$ is the trace operator ($\{\mathrm{tr}\,(A) = \sum_i a_{i,i}$, if $A = [a_{i,j}]$ is square)$\}$ with $Q_\alpha \in \Re^{m_\alpha \times m_\alpha}$ and $Q_\beta \in \Re^{m_\beta \times m_\beta}$ positive definite and diagonal. This Lyapunov candidate quantifies both the error in tracking and in the parameter estimates. Taking the derivative of (59) yields

$$\dot{V}_i = e_s[\dot{e}_s] + \mathrm{tr}\,(\Phi_\alpha^T Q_\alpha \dot{\Phi}_\alpha) + \mathrm{tr}\,(\Phi_\beta^T Q_\beta \dot{\Phi}_\beta). \tag{60}$$

Substituting in the derivative of the tracking error $\dot{e}_s$ from (33) yields

$$\begin{aligned}\dot{V}_i = {} & e_s\{[\hat{\alpha}(X) - \alpha(X)] + [\hat{\beta}(X) - \beta(X)]u_{ce} \\ & - \eta e_s - [\beta_k(t) + \beta(X)](u_{si} + u_{bi})\} \\ & + \mathrm{tr}\,(\Phi_\alpha^T Q_\alpha \dot{\Phi}_\alpha) + \mathrm{tr}\,(\Phi_\beta^T Q_\beta \dot{\Phi}_\beta).\end{aligned} \tag{61}$$

We may use (20), (21), and (24)–(27) to obtain

$$\begin{aligned}\dot{V}_i = {} & -\eta e_s^2 + \{z_\alpha^T \Phi_\alpha \zeta_\alpha - d_\alpha(X) + z_\beta \Phi_\beta^T \zeta_\beta u_{ce} \\ & - d_\beta(X)u_{ce} - [\beta_k(t) + \beta(X)](u_{si} + u_{bi})\}e_s \\ & + \mathrm{tr}\,(\Phi_\alpha^T Q_\alpha \dot{\Phi}_\alpha) + \mathrm{tr}\,(\Phi_\beta^T Q_\beta \dot{\Phi}_\beta).\end{aligned} \tag{62}$$

Now consider the following fuzzy system update laws

$$\dot{A}_\alpha(t) = -\,Q_\alpha^{-1} z_\alpha \zeta_\alpha^T e_s \tag{63}$$

$$\dot{A}_\beta(t) = -\,Q_\beta^{-1} z_\beta \zeta_\beta^T e_s u_{ce}. \tag{64}$$

Using the fact that $\dot{\Phi}_\alpha = \dot{A}_\alpha$, $\dot{\Phi}_\beta = \dot{A}_\beta$, and $\mathrm{tr}\,(AB) = \mathrm{tr}\,(BA)$ where $A \in \Re^{n\times m}$ and $B \in \Re^{m \times n}$, the adaptive update laws (63) and (64) may be used so that (62) is expressed as

$$\begin{aligned}\dot{V}_i = {} & -\eta e_s^2 + [z_\alpha^T \Phi_\alpha \zeta_\alpha - d_\alpha(X) \\ & + z_\beta \Phi_\beta^T \zeta_\beta u_{ce} - d_\beta(X)u_{ce}]e_s \\ & - [\beta_k(t) + \beta(X)](u_{si} + u_{bi})e_s \\ & - \mathrm{tr}\,(z_\alpha^T \Phi_\alpha \zeta_\alpha)e_s - \mathrm{tr}\,(z_\beta^T \Phi_\beta \zeta_\beta)e_s u_{ce}.\end{aligned} \tag{65}$$

Equation (65) may equivalently be expressed as

$$\begin{aligned}\dot{V}_i = {} & -\eta e_s^2 - [d_\alpha(X) + d_\beta(X)u_{ce}]e_s \\ & - [\beta_k(t) + \beta(X)](u_{si} + u_{bi})e_s.\end{aligned} \tag{66}$$

The fuzzy system adaptation laws defined by (63) and (64) do not guarantee that $A_\alpha \in \Omega_\alpha$ and $A_\beta \in \Omega_\beta$. To guarantee this, a "projection algorithm" is used. If the parameter spaces are defined so that the parameters are bounded by $A_\alpha \in [A_\alpha^{\min},\,A_\alpha^{\max}]$ and $A_\beta \in [A_\beta^{\min},\,A_\beta^{\max}]$ then a simple projection algorithm may be used (we use the notation $A \in [A^{\min},\,A^{\max}]$ to define bounds on the matrix elements such that $a_{i,j} \in [a_{i,j}^{\min},\,a_{i,j}^{\max}]$, $i = 1,\cdots,m$, $j = 1,\cdots,n$ where A, $A^{\min}$, $A^{\max} \in \Re^{m\times n}$ are defined element by element as $A = [a_{i,j}]$, $A^{\min} = [a_{i,j}^{\min}]$, and $A^{\max} = [a_{i,j}^{\max}]$). Define $\overline{a}_{\alpha_{i,j}}$ to be the i, jth element of $z_\alpha \zeta_\alpha^T e_s$ and $\overline{a}_{\beta_{i,j}}$ to be the i, jth element of $z_\beta \zeta_\beta^T e_s u_{ce}$. Then the parameter matrices are updated according to

$$\dot{A}_\alpha = -\,Q_\alpha^{-1}\hat{A}_\alpha \tag{67}$$

$$\dot{A}_\beta = -\,Q_\beta^{-1}\hat{A}_\beta \tag{68}$$

where

$$\hat{a}_{\alpha_{i,j}} = \begin{cases} 0, & \text{if } a_{\alpha_{i,j}} \notin (a_{\alpha_{i,j}}^{\min},\,a_{\alpha_{i,j}}^{\max}) \\ & \text{and } \overline{a}_{\alpha_{i,j}}(a_{\alpha_{i,j}} - a_{\alpha_{i,j}}^{c}) < 0 \\ \overline{a}_{\alpha_{i,j}}, & \text{otherwise}\end{cases} \tag{69}$$

$$\hat{a}_{\beta_{i,j}} = \begin{cases} 0, & \text{if } a_{\beta_{i,j}} \notin (a_{\beta_{i,j}}^{\min},\,a_{\beta_{i,j}}^{\max}) \\ & \text{and } \overline{a}_{\beta_{i,j}}(a_{\beta_{i,j}} - a_{\beta_{i,j}}^{c}) < 0 \\ \overline{a}_{\beta_{i,j}}, & \text{otherwise}\end{cases} \tag{70}$$

with some fixed $A_\alpha^c \in (A_\alpha^{\min},\,A_\alpha^{\max})$ and fixed $A_\beta^c \in (A_\beta^{\min},\,A_\beta^{\max})$. Assuming that the parameters are initialized such that $A_\alpha \in [A_\alpha^{\min},\,A_\alpha^{\max}]$ and $A_\beta \in [A_\beta^{\min},\,A_\beta^{\max}]$, the element $a_{\alpha,\beta_{i,j}}$ will become greater than that its maximum bound only if $a_{\alpha,\beta_{i,j}} = a_{\alpha,\beta_{i,j}}^{\max}$ and $\hat{a}_{\alpha,\beta_{i,j}} > 0$. Since the projection algorithm prevents this from occurring, we are ensured that $A_{\alpha,\beta} \le A_{\alpha,\beta}^{\max}$ where "$\le$" is an element-wise relation (a similar argument may be made ensuring that $A_{\alpha,\beta} \ge A_{\alpha,\beta}^{\min}$). Thus, using this modified update law will ensure that the parameter matrices will stay within the feasible parameter space.

Using the projection algorithm, we are also ensured that

$$\begin{aligned}\dot{V}_i \le {} & -\eta e_s^2 - [d_\alpha(X) + d_\beta(X)u_{ce}]e_s \\ & - [\beta_k(t) + \beta(X)](u_{si} + u_{bi})e_s\end{aligned} \tag{71}$$

since the modified adaptation law guides the searching algorithm toward the optimal parameters, A_α^* and A_β^*, thus, helping to decrease $\dot{V}_i$. To see this, we notice that from (62), the term

$$\begin{aligned} z_\alpha^T \Phi_\alpha \zeta_\alpha e_s &+ \mathrm{tr}\,(\Phi_\alpha^T Q_\alpha \dot{\Phi}_\alpha) \\ &= z_\alpha^T \Phi_\alpha \zeta_\alpha e_s - \mathrm{tr}\,(\Phi_\alpha^T \hat{A}_\alpha) && (72) \\ &= z_\alpha^T \Phi_\alpha \zeta_\alpha e_s - \mathrm{tr}\,[\Phi_\alpha^T (\hat{A}_\alpha - \overline{A}_\alpha) + \Phi_\alpha^T \overline{A}_\alpha] && (73) \\ &= -\mathrm{tr}\,[\Phi_\alpha^T (\hat{A}_\alpha - \overline{A}_\alpha)] && (74) \\ &= -\sum_i \sum_j \phi_{\alpha_{i,j}} (\hat{a}_{\alpha_{i,j}} - \overline{a}_{\alpha_{i,j}}) \le 0 && (75) \end{aligned}$$

where $\Phi_\alpha = [\phi_{\alpha_{i,j}}]$. Similarly, we have $z_\beta^T \Phi_\beta \zeta_\beta u_{ce} e_s + \mathrm{tr}\,(\Phi_\beta^T Q_\beta \dot{\Phi}_\beta) \le 0$, thus, we may establish the inequality of (71) using (65) and the modified fuzzy system update algorithm given by (67) and (68). Since the errors in representing the plant nonlinearities with fuzzy systems or neural networks $d_\alpha(X)$ and $d_\beta(X)$, in general, are nonzero, a sliding mode term is now defined which ensures negative semidefiniteness of (71).

D. Sliding-Mode Control Term

To ensure that (71) is negative semidefinite, we choose

$$u_{si} = \frac{k_{si}(t)}{\beta_0}\,\mathrm{sgn}\,(e_s) \tag{76}$$

where $k_{si}(t) = D_\alpha(X) + D_\beta(X)|u_{ce}|$. Since $-[d_\alpha(X) + d_\beta(X)u_{ce}]e_s \le [|d_\alpha(X)| + |d_\beta(X)u_{ce}|]|e_s|$, and from (37) $-[\beta_k(t) + \beta(X)]u_{bi}e_s = -[\beta_k(t) + \beta(X)]\Pi k_{bi}(t)|e_s| \le 0$, we may rewrite (71) as

$$\begin{aligned} \dot{V}_i \le -\eta e_s^2 &+ [|d_\alpha(X)| + |d_\beta(X)u_{ce}|]|e_s| \\ &- e_s[\beta_k(t) + \beta(X)]u_{si}. \end{aligned} \tag{77}$$

Combining (76) and (77) we have

$$\dot{V}_i \le -\eta e_s^2. \tag{78}$$

Thus, $\dot{V}_i$ is negative semidefinite and $V_i \in \mathcal{L}_\infty$. It should be noted that even though u_{si} is called a "sliding-mode" term, it does not guarantee that the state trajectory will "slide" along the manifold $e_s = 0$ as traditionally guaranteed with nonadaptive sliding-mode control [27]. The sliding-mode term is required to overcome modeling errors between the nonlinear functions of the system $\alpha(X)$ and $\beta(X)$, and the fuzzy systems or neural networks with optimal parameters $z_\alpha^T A_\alpha^* \zeta_\alpha$ and $z_\beta^T A_\beta^* \zeta_\beta$.

E. Stability Properties

The assumptions for the controller are summarized in the following:

C1) Control Assumption: The fuzzy systems (neural networks) are defined such that $D_\alpha(X), D_\beta(X) \in \mathcal{L}_\infty$, for $X \in S_x \subseteq \Re^n$ and there are some known continuous functions $\alpha_1(X)$ and $\beta_1(X)$ such that $\alpha_1(X) \ge |\alpha(X)|$ and $\beta_1(X) \ge |\beta(X)|$. The projection algorithm is defined such that $\hat{\beta}(X) + \beta_k(t) \ge \beta_0 > 0$.

We now summarize the properties of the indirect adaptive controller in Theorem 1.

Theorem 1: Stability and tracking results using indirect adaptive control:

If reference input assumption R1) holds, either plant assumption P1) or P2) holds, and the control law is defined by (28) with the control assumptions C1).

Then the following holds.

a) The plant output and its derivatives $y_p, \cdots, y_p^{(r-1)}$ are bounded.
b) The control signals are bounded, i.e., $u_{bi}, u_{ce}, u_{si} \in \mathcal{L}_\infty$.
c) The magnitude of the output error $|e_o|$ decreases at least asymptotically to zero, i.e., $\lim_{t\to\infty} |e_o| = 0$.

Proof of Theorem 1:

Part 1) Equations (43) and (44) guarantee that $|e_o^{(i)}| \in \mathcal{L}_\infty$, $i = 0, \cdots, r-1$ since $|e_s|$ is bounded from (78). By definition, $e_o^{(i)} = y_m^{(i)} - y_p^{(i)}, \quad \forall i = 0, \cdots, r-1$, with $y_m^{(i)}$ and $e_o^{(i)}$ bounded; therefore, $y_p^{(i)}, \quad \forall i = 0, \cdots, r-1$ is bounded.

Part 2) With $y_p, \cdots, y_p^{(r-1)} \in \mathcal{L}_\infty$, the plant states are bounded using plant assumptions *P1* or *P2*. This implies that $\alpha(X), \alpha_k(t), \beta(X), \beta_k(t) \in \mathcal{L}_\infty$. The projection algorithm ensures that $\beta_k(t) + \hat{\beta}(X)$ is bounded away from zero and that $\hat{\alpha}(X)$ is bounded, thus, $u_{ce} \in \mathcal{L}_\infty$. With $\alpha_1(X), \beta_1(X) \in \mathcal{L}_\infty$ we establish that $u_{bi} \in \mathcal{L}_\infty$. Since the fuzzy systems are defined appropriately so that $D_\alpha(X), D_\beta(X) \in \mathcal{L}_\infty$, then $u_{si} \in \mathcal{L}_\infty$.

Part 3) To show asymptotic stability of the output, we would like to find a bound on $\int_0^\infty e_s^2\,dt$. Using (78) we have

$$\begin{aligned} \int_0^\infty \eta e_s^2\,dt &\le -\int_0^\infty \dot{V}_i\,dt && (79) \\ &= V_i(0) - V_i(\infty). && (80) \end{aligned}$$

This establishes that $e_s \in \mathcal{L}_2$ ($\mathcal{L}_2 = \{z(t)\colon \int_0^\infty z^2(t)\,dt < \infty\}$) since $V_i(0), V_i(\infty) \in \mathcal{L}_\infty$. If $V_i \in \mathcal{L}_\infty$ then $e_s \in \mathcal{L}_\infty$ by the definition of V_i. In addition, we know that $e_o^{(i)} \in \mathcal{L}_\infty$, $i = 0, \cdots, r-1$ since $e_s \in \mathcal{L}_\infty$ and $e_o^{(i)} = G_i(s)e_s$, with all the poles of $G_i(s)$, $i = 0, \cdots, r-1$ in the open left-half plane. If $\alpha(X), \hat{\alpha}(X), \beta(X), \hat{\beta}(X), \beta_k(t), u_{ce}, u_{bi}, u_{si} \in \mathcal{L}_\infty$, then $\dot{e}_s \in \mathcal{L}_\infty$ from (33). Since $e_s \in \mathcal{L}_2, \mathcal{L}_\infty$, and $\dot{e}_s \in \mathcal{L}_\infty$, by Barbalat's Lemma we have asymptotic stability of e_s (i.e., $\lim_{t\to\infty} e_s = 0$), which implies asymptotic stability of e_o (i.e., $\lim_{t\to\infty} e_o = 0$). □

Remark 1.1: The bounding control term, u_{bi}, within the indirect adaptive control law defined by (28) is used to restrict the output trajectory so that fuzzy systems may be defined for a small range of inputs. Without the bounding control term, an output trajectory may travel over a space which is so large that specifying a fuzzy system or neural network is cumbersome. Using a bounding controller in this manner is similar to the supervisory control in [37]. The sliding-mode control term u_{si} is used to compensate for approximation errors in representing the actual nonlinear dynamics by fuzzy systems or neural networks with ideal parameter values. The certainty equivalence term is then used to "learn" the unknown dynamics of the system providing asymptotic convergence of the tracking error.

Remark 1.2: It is possible to incorporate linguistic information about the plant since $\alpha_k(t)$ and $\beta_k(t)$ may be linear combinations of Lipschitz functions. For example, we may use rules R_1 through R_{k_α} associated with α and the rules R_1 through R_{k_β} associated with β to describe the plant according to a set of linguistics, while rules $R_{k_\alpha+1}$ through R_{p_α} and $R_{k_\beta+1}$ through R_{p_β} may be used for the fuzzy estimation in the identifier. The advantage of using linguistic information in this manner is that if the first k rules describe the plant fairly well; then, in general, the magnitude of the fuzzy estimation error will be small, thus providing better tracking of the reference signal as the fuzzy estimators learn $\alpha(X)$ and $\beta(X)$. The linguistics used to describe $\alpha_k(t)$ may be obtained by setting $u_p = 0$, and describing how the system output, $y_p^{(r)}$, behaves. Once this is done, it is possible to find a set of linguistics describing $\beta_k(t)$. To do this, allow an input into the system and characterize how the same input effects the system output, $y_p^{(r)}$, in different operating regions. It is interesting to note that the designer does not have to know how to control the plant when incorporating linguistics, rather, one simply describes how the plant itself behaves, thus allowing the indirect adaptive fuzzy controller to use the linguistics to better control the system. It is also possible for $\alpha_k(t)$ and $\beta_k(t)$ to be the output of fuzzy systems separate from the fuzzy estimation systems. It should also be emphasized that little information about the plant is needed since $\alpha_k(t)$ and $\beta_k(t)$ may be arbitrarily set to zero. In [15], *a priori* knowledge of the plant can be incorporated, however, this is done simply by setting the initial conditions of the controller to prespecified values, rather than explicitly allowing for $\alpha_k(t)$ and $\beta_k(t)$.

Remark 1.3: The parameter error matrices, Φ_α and Φ_β, are bounded if A_α^* and A_β^* are bounded since we have $A_\alpha \in [A_\alpha^{\min}, A_\alpha^{\max}]$ and $A_\beta \in [A_\beta^{\min}, A_\beta^{\max}]$ according to our projection algorithm. It is important to pick the bounds on the elements of A_β so that $\beta_k(t) + \hat{\beta}(X) \geq \beta_0 > 0$. If $A_\beta^{\min}$ and $A_\beta^{\max}$ are not properly chosen, then it is possible that $\beta_k(t)+\hat{\beta}(X)$ is not bounded away from zero, thus causing (29) to become undefined. Consider the case in which a standard fuzzy system is used to represent $\beta(X)$ (i.e., $z_\beta = [1]$) and $\beta_k = 0$. Then $a_\beta^{\min} \leq \hat{\beta}(X) \leq a_\beta^{\max}$. This is true since each $\zeta_{\beta,i} \geq 0$ so that

$$\sum_{i=1}^{p_\beta} \alpha_{\beta,i}^{\min}\zeta_i = \alpha_\beta^{\min} \leq \hat{\beta}(X) \leq \sum_{i=1}^{p_\beta} \alpha_\beta^{\max}\zeta_{\beta,i} = \alpha_\beta^{\max}.$$

If $\beta_k(X)$ is nonzero, or Takagi–Sugeno fuzzy systems are used then the design of a projection algorithm will be dependent upon the choice of $\beta_k(X)$ and z_β^T.

Remark 1.4: Even though the bounding control term, u_{bi}, was not used explicitly in Theorem 1, it is used to confine the plant states to a known region so that the input membership functions of a fuzzy system may be defined over this region. Knowledge of the range over which the input membership functions must be defined is required for $D_\alpha(X)$ and $D_\beta(X)$ to be small. The fuzzy systems or neural networks are to be designed such that the approximation errors $D_\alpha(X)$ and $D_\beta(X)$ are small when the state trajectory travels within some region. When the state trajectory travels outside this region, however, $D_\alpha(X)$ and $D_\beta(X)$ may become large so that the control gain associated with u_{si} is large, thus causing the adaptive controller to act similar to a poorly designed sliding mode controller rather than an adaptive controller with a small sliding mode contribution.

Remark 1.5: The bounding control term requires that an upper bound on $\beta(X)$ (i.e., β_1) is known. This requirement may be eliminated since

$$\begin{aligned}\dot{v}_{bi} =& -\eta e_s^2 + e_s\{[\hat{\alpha}(X) - \alpha(X)] \\ &+ [\hat{\beta}(X) + \beta_k(t) - \beta(X) - \beta_k(t)]u_{ce}\} \\ &- e_s[\beta_k(t) + \beta(X)](u_{si} + u_{bi}) \\ \leq& -\eta e_s^2 + |e_s|\{|\hat{\alpha}(X)| + |\alpha(X)| \\ &+ [|\hat{\beta}(X) + \beta_k(t)|]|u_{ce}|\} - [\beta(X) + \beta_k(t)]u_{ce}e_s \\ &+ |e_s|\{[\beta_k(t) + \beta(X)]|u_{si}|\} \\ &- e_s[\beta_k(t) + \beta(X)]u_{bi}. \end{aligned} \tag{81}$$

Defining the bounding control term as

$$\begin{aligned} u_{bi} = \Bigg(\Bigg\{ &\frac{1}{\beta_0}[|\hat{\alpha}(X) + |\alpha_1(X)| \\ &+ |\hat{\beta}(X) + \beta_k(t)|\,|u_{ce}|] + |u_{si}|\Bigg\} \\ &\cdot \operatorname{sgn}(e_s) - u_{ce}\Bigg)\Pi(t)\end{aligned}$$

will once again ensure that

$$\dot{v}_{bi} \leq -\eta e_s^2, \quad \text{if} \quad |e_s| \geq M_e.$$

Remark 1.6: While the bounding control term was added to help with the definition of the input membership functions of the fuzzy system, if the bounding control term is removed, then asymptotic stability is still achieved since the plant states are still bounded so Barbalat's lemma may still be applied. This is true since

$$\begin{aligned}V_i(0) =& \tfrac{1}{2}e_s^2(0) + \tfrac{1}{2}\mathrm{tr}\,[\Phi_\alpha^T(0)Q_\alpha\Phi_\alpha(0)] \\ &+ \tfrac{1}{2}[\Phi_\beta^T(0)Q_\beta\Phi_\beta(0)]\end{aligned} \tag{82}$$

and since the Lyapunov function $V_i(t)$ is positive and nonincreasing

$$e_s^2(t) \leq V_i(t) \leq V_i(0). \tag{83}$$

Thus the size of the tracking error may be reduced by either making $\Phi_\alpha(0)$ or $\Phi_\beta(0)$ small, or increasing the adaptation gains, Q_α^{-1} and Q_β^{-1}. However, in implementation, large Q_α^{-1} and Q_β^{-1} may result in instabilities because of the time delays associated with digital input and output. Thus, if a bounding term is not to be used, then a good initial estimate of $\alpha(X)$ and $\beta(X)$ may improve the performance of the closed-loop system.

Remark 1.7: If there exists some *constants* D_α and D_β such that $|d_\alpha(X)| \leq D_\alpha$ and $d_\beta(X) \leq D_\beta$ for all $X \in S_x$, then it is possible to use an adaptive routine to find D_α and D_β while preserving the previous stability results [17], [21]. The sliding-mode gain may be modified as $k_{si}(t) = \hat{D}_\alpha + \hat{D}_\beta|u_{ce}|$ where $\hat{D}_\alpha$ and $\hat{D}_\beta$ are the current estimates of D_α and D_β, respectively. The update laws for the estimates of the bounds on the errors in representing the nonlinear dynamics of the plant with fuzzy systems or neural networks are given as

$$\dot{\hat{D}}_\alpha = \frac{1}{q_\alpha}|e_s| \quad (84)$$

$$\dot{\hat{D}}_\beta = \frac{1}{q_\beta}|e_s u_{ce}| \quad (85)$$

where q_α, $q_\beta > 0$. Though $|e_s|$ converges to zero over time, we notice that $\hat{D}_\alpha$ and $\hat{D}_\beta$ may become large since $\dot{\hat{D}}_\alpha$ and $\dot{\hat{D}}_\beta$ are always positive. A projection algorithm may be used to bound the growth of $\hat{D}_\alpha$ and $\hat{D}_\beta$, but then the bounds must be known ahead of time so that the sliding mode gain may over time become equal to the original nonadaptive sliding mode gain. Stability is proven by adding the terms $q_\alpha(\hat{D}_\alpha - D_\alpha)^2/2$ and $q_\beta(\hat{D}_\beta - D_\beta)^2/2$ to the Lyapunov candidate for the indirect adaptive algorithm defined in (59).

Remark 1.8: Though many of the concepts within this paper are similar to those used within previous work, specifically within [15], [20], and [22], there are many significant differences: i) the results of Theorem 1 may be applied to a class of plants with zero dynamics (this was not done in [15], [20], and [22]), ii) this paper ensures asymptotic tracking convergence using a larger class of fuzzy systems (i.e., Takagi–Sugeno fuzzy systems) than in [15] and a larger class of neural networks (i.e., those with a second hidden layer) than in [20], iii) within this paper we allow for the direct inclusion of a mathematical description of the known part of the plant through $\alpha_k(t)$ and $\beta_k(t)$ (this was not done in [15], [20], and [22]), iv) unlike [15], [20], and [22], this paper uses a manifold to develop an error measurement which allows for asymptotic stability of the output error, even if the modeling error (i.e., $d_\alpha(X) + d_\beta(X)u_{ce}$) does not go to zero, and v) tracking convergence is guaranteed to a boundary layer of zero using the smoothed version of the control law, as we show next.

F. Smoothing The Control Action

The sliding-mode control term, u_{si}, can introduce a high-frequency signal to the plant which may excite unmodeled dynamics. To avoid this, we now consider a "smoothed" version of the previous indirect adaptive controller in which the tracking error e_s is driven to an ϵ-neighborhood of $e_s = 0$. Using the error measurement of [38], we define

$$e_\epsilon := e_s - \epsilon \operatorname{sat}\frac{e_s}{\epsilon} \quad (86)$$

where $\epsilon > 0$ and

$$\operatorname{sat}(x) = \begin{cases} 1, & \text{if } 1 \leq x \\ x, & \text{if } -1 < x < 1. \\ -1, & \text{if } x \leq -1 \end{cases} \quad (87)$$

From the above definition, we see that e_ϵ measures the distance between e_s and the desired boundary layer, and $e_\epsilon = 0$ when e_s is within the boundary layer.

The bounding controller is now defined as

$$u_{bi} = \Pi(t)k_{bi}(t)\operatorname{sgn}(e_\epsilon) \quad (88)$$

with $\epsilon < M_e$ and $\Pi(t)$ as defined in (38). The certainty equivalence controller is redefined as

$$u_{ce} = \frac{1}{\beta_k(t) + \hat{\beta}(X)}\{-[\alpha_k(t) + \hat{\alpha}(X)] + \nu_\epsilon(t)\} \quad (89)$$

where $\nu_\epsilon(t) := y_m^{(r)} + \eta e_\epsilon + \overline{e}_s$, with $\overline{e}_s$ as defined before. Thus, (33) is now given as

$$\begin{aligned} \dot{e}_s + \eta e_\epsilon = {} & [\hat{\alpha}(X) - \alpha(X)] + [\hat{\beta}(X) - \beta(X)]u_{ce} \\ & - [\beta_k(t) + \beta(X)](u_{si} + u_{bi}). \end{aligned} \quad (90)$$

Now consider (59) with the e_ϵ as the tracking error measurement

$$V_i = \tfrac{1}{2}e_\epsilon^2 + \tfrac{1}{2}\operatorname{tr}(\Phi_\alpha^T Q_\alpha \Phi_\alpha) + \tfrac{1}{2}\operatorname{tr}(\Phi_\beta^T Q_\beta \Phi_\beta). \quad (91)$$

Taking the derivative of (91), we obtain

$$\dot{V}_i = e_\epsilon[\dot{e}_s] + \operatorname{tr}(\Phi_\alpha^T Q_\alpha \dot{\Phi}_\alpha) + \operatorname{tr}(\Phi_\beta^T Q_\beta \dot{\Phi}_\beta). \quad (92)$$

We change the update laws so that $\overline{a}_{\alpha_{i,j}}$ is the i, jth element of $z_\alpha \zeta_\alpha^T e_\epsilon$ and $\overline{a}_{\beta_{i,j}}$ is the i, jth element of $z_\beta \zeta_\beta^T e_\epsilon u_{ce}$, with the remaining projection algorithm unchanged. With this, (77) is expressed as

$$\dot{V}_i \leq -\eta e_\epsilon^2 + [|d_\alpha(X)| + |d_\beta(X)u_{ce}|]|e_\epsilon| - e_\epsilon\beta(X)u_{si}. \quad (93)$$

We now redefine the sliding-mode control term as

$$u_{si} = \frac{k_{si}(t)}{\beta_0}\operatorname{sat}\left(\frac{e_s}{\epsilon}\right) \quad (94)$$

so that we now have smooth control action. With respect to Theorem 1, we now simply use the fact that $e_\epsilon \operatorname{sat}(e_s/\epsilon) = |e_\epsilon|$ to see that $\dot{V}_i \leq -\eta e_\epsilon^2$, which ensures asymptotic stability of e_ϵ using Barbalat's Lemma. This implies that e_s will converge asymptotically to an ϵ neighborhood of $e_s = 0$, and using (43), e_o will converge to an $\epsilon\|\hat{G}_0(s)\|_1$-neighborhood of $e_o = 0$. We may additionally redefine the adaptation laws for the scheme presented within Remark 1.5 so that the sliding mode gains do not increase when the tracking error has converged to within the ϵ-neighborhood of $e_s = 0$.

IV. Direct Adaptive Control

Within this section, we define an "output-error direct adaptive controller" (using the terminology from [25]), as shown in Fig. 4. An indirect adaptive controller attempts to identify the plant dynamics and then develop a controller based on the current best guess at the plant dynamics. A direct adaptive controller, on the other hand, directly adjusts the parameters of a controller to meet some performance specifications.

In addition to the plant assumptions P1) or P2), we require the following plant assumption when using the direct adaptive controller.

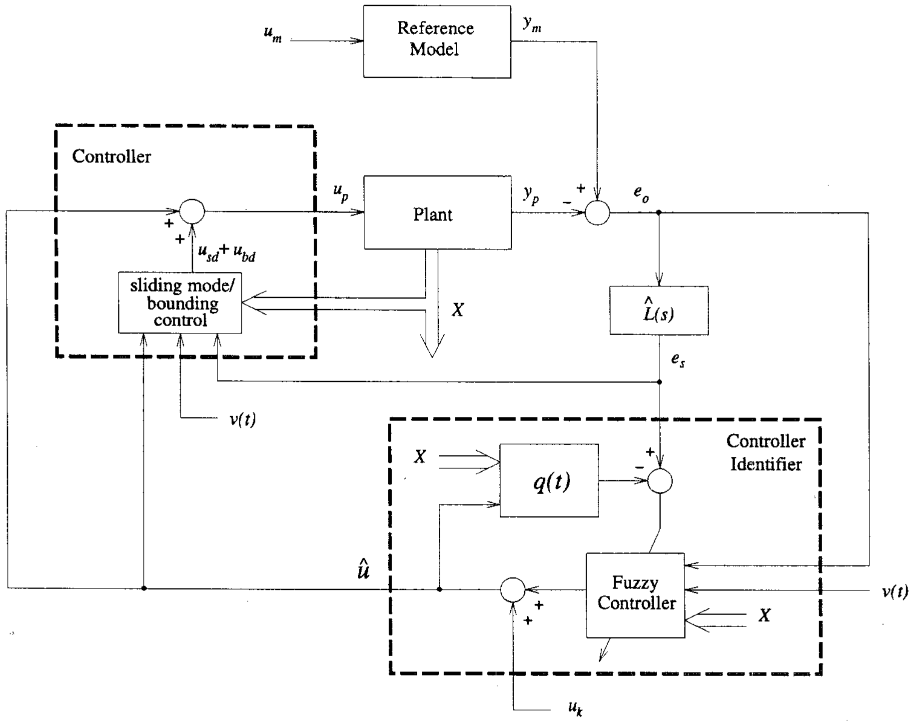


Fig. 4. A direct adaptive control system with a reference model.

P3) Plant Assumption: Given $y_p^{(r)} = [\alpha(X) + \alpha_k(t)] + [\beta(X) + \beta_k(t)]u_p$, we require that $\beta_k(t) = 0$, $t \geq 0$, and that there exists positive constants β_0 and β_1 such that $0 < \beta_0 \leq \beta(X) \leq \beta_1 < \infty$ and some function $B(X) \geq 0$ such that $|\dot{\beta}(X)| = |(\partial\beta/\partial X)\dot{X}| \leq B(X)$ for all $X \in S_x$. Here, as earlier, $\alpha_k(t)$ is a known time-dependent signal.

The first part of P3) introduces a new requirement that the controller gain $\beta(X)$ be bounded from above by a constant β_1. In general, this will not pose a large restriction upon the class of plants since situations in which a finite input will cause an infinitely large effect upon $y_p^{(r)}$ rarely occur in physical plants. The second restriction within P3) requires that $|\dot{\beta}(X)| \leq B(X)$ for some $B(X) > 0$. We know that $|\dot{\beta}(X)| \leq \|\partial\beta(X)/\partial X\| \|\dot{X}\|$ thus if $\|\partial\beta(X)/\partial X\|$ and $\|\dot{X}\|$ are bounded, then some $B(X)$ may be found. Once again if we consider physical plants with finite controller gain, then $\|\partial\beta(X)/\partial X\|$ will be bounded. If $y_p^{(i)}$, $i = 0, \cdots, r$ is bounded, then plants with no zero dynamics are ensured that $\|\dot{X}\|$ is bounded since the states may be written in terms of the outputs, $y_p^{(i)}$, $i = 0, \cdots, r-1$. If a plant has zero dynamics, but $\beta(X)$ is not dependent upon the zero dynamics, then once again we have $|\dot{\beta}(X)|$ bounded.

Using feedback linearization [25], we know that there exists some ideal controller

$$u^* = \frac{1}{\beta(X)}[-\alpha(X) + \nu(t)] \tag{95}$$

where $\nu(t)$ is a free parameter. We may express u^* in terms of a Takagi–Sugeno fuzzy model, so that

$$u^* = z_u^T A_u^* \zeta_u + u_k + d_u(X) \tag{96}$$

where u_k is a known part of the controller (possibly a fuzzy, proportional integral derivative (PID), or some other type of controller). Since the indirect adaptive controller attempted to determine a feedback linearizing controller based on a best guess of the plant dynamics, we allowed for the inclusion of $\alpha_k(t)$ and $\beta_k(t)$ so that known parts of the plant may be included. The direct adaptive controller, however, attempts to directly determine a controller, so within this section we allow for a known part of the controller that is perhaps specified via heuristics or past experience with the application of conventional direct control. We also define the ideal direct control parameters

$$A_u^* \in \Re^{m_u \times p_u}$$

$$A_u^* := \arg \min_{A_u \in \Omega_u} \left[\sup_{X \in S_x,\, \nu \in S_m} |z_u^T A_u \zeta_u - (u^* - u_k)| \right] \tag{97}$$

so that $d_u(X)$ is an approximation error which arises when u^* is represented by a fuzzy system. We assume that $D_u(X) \geq |d_u(X)|$, where $D_u(X)$ is a known bound on the error in representing the ideal controller with a fuzzy system. We see that if $|d_u(X)|$ is to be small, then our fuzzy controller will require X and ν to be available, either through the input membership functions or through z_u^T. The fuzzy approximation of the desired control is

$$\hat{u} = z_u^T A_u \zeta_u + u_k \tag{98}$$

where the matrix A_u is updated on line. The parameter error matrix for the direct adaptive controller

$$\Phi_u(t) = A_u(t) - A_u^* \tag{99}$$

is used to define the difference between the parameters of the current controller and the desired controller.

Consider the control law

$$u_p = \hat{u} + u_{sd} + u_{bd}. \tag{100}$$

The direct adaptive control law is comprised of a bounding control term, u_{bd}, a sliding-mode control term u_{sd}, and an adaptive control term, $\hat{u}$. Here, we define $\nu(t) := y_m^{(r)} + \eta e_s + \overline{e}_s - \alpha_k(t)$ with e_s and $\overline{e}_s$ as defined for the indirect adaptive controller so that $e_s = [e_o \ \cdots \ e_o^{(r-1)}][k_0 \ \cdots \ k_{r-1}]^T$ and $\overline{e}_s = [\dot{e}_o \ \cdots \ e_o^{(r-1)}][k_0 \ \cdots \ k_{r-2}1]^T$ where $\hat{L}(s) := s^{r-1} + k_{r-2}s^{r-2} + \cdots + k_1 s + k_0$ has its poles in the open left-half plane.

Using the control (100), the rth derivative of the output error becomes

$$e_o^{(r)} = y_m^{(r)} - \alpha(X) - \alpha_k(t) - \beta(X)(\hat{u} + u_{sd} + u_{bd}). \tag{101}$$

Using the definition of u^* (95) we may rearrange (101) so that

$$\begin{aligned} e_o^{(r)} =& y_m^{(r)} - \alpha(X) - \alpha_k(t) - \beta(X)u^* \\ &- \beta(X)(\hat{u} - u^*) - \beta(X)(u_{sd} + u_{bd}) \quad (102) \\ =& -\eta e_s - \overline{e}_s - \beta(X)(\hat{u} - u^*) \\ &- \beta(X)(u_{sd} + u_{bd}). \quad (103) \end{aligned}$$

We may alternatively express (103) as

$$\dot{e}_s + \eta e_s = -\beta(X)(\hat{u} - u^*) - \beta(X)(u_{sd} + u_{bd}). \tag{104}$$

We now define the bounding control term u_{bd} for the direct adaptive controller.

A. Bounding Control

The bounding control for the direct adaptive controller is determined by considering

$$v_{bd} = \tfrac{1}{2}e_s^2. \tag{105}$$

We differentiate (105) and use (104) to obtain

$$\begin{aligned} \dot{v}_{bd} =& -\eta e_s^2 - e_s[\beta(X)(\hat{u} - u^*) + \beta(X)(u_{sd} + u_{bd})] \quad (106) \\ \le& -\eta e_s^2 + |e_s|[\beta(X)(|\hat{u}| + |u^*|) + \beta(X)|u_{sd}|] \\ &- \beta(X)u_{bd}e_s. \quad (107) \end{aligned}$$

We do not explicitly know u^*, however, so the bounding controller will be implemented using $\alpha_1(X) \ge |\alpha(X)|$ as defined for the indirect adaptive controller. We choose

$$u_{bd} = \Pi(t)k_{bd}(t)\,\mathrm{sgn}\,(e_s) \tag{108}$$

where $\Pi(t)$ is as defined in (38) and

$$k_{bd}(t) = |\hat{u}| + |u_{sd}| + \frac{\alpha_1(X) + |\nu|}{\beta_0} \tag{109}$$

(we note that $|u^*| \le [\alpha_1(X) + |\nu|]/\beta_0$). With this definition of u_{bd}, we are once again guaranteed that $|e_s| \le M_e$ if the initial conditions are such that $|e_s(0)| \le M_e$. The plant assumptions *P1* or *P2* are then required so that state boundedness is guaranteed.

B. Adaptation Algorithm

Consider the following Lyapunov equation candidate

$$V_d = \frac{1}{2\beta(X)}\,e_s^2 + \frac{1}{2}\mathrm{tr}\,(\Phi_u^T Q_u \Phi_u) \tag{110}$$

where $Q_u \in \Re^{m_u \times m_u}$ is positive definite and diagonal. Since $0 < \beta_0 \le \beta(X) \le \beta_1 < \infty$, V_d is radially unbounded. The Lyapunov candidate V_d is used to describe both the error in tracking and the error between the desired controller and current controller. If $V_d \to 0$, then both the tracking and learning objectives have been fulfilled. Taking the derivative of (110) yields

$$\dot{V}_d = \frac{e_s}{\beta(X)}\,[\dot{e}_s] + \mathrm{tr}\,(\Phi_u^T Q_u \dot{\Phi}_u) - \frac{\dot{\beta}(X)e_s^2}{2\beta^2(X)}. \tag{111}$$

Substituting $\dot{e}_s$, as defined in (104), yields

$$\begin{aligned} \dot{V}_d =& \frac{e_s}{\beta(X)}\,[-\eta e_s - \beta(X)(\hat{u} - u^*) - \beta(X)(u_{sd} + u_{bd})] \\ &+ \mathrm{tr}\,(\Phi_u^T Q_u \dot{\Phi}_u) - \frac{\dot{\beta}(X)e_s^2}{2\beta^2(X)}. \end{aligned} \tag{112}$$

Now consider the following fuzzy controller update law

$$\dot{A}_u(t) = Q_u^{-1} z_u \zeta_u^T [e_s - q(t)] \tag{113}$$

where $q(t)$ is a function yet to be defined. Since $\dot{\Phi}_u = \dot{A}_u$

$$\begin{aligned} \dot{V}_d =& -\frac{\eta}{\beta(X)}\,e_s^2 - [z_u^T \Phi_u \zeta_u - d_u + u_{sd} + u_{bd}]e_s \\ &+ \mathrm{tr}\,(z_u^T \Phi_u \zeta_u)[e_s - q(t)] - \frac{\dot{\beta}(X)e_s^2}{2\beta^2(X)}. \end{aligned} \tag{114}$$

Equation (114) may equivalently be expressed as

$$\begin{aligned} \dot{V}_d =& -\frac{\eta}{\beta(X)}\,e_s^2 - q(t)z_u^T \Phi_u \zeta_u \\ &- \left[\frac{\dot{\beta}(X)e_s}{2\beta^2(X)} - d_u\right] e_s - e_s(u_{sd} + u_{bd}). \end{aligned} \tag{115}$$

Typically, we will choose $q(t) = 0$, for all $t \ge 0$, however, we will later show how to incorporate information about the plant inverse dynamics so that $\mathrm{sgn}\,[q(t)] = \mathrm{sgn}\,(z_u^T \Phi_u \zeta_u)$ to improve adaptation.

Using the fuzzy adaptation law defined by (113), we are not guaranteed that $A_u \in \Omega_u$. Once again we use a projection algorithm. The parameter space is defined so that the parameters are bounded by $A_u \in [A_u^{\min}, A_u^{\max}]$. Define $\overline{a}_{u_{i,j}}$ to be the i, jth element of $z_u \zeta_u^T [e_s - q(t)]$. The parameter matrix is updated according to

$$\dot{A}_u(t) = Q_u^{-1} \hat{A} \tag{116}$$

where the elements of $\hat{A}_u(t)$ are defined by

$$\hat{a}_{u_{i,j}} = \begin{cases} 0, & \text{if } a_{u_{i,j}} \notin (A_u^{\min}, A_u^{\max}) \\ & \quad \text{and } \overline{a}_{u_{i,j}}(a_{u_{i,j}} - a_{u_{i,j}}^c) > 0 \\ \overline{a}_{u_{i,j}}, & \text{otherwise} \end{cases} \tag{117}$$

with $A_u^c \in (A_u^{\min}, A_u^{\max})$. Using this modified update law will ensure that the parameter matrices will stay within the feasible parameter space and that

$$\dot{V}_d \leq -\frac{\eta}{\beta(X)} e_s^2 - q(t) z_u^T \Phi_u \zeta_u - \left[\frac{\dot{\beta}(X) e_s}{2\beta^2(X)} - d_u\right] e_s - e_s(u_{sd} + u_{bd}) \quad (118)$$

since the modified adaptation law guides the searching algorithm toward the optimal parameters A_u^*.

C. Sliding-Mode Control Term

We once again need to define a sliding-mode control term to compensate for the approximation error in modeling u^* by a fuzzy system or neural network. If $q(t) = 0$, for all $t \geq 0$, or $\mathrm{sgn}\,[q(t)] = \mathrm{sgn}\,(z_u^T \Phi_u \zeta_u)$, and u_{bd} is as defined in (108), then

$$\dot{V}_d \leq -\frac{\eta}{\beta(X)} e_s^2 - \left[\frac{\dot{\beta}(X) e_s}{2\beta^2(X)} - d_u\right] e_s - e_s u_{sd} \quad (119)$$

$$\leq -\frac{\eta}{\beta_1} e_s^2 + \left[\frac{|\dot{\beta}(X)|\,|e_s|}{2\beta^2(X)} + |d_u|\right] |e_s| - e_s u_{sd}. \quad (120)$$

We now define the sliding-mode control term for the direct adaptive controller as

$$u_{sd} = k_{sd}(t)\,\mathrm{sgn}\,(e_s) \quad (121)$$

where

$$k_{sd}(t) = \frac{B(X)|e_s|}{2\beta_0^2} + D_u(X) \quad (122)$$

which ensures that $\dot{V}_d \leq -\eta e_s^2/\beta_1$ as long as we choose $q(t) = 0$, for all $t \geq 0$, or $\mathrm{sgn}\,[q(t)] = \mathrm{sgn}\,(z_u^T \Phi_u \zeta_u)$.

D. Stability Properties

The controller assumption for the direct adaptive control scheme is given as follows.

C2) Control Assumption: The fuzzy systems (neural networks) are defined such that $D_u(X) \in \mathcal{L}_\infty$, for $X \in S_x \subseteq \Re^n$ and there are some known continuous functions $\alpha_1(X)$ and $\beta_1(X)$ such that $\alpha_1(X) \geq |\alpha(X)|$ and $\beta_1(X) \geq |\beta(X)|$. The function $q(t) = 0$, for all $t \geq 0$, or $\mathrm{sgn}\,[q(t)] = \mathrm{sgn}\,(z_u^T \Phi_u \zeta_u)$.

We now summarize the properties of the direct adaptive controller in Theorem 2.

Theorem 2: Stability and tracking results using direct adaptive control:

If reference input assumption R1) holds, either plant assumption P1) or P2) holds, plant assumption *P3* holds and the control law is defined by (100) with the control assumptions C2).

Then the following holds.

a) The plant output and its derivatives $y_p, \cdots, y_p^{(r-1)}$ are bounded.
b) The control signals are bounded, i.e., $u_{bd}, u_{sd}, \hat{u} \in \mathcal{L}_\infty$.
c) The magnitude of the output error $|e_o|$ decreases at least asymptotically to zero, i.e., $\lim_{t\to\infty} |e_o| = 0$.

Proof of Theorem 2: Follows from proof of Theorem 1. □

Remark 2.1: The bounding control-term u_{bd} for the direct adaptive controller is once again used to restrict the output trajectory so that a smaller fuzzy controller or neural network may be used to approximate the ideal feedback controller, u^*. The sliding-mode control term u_{sd} is required due to the modeling errors between the ideal feedback controller and the fuzzy controller or neural network with optimal parameters. The adaptive control term $\hat{u}$ is then used to ensure asymptotic convergence of the tracking error.

Remark 2.2: The direct adaptive scheme allows for the inclusion of u_k so that a control engineer may use conventional techniques to develop an initial control design and then use the above adaptive technique to work in parallel to meet the tracking requirements. For example, some PID controller design may provide moderate performance, however, the above direct adaptive technique may be used to meet tracking requirements. Even if u_k produces an unstable closed-loop system by itself, the use of the above direct adaptive scheme will result in asymptotically stable tracking.

Remark 2.3: The direct adaptive scheme does not require that the parameter set Ω_u be defined so that $z_u A_u \zeta_u + u_k$ is bounded away from a particular value as required with the indirect adaptive scheme.

Remark 2.4: A smoothed version of the direct adaptive controller may be designed such that the output will converge to an $\epsilon\|\hat{G}_0(s)\|_1$ neighborhood of $e_o = 0$. Since this is accomplished exactly the same as for the indirect case, we do not include the details here.

Remark 2.5: We may also use an adaptive estimate for some constant such that $|d_u| \leq D_u$ in a similar fashion as described in Remark 1.5. The sliding-mode gain is modified to be $k_{sd}(t) = B(X)|e_s|/2\beta_0^2 + \hat{D}_u$, where $\dot{\hat{D}}_u = |e_s|/q_u$ with $q_u^{-1} > 0$. To show stability using this adaptive algorithm for the sliding-mode gain, the term $q_u(\hat{D}_u - D_u)^2/2$ may be added to the Lyapunov candidate for the direct adaptive controller defined in (110). This may also be modified for the smoothed version of the direct adaptive controller.

Remark 2.6: As with the indirect adaptive scheme, our direct adaptive scheme has many differences from the existing techniques (i.e., those presented in [16], [17], and [21]), particularly i) the results of Theorem 2 may be applied to systems with a state dependent input gain $\beta(X)$ whereas [16] and [21] consider a class of nonlinear plants with constant input gain (i.e., $\beta(X) = \beta$; a constant) and [17] only considers the special case of $\beta = 1$, ii) none of the results in [16], [17], and [21] considered systems containing zero dynamics, iii) unlike [16], our direct adaptive algorithm ensures that even if the approximation error d_u is not square integrable, the tracking error will go to zero (or to an ϵ-boundary layer of zero for the smoothed control version), iv) our direct adaptive controller allows for Takagi–Sugeno fuzzy systems, standard fuzzy systems, or neural networks, v) the above direct adaptive technique allows for the inclusion of a known controller u_k so that it may be used to either enhance the performance of some prespecified controller or stand alone as a stable adaptive controller, and vi) furthermore, as we show in the next section,

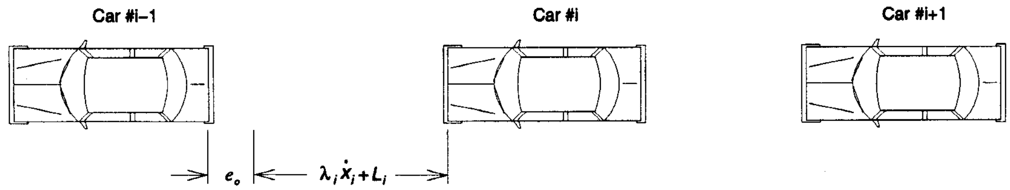


Fig. 5. Car following within an automated lane.

our approach allows for the incorporation of heuristics about the inverse plant dynamics to speed adaptation.

E. Inverse Model Linguistics

Although we typically set $q(t) = 0$ for all $t \geq 0$, there may be cases in which a control engineer is able to look at the plant output and determine if $\hat{u}$, (98), is larger or smaller than u^* [i.e., (96)]. This information may be incorporated, for example, by using a fuzzy model of the inverse plant dynamics (i.e., a fuzzy system that is heuristically designed to roughly approximate the plant's inverse dynamics). Consider a MISO fuzzy system, which is developed so that it provides a measure of the error between the actual control $\hat{u}$ and the ideal control, u^*. Then we may use a fuzzy system, for example, with rules defined as

R_1: **If** ($\tilde{e}_o$ is "Negative Large" **and** $\tilde{x}_1$ is "slow")
Then $\tilde{p}_1$ is "Positive Large"

⋮

R_p: **If** ($\tilde{e}_o$ is "Positive Large" **and** $\tilde{x}_1$ is "fast")
Then $\tilde{p}_p$ is "Negative Large."

(Note that we switch to the standard notation for the consequences of these rules. This can be done since our class of fuzzy systems includes standard fuzzy systems.) The first rule might correspond to the case in which the input u_p should be decreased since the output error $e_o = y_m - y_p$ is negative which implies that y_p is too large. Thus, the actual output is too large so $\hat{u} - u^* > 0$, implies that the output of the fuzzy system should be positive. Using fuzzy systems to describe the inverse dynamics of a plant is not a new concept. "Fuzzy model reference learning control" is, for example, one such technique which allows for the incorporation of information about the inverse dynamics of a plant to help an adaptive scheme learn how to control the plant [9].

From (118) and the definitions of u_{sd} and u_{bd}, we have

$$\dot{V}_d \leq -\frac{\eta}{\beta(X)} e_s^2 - q(t) z_u^T \Phi_u \zeta_u \tag{123}$$

which may be rewritten as

$$\dot{V}_d \leq -\frac{\eta}{\beta(X)} e_s^2 - q(t)[\hat{u} - u^* + d_u(X)]. \tag{124}$$

We want to be able to incorporate linguistic information so that the quantity $q(t)[\hat{u} - u^* + d_u(X)]$ is positive semidefinite (but not zero for all time, since if it is large, it will tend to increase the rate of convergence). Define

$$q(t) := \left\{ p(t) - \epsilon_d \,\mathrm{sat}\left[\frac{p(t)}{\epsilon_d}\right] \right\} \tag{125}$$

where $p(t)$ is the output of the inverse model, and $\epsilon_d \geq 0$ is a fixed parameter. Suppose that, for instance, p only depends on X. In this case, we denote $p(t)$ by $p(X)$ and require that $p(X)$ is defined such that if $p(X) \geq \epsilon_d$ then $\hat{u} - u^* \geq D_u(X)$ and if $p(X) \leq -\epsilon_d$ then $\hat{u} - u^* \leq -D_u(X)$. Note that if $\hat{u} - u^* \geq D_u(X)$, we know that $\hat{u} - u^* + d_u(X) \geq 0$. Similarly, $\hat{u} - u^* \leq -D_u(X)$ implies that $\hat{u} - u^* + d_u(X) \leq 0$. We may now consider three ranges of $p(X)$. If $p(X) \geq \epsilon_d$, then $q(t) \geq 0$ and $\hat{u} - u^* + d_u(X) \geq 0$, thus $q(t)[\hat{u} - u^* + d_u(X)] \geq 0$. If $-\epsilon_d < p(X) < \epsilon_d$, then $q(t) = 0$ and $q(t)[\hat{u} - u^* + d_u(X)] = 0$. If $p(X) \leq -\epsilon_d$, then $q(t) \leq 0$ and $\hat{u} - u^* + d_u(X) \leq 0$, thus, $q(t)[\hat{u} - u^* + d_u(X)] \geq 0$. Therefore, we here established that $\mathrm{sgn}\,[q(t)] = \mathrm{sgn}(z_u^T \Phi_u \zeta_u)$ so that

$$\dot{V}_d \leq -\frac{\eta}{\beta_1} e_s^2. \tag{126}$$

Thus, incorporating linguistic information into the fuzzy model of the inverse dynamics may improve controller performance since $q(t)[\hat{u} - u^* + d_u(X)]$ is positive semidefinite, causing the Lyapunov function V_d to decrease more quickly than if no linguistic information is used [i.e., $q(t) = 0$, $t \geq 0$].

V. Example: An Automated Highway System

Due to increasing traffic congestion, there has been an renewed interest in the development of an automated highway system (AHS) in which high traffic flow rates may be safely achieved. Since many of today's automobile accidents are caused by human error, automating the driving process may actually increase the safety of the highway. Vehicles will be driven automatically with onboard lateral and longitudinal controllers. The lateral controllers will be used to steer the vehicles around corners, make lane changes, and perform additional steering tasks. The longitudinal controllers will be used to maintain a steady velocity if a vehicle is traveling alone (conventional cruise control), following a lead vehicle at a safe distance (car following, see Fig. 5), or performing other speed/tracking tasks. For more details on intelligent vehicle highway systems see [39] and [40]. Within this section, we will apply the above adaptive techniques to the car following problem or longitudinal control of a vehicle within an AHS.

The dynamics of the car following system for the ith vehicle may be described by the state vector $X_i = [\delta_i, v_i, f_i]^T$ where $\delta_i = x_i - x_{i-1}$ is the intervehicle spacing between the ith and $i-$ 1st vehicles, v_i is the ith vehicle's velocity, and f_i is the driving/braking force applied to the longitudinal dynamics of the ith vehicle. The longitudinal dynamics may be expressed as

$$\dot{\delta} = v - v_{i-1} \tag{127}$$

$$\dot{v} = \frac{1}{m}(-A_\rho v^2 - d + f) \tag{128}$$

TABLE II
AUTOMOBILE VARIABLES AND PARAMETERS

x	vehicle position
v	vehicle velocity
f	applied force in longitudinal direction
$m = 1300kg$	mass of the vehicle
$A_\rho = 0.3Ns^2/m^2$	aerodynamic drag
$d = 100N$	constant frictional force
$\tau = 0.2s$	engine/brake time constant

$$\dot{f} = \frac{1}{\tau}(-f + u_p) \tag{129}$$

where u_p is the control input (if $u_p > 0$, then it represents a throttle input and if $u_p < 0$, it represents a brake input), and the vehicle variables and parameters are summarized in Table II (we assume that the variables and parameters are associated with the ith vehicle, unless subscripts indicate otherwise).

The plant output is $y_p = \delta + \lambda v$, $\lambda > 0$. This measurement allows for a velocity dependent intervehicle spacing. As the velocity of the ith vehicle increases, the distance between the ith and $i-$ 1st vehicles should increase. A standard good driving rule for humans is to allow an intervehicle spacing of one vehicle length per 10 mph (this roughly corresponds to $\lambda = 0.9$). With $\lambda \neq 0$, the plant is of relative degree two since

$$y_p^{(2)} = \dot{v} + \lambda\ddot{v} - \dot{v}_{i-1} \tag{130}$$

$$= \frac{1}{m}[-A_\rho v^2 - d + f] + \frac{\lambda}{m}\left[-2A_\rho v\dot{v} - \frac{1}{\tau}f\right] + \frac{\lambda}{m\tau}u_p - \dot{v}_{i-1}. \tag{131}$$

This is clearly of the form required by both the indirect and direct adaptive schemes [i.e., (19)] with

$$\alpha(X) = \frac{1}{m}[-A_\rho v^2 - d + f] + \frac{\lambda}{m}\left[-2A_\rho v\dot{v} - \frac{1}{\tau}f\right] \tag{132}$$

$$\beta(X) = \frac{\lambda}{m\tau} \tag{133}$$

where $\alpha_k(t) = -\dot{v}_{i-1}$ and $\beta_k(t) = 0$ for all $t \geq 0$ and for any $X \in \Re^3$. We see that $\beta(X) \geq \beta_0 > 0$ for $\beta_0 = \lambda/m_1\tau_1$, where the vehicle parameters are defined within the intervals $m \in [m_0, m_1]$ and $\tau \in [\tau_0, \tau_1]$, where $m_0, \tau_0 > 0$.

The zero dynamics are found by setting $y_p = 0$, which results in $\lambda\dot{\delta} = -\delta - \lambda v_{i-1}$. The zero dynamics are, thus, exponentially attractive since if we let $v_1 = \delta^2$, we obtain

$$\dot{v}_1 = -\frac{2}{\lambda}\delta(\delta + \lambda v_{i-1}). \tag{134}$$

If we assume that $|v_{i-1}| \leq V_m$, some bound on achievable velocities for the vehicles, then

$$\dot{v}_1 \leq -\frac{2a}{\lambda}\delta^2, \quad \text{if} \quad |\delta| \geq \left|\frac{\lambda V_m}{1-a}\right| \tag{135}$$

where $0 < a < 1$. Thus, as long as $\lambda > 0$, we are ensured exponential attractivity of the zero dynamics.

A. Indirect Adaptive Control

Since we desire that $y_p \to 0$, here we simply select $y_m := 0$ so that

$$e_o = -y_p. \tag{136}$$

Since the plant is of relative degree two, the error metric is defined as

$$e_s = \dot{e}_o + k_0 e_o. \tag{137}$$

For this example, we simply choose $k_0 = 1$ (i.e., the desired tracking eigenvalue is at -1). If e_s is to be measured, then sensors will need to obtain δ, $\dot{\delta}$, v, $\dot{v}$, and $\dot{v}_{i-1}$. With such sensors, assumption P2) is satisfied. Using the definition of $\hat{G}_i(s)$ from (42) we see that $\|\hat{G}_0(s)\|_1 = 1/k_0 = 1$. Thus if we want the bounding control term to be defined such that $|e_o| \leq$ 1 meter, we use (43) to pick $M_e = 1$ and $\epsilon_M = 0.1$. Ideally, we will not need to use the bounding controller unless the initial conditions are such that $e_o \geq 1$. If the following vehicle becomes too close to the lead vehicle, however, the bounding controller may be used to ensure that the two vehicles do not collide. The upper bounds on $\alpha(X)$ and $\beta(X)$ are found from

$$|\alpha(X)| \leq \frac{A_\rho}{m}(|v| + 2\lambda|\dot{v}|)|v| + \frac{|d|}{m} + \frac{1}{m}\left(1 + \frac{\lambda}{\tau}\right)|f| \tag{138}$$

$$|\beta(X)| \leq \frac{\lambda}{m\tau}. \tag{139}$$

Using bounds on the vehicle parameters, we obtain

$$|\alpha(X)| \leq \frac{A_{\rho_1}}{m_0}(|v| + 2\lambda|\dot{v}|)|v| + \frac{|d_1|}{m_0} + \frac{1}{m_0}\left(1 + \frac{\lambda}{\tau_0}\right)|f| := \alpha_1(X) \tag{140}$$

$$|\beta(X)| \leq \frac{\lambda}{m_0\tau_0} := \beta_1(X). \tag{141}$$

where $A_\rho \in [A_{\rho_0}, A_{\rho_1}]$ and $d \in [d_0, d_1]$, with $A_{\rho_0}, d_0 \geq 0$. If vehicle variable bounds are known, these may be used within (140) and (141) rather than the instantaneous variable values.

Here, we will consider six rules in the fuzzy system $\hat{\alpha}(X)$ to approximate $\alpha(X)$

R_1: If vel is slow **and** acc is neg
Then $c_1 = a_{1,0} + a_{1,1}v^2$
R_2: If vel is med **and** acc is neg
Then $c_2 = a_{2,0} + a_{2,1}v^2$
R_3: If vel is fast **and** acc is neg
Then $c_3 = a_{3,0} + a_{3,1}v^2$
R_4: If vel is slow **and** acc is pos
Then $c_4 = a_{4,0} + a_{4,1}v^2$
R_5: If vel is med **and** acc is pos
Then $c_5 = a_{5,0} + a_{5,1}v^2$
R_6: If vel is fast **and** acc is pos
Then $c_6 = a_{6,0} + a_{6,1}v^2$

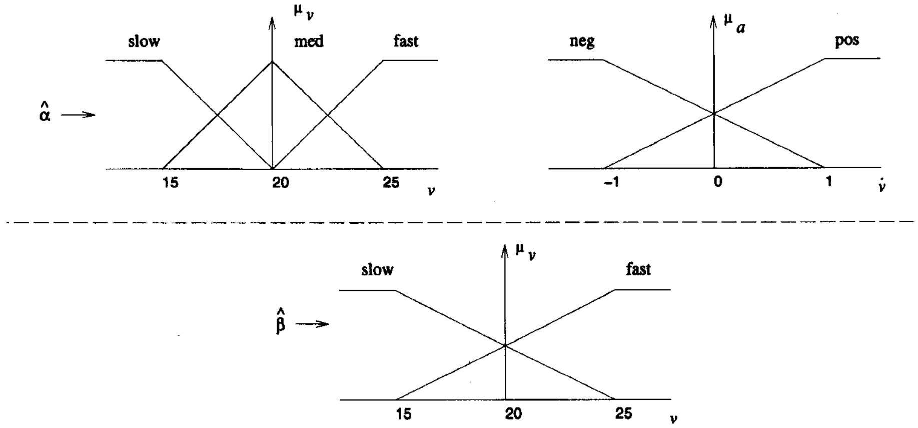


Fig. 6. Membership functions for the indirect adaptive routine.

so that $z_\alpha = [1, v^2]^T$, $A_\alpha \in \Re^{2\times 6}$, and $\zeta_\alpha \in \Re^6$. We chose $\theta_1(X) = v^2$ in (7) since we expect the longitudinal dynamics of the vehicle to depend upon v^2 when looking at (129). We use two rules to approximate $\beta(X)$

R_1: **If** vel is slow **Then** $c_1 = a_{1,0}$

R_2: **If** vel is fast **Then** $c_2 = a_{2,0}$

so that $z_\beta = [1]$, $A_\beta \in \Re^{1\times 2}$, and $\zeta_\beta \in \Re^2$. We use this simple fuzzy system since $\hat{\beta}$ is only approximating the constant β. The t-norm was taken as the product operator (3) for both fuzzy systems. The membership functions are shown in Fig. 6 where standard linguistic variables ("vel" and "acc") and linguistic values ("slow," "med," etc.) are used in R_1–R_6 and to label the membership functions. Triangular membership functions were chosen in this example even though any standard membership functions may have been used.

Since $\bar{e}_s = \dot{e}_s - e_o^{(2)} = k_0\dot{e}_o$, we have $\nu(t) = e_s + \bar{e}_s = 2\dot{e}_o + e_o$, using $\eta = 1$. The smoothed version of the adaptive controller described in Section III-F is used to help avoid overactuation of the throttle and brakes. We chose a value of $\epsilon = 0.1$. To determine the possible tracking error, we note that $\|G_0(s)\|_1 = 1$, so that e_o will asymptotically converge to $|e_o| \le 0.05$. In addition, we need to specify bounds for the parameter matrix A_β. Since $A_\beta\zeta_\beta \ge A_\beta^{\min}\zeta_\beta$ with each element $a_{\beta_{i,j}}^{\min} = \beta_{\min} > 0$ a positive constant, we choose $a_{\beta_{i,j}}^{\min} = 0.001$ and $a_{\beta_{i,j}}^{\max} = 1.0$ for all i, j. This will ensure that $\hat{\beta}(X) \ge \beta_0 = \beta_{\min} > 0$ as required by our controller assumption, *C1*. No bounds are placed on the parameters A_α. In addition, the initial conditions are picked such that $a_{\alpha_{i,j}} = 0$ and $a_{\beta_{i,j}} = 0.02$ for all i, j. The sliding mode gain was taken as $k_{si}(t) = 0.1 + 0.001|u_{ce}|$ [i.e., $D_\alpha(X) \le 0.1$ and $D_\beta(X) \le 0.001$]. These values were picked after taking into consideration the size of $\alpha(X)$ and $\beta(X)$. Since $\alpha(X)$ is proportional to $1/m$, where the vehicle mass m is large, the error in representing $\alpha(X)$ with a fuzzy system should not be greater than 0.1. In addition, since $\beta(X)$ is a constant, a fuzzy system with a single membership function can represent $\beta(X)$ exactly, however, here we choose two membership functions for the sake of illustration. The fuzzy system update parameters and other parameters for the adaptive controller are summarized in Table III. Notice that the elements of Q_α^{-1} and Q_β^{-1} are small since the functions $\alpha(X)$ and $\beta(X)$ are typically small.

TABLE III
CONTROL PARAMETERS FOR THE DIRECT AND INDIRECT ADAPTIVE ALGORITHMS

$M_e = 1$	$k_0 = 1$	$\eta = 1$
$\epsilon_M = 0.1$	$\epsilon = 0.05$	$q(t) = 0$
$Q_\alpha^{-1} = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$	$Q_\beta^{-1} = \begin{bmatrix} 0.01 \end{bmatrix}$	$Q_u^{-1} = \begin{bmatrix} 500 & 0 \\ 0 & 500 \end{bmatrix}$

Within our simulation, the following car attempted to track a lead car during a series of accelerations and decelerations (note that similar results are obtained if the controller is used on N vehicles in a "platoon"). The velocity profiles for the lead and following vehicles are shown in Fig. 7 (note that the deceleration of the lead vehicle is abrupt enough so that braking is required by the following vehicle). The output error e_o during the simulation is shown in Fig. 8. The indirect adaptive scheme is able to quickly control the intervehicle spacing, allowing $|e_o|$ to increase only slightly above $\epsilon = 0.05$ at any given time. The control inputs for the lead and following cars are shown in Fig. 9. We notice that the control action for the following car using the above indirect adaptive routine is smooth and similar to the control action applied to the lead car. Finally, if the reader is concerned with "slinky effects" (i.e., where spacing and velocity deviations are propagated along a string of N vehicles) consult, e.g., [41].

B. Direct Adaptive Control

We will now show how to apply the direct adaptive control scheme to the car following problem. Plant assumption *P3* is clearly satisfied since $\beta(X)$ is a constant so that $\dot{\beta}(X) = 0$. We define the known controller to be a simple proportional controller

$$u_k = k_p e_o \tag{142}$$

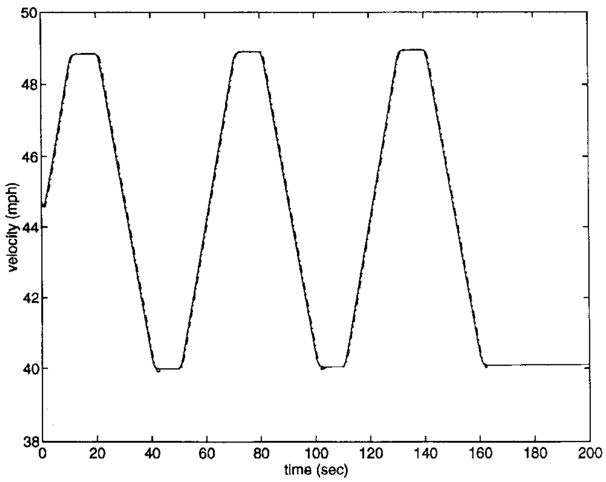


Fig. 7. Velocity profiles of the lead car (______) and following car (···).

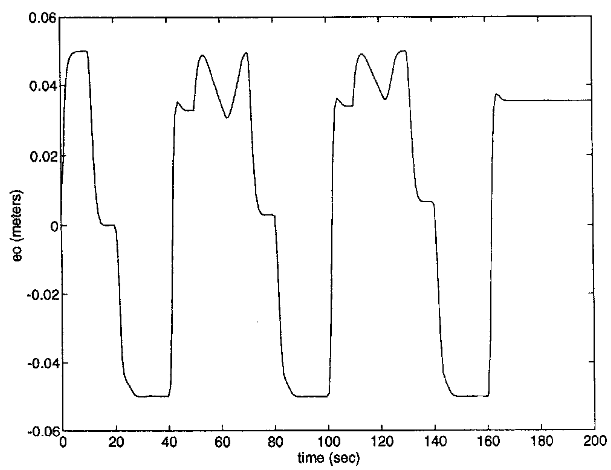


Fig. 8. Output error e_0 for the car following problem using indirect adaptive control.

with $k_p = 100$. The rule base for the fuzzy controller that generates $\hat{u}$ in (98) is

R_1: **If** vel is slow **and** e_s is neg
Then $c_1 = a_{1,0} + a_{1,1}\nu(t)$
R_2: **If** vel is med **and** e_s is neg
Then $c_2 = a_{2,0} + a_{2,1}\nu(t)$
R_3: **If** vel is fast **and** e_s is neg
Then $c_3 = a_{3,0} + a_{3,1}\nu(t)$
R_4: **If** vel is slow **and** e_s is zero
Then $c_4 = a_{4,0} + a_{4,1}\nu(t)$
R_5: **If** vel is med **and** e_s is zero
Then $c_5 = a_{5,0} + a_{5,1}\nu(t)$
R_6: **If** vel is fast **and** e_s is zero
Then $c_6 = a_{6,0} + a_{6,1}\nu(t)$
R_7: **If** vel is slow **and** e_s is pos

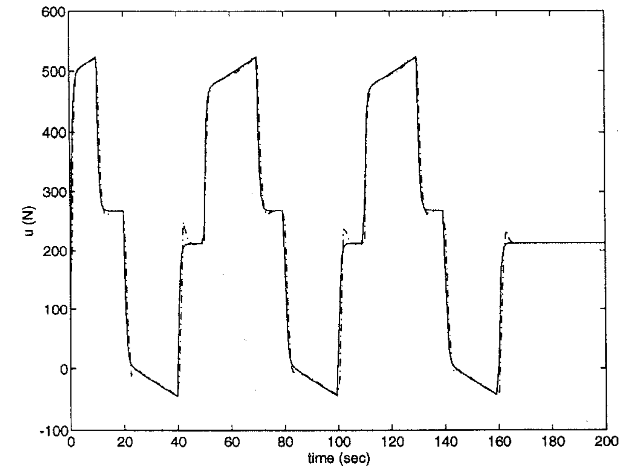


Fig. 9. Control input for the lead car (______) and the following car (···).$\hat{u}$ in (98) is

Then $c_4 = a_{7,0} + a_{7,1}\nu(t)$
R_8: **If** vel is med **and** e_s is pos
Then $c_5 = a_{8,0} + a_{8,1}\nu(t)$
R_9: **If** vel is fast **and** e_s is pos
Then $c_6 = a_{9,0} + a_{9,1}\nu(t)$

so that $z_u = [1, \nu(t)]^T$, $A_u \in \Re^{2\times 9}$, and $\zeta_u \in \Re^9$, with $\nu(t) = 2\dot{e}_o + e_o + \dot{v}_{i-1}$ [recall that $r = 2$, $k_0 = 1$, $y_m = 0$, and $\alpha_k(t) = -\dot{v}_{i-1}$]. Here, we picked $\nu(t)$ so that the fuzzy system may approximate u^* with a small number of rules since u^* is directly a function of $\nu(t)$. We once again use the smoothed version of the adaptive control. The t-norm was taken as the minimum operator as in (2). The membership functions were defined as Gaussian membership functions, as shown in Fig. 10 (the Gaussian membership functions are defined as in Table I with $\sigma = 2.5$ and 0.5 for μ_{vel} and μ_{e_s}, respectively).

The bounding functions $\alpha_1(X)$ and $\beta_1(X)$ are defined as before. Here we assume that our adaptive system when properly tuned should be able to approximate the ideal control to within 100 Newtons, so $D_u(X) = 100$. The projection limits were chosen to be $a_{u_{i,j}}^{\min} = -2000$ and $a_{u_{i,j}}^{\max} = 2000$ for all i, j. The adaptation parameters were picked to be the same as for the indirect adaptive case (see Table III).

Using the direct adaptive controller, we once again tried to control the following vehicle to track the lead vehicle. The velocity profiles for the lead and following vehicles are shown in Fig. 11. The output error, e_o, is shown in Fig. 12. Even though the output error is slightly larger using the direct adaptive routine than that for the indirect adaptive routine, it is of the order of magnitude for which we designed our adaptive controller. The control input for the lead and following vehicles is shown in Fig. 13. We see that although the controller required a few seconds to "learn" how to control the vehicle, the output of the following vehicle quickly matched the output of the lead vehicle. Fig. 14 shows how well the known control portion, u_k, is able to track the position of the lead vehicle when no adaptive control portion is included,

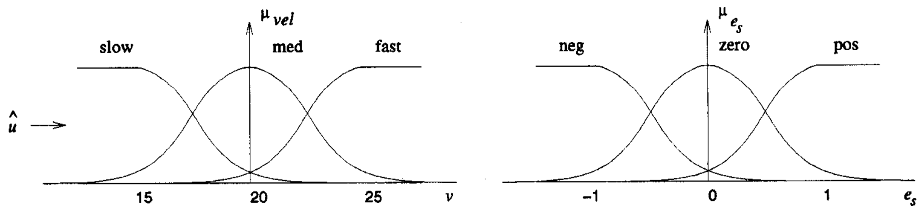


Fig. 10. Membership functions for the direct adaptive controller.

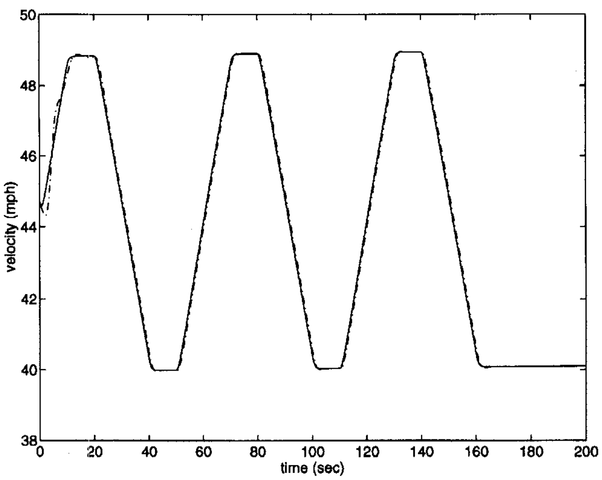


Fig. 11. Velocity profiles of the lead car (——) and following car (· · ·).

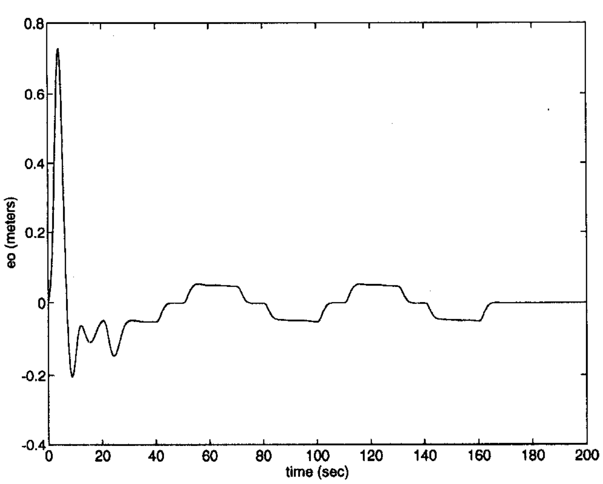


Fig. 12. Output error e_0 for the car following problem using direct adaptive control.

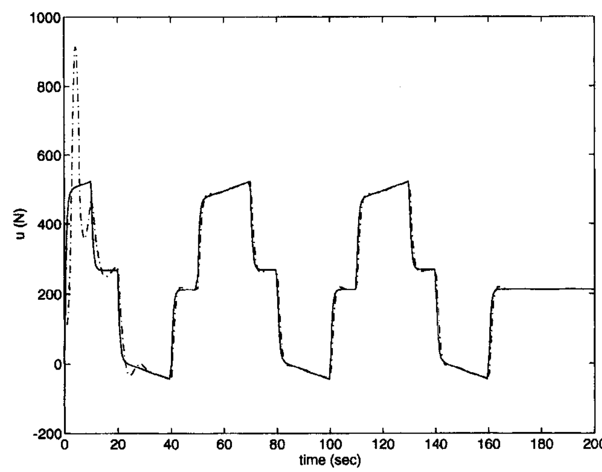


Fig. 13. Control input for the lead car (——) and the following car (· · ·).

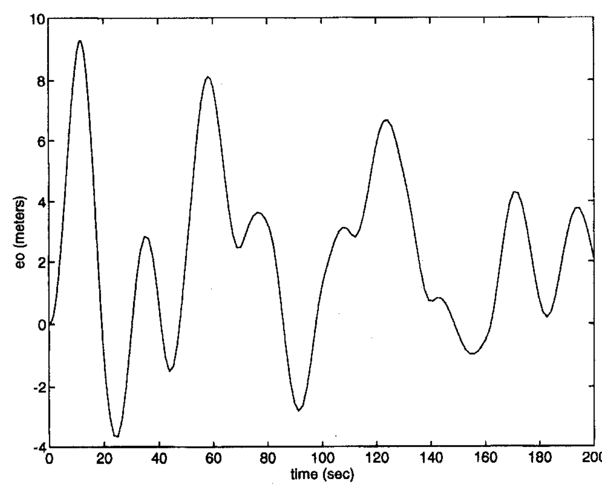


Fig. 14. Tracking performance e_o when the direct adaptive control is not used, i.e., $u = u_k$.

i.e., $u_p = u_k$. Even though u_k is apparently able to stabilize the system, the control results in very poor performance which illustrates the advantage of using the adaptation algorithm.

VI. Concluding Remarks

Indirect and direct adaptive control schemes were presented for a class of continuous-time nonlinear plants, conditions were provided for their stable operation, and we showed how they could be used for the longitudinal control of a vehicle in an automated lane. The adaptive schemes are particularly powerful since they: i) do not require a complete model of the plant, ii) use a general Takagi–Sugeno fuzzy system that includes a certain class of fuzzy systems and neural networks as a special case (note that for the indirect scheme, if desired, one can use a neural network for estimating $\beta(X)$ and a fuzzy system for $\alpha(X)$ or vice versa), and iii) allow for the inclusion of *a priori* knowledge in the form of mathematical equations or heuristics to be loaded into the identifier (indirect case) or controller (direct case). Consider, for example, the case where a feedback linearizing controller may be designed based on the nominal plant. The indirect adaptive controller may then be used basing α_k and β_k on the linearization of the nominal plant, then using the fuzzy controller to compensate for unknown parameters or unmodeled dynamics. Another possibility is to design a linear quadratic regulator (LQR) based on a linear representation of a nonlinear plant. Setting z_u equal to the plant states, it is possible to use the direct adaptive controller as an adaptive LQR, by assigning each column of A_u to the coefficients found for the standard LQR. Each rule may be used to define the space over which each "local LQR" is to adapt. These are just two examples in which the techniques presented here may be used to enhance existing control designs. An upcoming paper [42] will show how each of the above ideas can be used in the implementation of the adaptive fuzzy controllers presented in this paper for a variety of applications.

The indirect and direct adaptive fuzzy controllers both allow for Takagi–Sugeno fuzzy systems, require the same inputs, and provide the same stability results for the same class of plants, however, there are some differences.

- The direct adaptive controller requires a known $B(X) > |\dot{\beta}(X)|$.
- The direct adaptive controller does not require a projection algorithm. For the direct adaptive fuzzy controller, the adjustable parameters are guaranteed to be bounded because of the choice of our Lyapunov function, V_d.
- The direct adaptive controller requires only a single fuzzy system, and there may be resulting computational advantages if the control update rate is a limiting factor for your implementation. For the indirect adaptive controller, even though there is the need for two fuzzy systems, one can use the same antecedents for the rule bases used to approximate $\alpha(X)$ and $\beta(X)$ (different consequence values, i.e., A_α and A_β, will still be used). This will speed up the implemented indirect adaptive controller considerably since once the implied degree of membership for each rule is determined for one fuzzy system, you have done most of the work to calculate the output of the other fuzzy system. For more details, see the work in [42].
- It has been our experience that the direct adaptive controller is easier to adjust in simulations and in the laboratory. It should be emphasized, however, this is only based on personal observations with a limited number of experiments. For more details, see the work in [42].

While the adaptive control schemes present significant advantages, they have several drawbacks which cannot be overlooked. First, as with all mathematical stability analysis, one proves stability of the model of the closed-loop system and *not* the dynamics of the actual physical system. If the model of the physical system is reasonably accurate, then the stability of the algorithms will take on real physical meanings; however, if the model is inaccurate [e.g., the plant is only accurately represented by $\dot{X} = \bar{f}(X, t, u_p)$, $y_p = h(X)$ and cannot be split so that $\dot{X} = f(X) + g(X)u_p$, $y_p = h(X)$], the stability analyzes are certainly of limited value in verifying the properties of the physical control system. Furthermore, we do constrain, somewhat, the types of heuristic information that can be included so that we can guarantee stability of our adaptive systems (e.g., we do not allow for the inclusion of heuristics to tune the membership functions in the antecedents of the rule as done in [11] and [15]). It is important to remember that the Lyapunov approach used here provides *sufficient* conditions for stability; it is possible that some simpler heuristic approach to adaptive fuzzy control could provide stable operation of the closed-loop system and it may be difficult to use the somewhat conservative Lyapunov stability analysis approach to verify its stability (this is a standard problem in nonlinear analysis).

Second, our adaptive schemes are only for continuous-time SISO nonlinear systems of a specific form. Extensions to discrete-time and MIMO nonlinear systems is an important direction. Third, there may be other adaptive schemes that do not rely as heavily on the need to measure certain plant variables (of course other approaches, i.e., [15], [20], have similar reliance on plant information). It may be possible, however, that if certain variables are not available, then the approximation errors d_α, d_β or d_u may simply become larger while still providing stable closed-loop control. Fourth, while the implementation of our adaptive schemes is certainly feasible, since not too many computational resources are needed (consider, e.g., the complexity of the controllers for the vehicle following problem), there may exist simpler approaches that are perhaps more desirable from an implementation standpoint. Fifth, many of the heuristic fuzzy and neural adaptive control approaches have been studied in experimental testbeds; similar experimental analysis of the adaptive schemes in this paper are needed but are beyond the scope of the present study.

References

[1] T. Procyk and E. Mamdani, "A linguistic self-organizing process controller," *Automatica,* vol. 15, no. 1, pp. 15–30, 1979.

[2] W. Pedrycz, *Fuzzy Control and Fuzzy Systems,* 2nd ed. New York: Wiley, 1993.

[3] M. Sugeno, Ed., *Industrial Applications of Fuzzy Control.* Amsterdam: Elsevier Science, 1985.

[4] A. Kandel and G. Langholz, *Fuzzy Control Systems.* Boca Raton, FL: CRC Press, 1993.

[5] D. Driankov, H. Hellendoorn, and M. M. Reinfrank, *An Introduction to Fuzzy Control.* Berlin Heidelberg: Springer-Verlag, 1993.

[6] R. J. Marks, II, Ed., *Fuzzy Logic Technology and Applications.* New York: IEEE Press, 1994.

[7] P. J. Antsaklis and K. M. Passino, Eds., *An Introduction to Intelligent and Autonomous Control.* Norwell, MA: Kluwer, 1993.

[8] J. R. Layne, K. M. Passino, and S. Yurkovich, "Fuzzy learning control for anti-skid braking systems," *IEEE Trans. Contr. Syst. Tech.,* vol. 1, pp. 122–129, June 1993.

[9] J. R. Layne and K. M. Passino, "Fuzzy model reference learning control for cargo ship steering," *IEEE Contr. Syst. Mag.,* vol. 13, pp. 23–34, Dec. 1993.

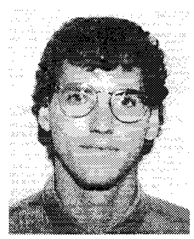
[10] ——, "Fuzzy model reference learning control," *J. Intelligent Fuzzy Syst.,* to be published.

[11] W. A. Kwong and K. M. Passino, "Dynamically focused fuzzy learning control," *IEEE Trans. Syst., Man, Cybern.,* vol. 26, pp. 53–74, Feb. 1996.

[12] V. G. Moudgal, W. A. Kwong, K. M. Passino, and S. Yurkovich, "Fuzzy learning control for a flexible-link robot," *IEEE Trans. Fuzzy Syst.,* vol. 3, pp. 199–210, May 1995.

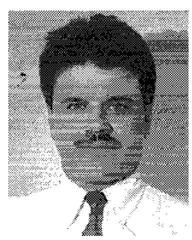
[13] W. A. Kwong, K. M. Passino, E. G. Lauknonen, and S. Yurkovich, "Expert supervision of fuzzy learning systems for fault tolerant aircraft control," in *Proc. IEEE, Spec. Iss. Fuzzy Logic Eng. Applicat.,* vol. 83, pp. 466–483, Mar. 1995.

[14] G. Langari and M. Tomizuka, "Stability of fuzzy linguistic control systems," in *Proc. 29th Conf. Decision Contr.*, Honolulu, HI, Dec. 1990, vol. 4, pp. 2185–2190.
[15] L.-X. Wang, *Adaptive Fuzzy Systems and Control: Design and Stability Analysis.* Englewood Cliffs, NJ: Prentice Hall, 1994.
[16] ——, "Stable adaptive fuzzy control of nonlinear systems," in *Proc. 31st Conf. Decision Contr.*, Tucson, AZ, Dec. 1992, pp. 2511–2516.
[17] C.-Y. Su and Y. Stepanenko, "Adaptive control of a class of nonlinear systems with fuzzy logic," *IEEE Trans. Fuzzy Syst.*, vol. 2, pp. 285–294, Nov. 1994.
[18] T. A. Johansen, "Fuzzy model based control: Stability, robustness, and performance issues," *IEEE Trans. Fuzzy Syst.*, vol. 2, pp. 221–234, Aug. 1994.
[19] K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Trans. Neural Networks*, vol. 1, pp. 4–27, 1990.
[20] M. M. Polycarpou and P. A. Ioannou, "Identification and control of nonlinear systems using neural network models: Design and stability analysis," Elect. Eng.—Syst. Rep. 91-09-01, Univ. Southern California, Los Angeles, Sept. 1991.
[21] R. M. Sanner and J.-J. E. Slotine, "Gaussian networks for direct adaptive control," *IEEE Trans. Neural Networks*, vol. 3, pp. 837–863, 1992.
[22] A. Yeşildirek and F. L. Lewis, "A neural network controller for feedback linearization," in *Proc. 33rd Conf. Decision Contr.*, Lake Buena Vista, FL, Dec. 1994, pp. 2494–2499.
[23] F.-C. Chen and H. K. Khalil, "Adaptive control of nonlinear systems using neural networks," *Int. J. Control*, vol. 55, pp. 1299–1317, 1992.
[24] G. A. Rovithakis and M. A. Christodoulou, "Adaptive control of unknown plants using dynamical neural networks," *IEEE Trans. Syst. Man, Cybern.*, vol. 24, pp. 400–412, Mar. 1994.
[25] S. Sastry and M. Bodson, *Adaptive Control: Stability, Convergence, and Robustness.* Englewood Cliffs, NJ: Prentice Hall, 1989.
[26] L.-X. Wang, "Design and analysis of fuzzy identifiers of nonlinear dynamic systems," *IEEE Trans. Automat. Contr.*, vol. 40, pp. 11–23, Jan. 1995.
[27] V. I. Utkin, "Variable structure systems with sliding modes," *IEEE Trans. Automat. Contr.*, vol. 22, pp. 212–222, Apr. 1977.
[28] D. G. Taylor, P. V. Kokotovic, R. Marino, and I. Kanellakopoulos, "Adaptive regulation of nonlinear systems with unmodeled dynamics," *IEEE Trans. Automat. Contr.*, vol. 34, pp. 405–412, Apr. 1989.
[29] J. T. Spooner and K. M. Passino, "Stable adaptive fuzzy control for an automated highway system," in *Proc. IEEE Int. Symp. Intell. Contr.*, Monterey, CA, 1995, pp. 531–536.
[30] ——, "Stable indirect adaptive control using fuzzy systems and neural networks," in *Proc. IEEE Conf. Decision Contr.*, New Orleans, LA, 1995, pp. 243–248.
[31] ——, "Stable direct adaptive control using fuzzy systems and neural networks," in *Proc. IEEE Conf. Decision Contr.*, New Orleans, LA, 1995, pp. 249–254.
[32] ——, "Stable adaptive control using fuzzy systems and neural networks," IVHS-OSU Rep. 95-01, Ohio State Univ., Feb. 1995.
[33] K. M. Passino and S. Yurkovich, "Fuzzy control," in *The Control Handbook*, W. Levine, Ed. Boca Raton, FL: CRC Press, 1996, pp. 1001–1017.
[34] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its application to modeling and control," *IEEE Trans. Syst. Man, Cybern.*, vol. 15, pp. 116–132, Jan. 1985.
[35] J. Hertz, A. Krogh, and R. G. Palmer, *Introduction to the Theory of Neural Computation.* Reading, MA: Addison-Wesley, 1991.
[36] S. S. Sastry and A. Isidori, "Adaptive control of linearizable systems," *IEEE Trans. Automat. Contr.*, vol. 34, pp. 1123–1131, Nov. 1989.
[37] L.-X. Wang, "A supervisory controller for fuzzy control systems that guarantees stability," *IEEE Trans. Automat. Contr.*, vol. 39, pp. 1845–1847, Sept. 1994.
[38] J.-J. E. Slotine and J. A. Coetsee, "Adaptive sliding controller synthesis for nonlinear systems," *Int. J. Control*, vol. 43, no. 6, pp. 1631–1651, 1986.
[39] R. E. Fenton and R. J. Mayhan, "Automated highway studies at The Ohio State University—An overview," *IEEE Trans. Veh. Technol.*, vol. 40, pp. 100–113, Feb. 1991.
[40] S. E. Shladover, C. A. Desoer, J. K. Hedrick, M. Tomizuka, J. Walrand, W.-B. Zhang, D. H. McMahon, H. Peng, S. Sheikholeslam, and N. McKeown, "Automatic vehicle control developments in the PATH program," *IEEE Trans. Veh. Technol.*, vol. 40, pp. 114–130, Feb. 1991.
[41] J. T. Spooner and K. M. Passino, "Fault tolerant longitudinal and lateral control for automated highway systems," in *Proc. '95 Amer. Contr. Conf.*, Seattle, WA, June 1995, pp. 663–667.
[42] R. Ordó nez, J. Zumberge, J. T. Spooner, and K. M. Passino, "Adaptive fuzzy control: Experiments and comparative analyzes," to be published.



Jeffrey T. Spooner received the B.A. degree in physics from Wittenberg University, Springfield, OH, in 1991, the B.S. degree in electrical engineering from The Ohio State University, Columbus, in 1993, and the M.S. degree in electrical engineering with a specialization in control systems, in 1995.

Since 1995, he has been with the Control Subsystems Department at Sandia National Laboratories, Albuquerque, NM. His areas of interest include adaptive control, fuzzy systems, neural networks, and modeling.



Kevin M. Passino (S'79–M'90–SM'96) received the B.S.E.E. degree from Tri-State University, Angola, IN, in 1983, and the M.S. and Ph.D. degrees in electrical engineering from the University of Notre Dame, IN, in 1989.

He has worked in the control systems group at Magnavox Electronic Systems Co., Ft. Wayne, IN, on research in missile control, and at McDonnell Aircraft Co., St. Louis, MO, on research in flight control. He spent a year at Notre Dame as a Visiting Assistant Professor and is currently an Associate Professor in the Department of Electrical Engineering at the Ohio State University. He is co-editor (with P. J. Antsaklis) of the book *An Introduction to Intelligent and Autonomous Control* (Kluwer Academic Press, 1993). He is an Associate Editor for the IEEE TRANSACTIONS ON AUTOMATIC CONTROL, served as the Guest Editor for the 1993 *IEEE Control Systems Magazine Special Issue on Intelligent Control*, a Guest Editor for a special track of papers on *Intelligent Control for IEEE Expert Magazine* in 1996, and is on the Editorial Board of the *International Journal for Engineering Applications of Artificial Intelligence*. His research interests include intelligent and autonomous control techniques, nonlinear analysis of intelligent control systems, failure detection and identification systems, and genetic algorithms for control.

He is a member of the IEEE Control Systems Society Board of Governors. He is also the Publicity Co-Chair for the IEEE Conference on Decision and Control in Japan in 1996 and is the Workshops Chair for the 1997 IEEE Conference on Decision and Control. He was a Program Chairman for the 8th IEEE International Symposium on Intelligent Control (1993). He served as the Finance Chair for the 9th IEEE International Symposium on Intelligent Control, and he is serving as the General Chair for the 11th IEEE International Symposium on Intelligent Control.