A hybrid adaptive fuzzy tracking control for robotic systems

Abdel Badie Sharkawy
Mechanical Eng. Dept., Faculty of Eng., Assiut University, 715 16 Assiut, Egypt
e-mail: ab.shark@aun.edu.eg

In this article, a new hybrid adaptive fuzzy controller is derived for robotic systems. The sliding mode control concept is combined with fuzzy control strategy to design a model-free adaptive fuzzy sliding mode control. The equivalent controller has been substituted for by adaptive fuzzy system and the uncertainties are estimated on-line. The approach has the learning ability to generate the fuzzy control actions and adaptively compensates for the uncertainties. Simulations have been carried out on a two link planar robot. Despite the high nonlinearity and coupling effects, the control input has been decoupled leading to a simplified adaptive mechanism for robotic systems. Results show the effectiveness of the proposed control system.

يهدف البحث الى أستنتاج وتحليل نظام تحكم ثنائى متأقلم للمناولات الآليه. النظام المقترح يستخدم مفهوم التحكم بأستعمال المستويات المنزلقه مع الأنظمه الهلاميه ليكون نظاما جديدا للتحكم متأقلم و لايعتمد على معرفه مسبقه بالنموذج الرياضى. فقد تم الأستعاضه عن مركبات التحكم بأخرى أحدها هلاميه والثانيه غير هلاميه و يمكن تحديد معالمهما وتأقلمهما أثناء عمل المناول الآلى. وبتطبيق هذا النظام على مناول آلى ثنائى الأزرع أظهرت النتائج قدرات ممتازه لنظام التحكم المقترح.

Keywords: Sliding mode control, Hybrid adaptive fuzzy control, Fuzzy logic control, Adaptive laws, Robotic control

1. Introduction

Performance of many tracking control systems is limited by variation of parameters and disturbances. This specially applies for direct drive robots with highly nonlinear dynamics and model uncertainties. Payload changes and/or its exact position in the end effector are examples of uncertainties. The control methodologies that can be used are ranging from classical adaptive control and robust control to the new methods that usually combine good properties of the classical control schemes to fuzzy [1], genetic [2] and neural network [3,4] based approaches. Classical adaptive control of manipulators requires a precise mathematical model of the system's dynamics and the property of linear parameterization of the system's uncertain physical parameters [5].

The study of output tracking problems has a long-standing history. Sliding Mode Control (SMC) is often favored basic control approach, because the insensitivity to parametric uncertainties and external disturbances [6-8]. The theory is based on the concept of changing the structure of the controller to achieve a desired response of the system. By using a variable high speed switching feedback gain, the trajectory of the system can be forced on a chosen manifold, which is called sliding surfaces or switching surfaces, and remains thereafter. The design of proper switching surfaces to obtain the desired performance of the system is very important and has been the topic of many previous works [9-11]. With the desired switching surface, we need to design a SMC such that any state out side the switching surface can be driven to the switching surface in finite time. Generally, in the SMC design, the uncertainties are assumed to be bounded. This assumption may be reasonable for external disturbance, but it is rather restrictive as far as unmodelled dynamics are concerned.

Nowadays, Fuzzy Logic Control (FLC) systems have been proved to be a promising approach to solve complex nonlinear control problems. They provide an effective means to capture the approximate nature of real world. Examples are numerous; see [12] for instance. While non-adaptive fuzzy control has proven its value in some applications [13], it is sometimes difficult to specify the rule base for some plants, or the need could arise to tune the rule-base parameters if the plant changes. This provides the motivation for adaptive fuzzy control, where the focus is on the automatic on-line synthesis and tuning of fuzzy controller parameters (i.e., the use of on-line data to continually "learn" the fuzzy controller, which will ensure that the performance objectives are met).

Recently, adaptive FLC design has drawn much attention of many researchers. Palm [14] and Wang [15] employed a switch function as the input fuzzy variables and proposed a fuzzy sliding mode controller. Lu and Chen [16] extended this approach and developed a self organized fuzzy sliding mode controller to smooth the chattering phenomenon. They employed a fuzzy system adjusted by an adaptive law to approximate an optimal controller to a specified accuracy based on the Lyapunov stability theory. However, this kind of direct adaptive law is limited to nonlinear system with constant control gain. After that, Chai and Tong [17] proposed a fuzzy direct control scheme by using a fuzzy system to approximate an optimal controller that was designed based on the assumption that all of the system's dynamics were known. Then a fuzzy sliding mode controller was added to the adaptive controller to compensate for the uncertainties and smoothing the control signal. Ting et al. [18] employed fuzzy sliding mode controller for the active suspension system to investigate the ride comfort. Their approach however, depends on a certain model for a sliding mode to operate. In addition, the database of the fuzzy system is still complicated. In the work of Hsu et al. [19], the fuzzy data base is complicated and expert knowledge is still needed to enhance the performance.

In this work, a Hybrid Adaptive Fuzzy Control (HAFC) is proposed for robotic systems. The scheme is based on the universal approximation property of fuzzy systems and the powerfulness of SMC theory. A one dimensional adaptive FLC is designed to generate the appropriate control actions so that the system's trajectories stick to the sliding surfaces. Adaptive control laws are developed to determine the fuzzy rule base and the uncertainties. With respect to SMC, the proposed algorithm eliminates the usual assumptions of SMC and faster convergence can be achieved. Simulation tests are reported and discussed.

The paper is organized as follows. In Section 2, the equivalent control method is used to derive a SMC for rigid robots. Section 3 introduces the proposed HAFC which is a model free approach. Simulation results which include comparison between HAFC and SMC are presented in Section 4. Section 5 offers our concluding remarks.

2. Sliding Mode Control (SMC) design

In this section, the well-developed literature is used to demonstrate the main features and assumptions needed to synthesis a SMC for robotic systems. SMC employs a discontinuous control effort to derive the system trajectories toward a sliding surface, and then switching on that surface. Then, it i.e. will gradually approach the control objective, the origin of the phase plane. To this end, consider a general n-link robot arm, which takes into account the friction forces, unmodeled dynamics, and disturbances, with the equation of motion given by [19].

$$M(x)\ddot{x} + C(x,\dot{x})\dot{x} + G(x) + F_d\dot{x} + F_s(x) + T_d(t) = \tau(t)\,, \quad (1)$$

where:

$x \in R^n$	joint angular position vector of the robot,
$\tau \in R^n$	Applied joint torques (or forces),
$M(x) \in R^{n\times n}$	Inertia matrix, positive definite,
$C(x,\dot{x})\dot{x} \in R^n$	effect of Coriolis and centrifugal forces,
$G(x) \in R^n$	gravitational torques,
$F_d \in R^{n\times n}$	diagonal matrix of viscous and/ or dynamic friction coefficient,
$F_s(x) \in R^n$	vector of unstructured friction effects and static friction terms, and
$T_d \in R^n$	vector of generalized input due to disturbances or unmodeled dynamics.

The controller design problem is as follows. Given the desired trajectories $x_d, \dot{x}_d, \ddot{x}_d$, with some (or all) system parameters being unknown, derive a control law for the torque (or force) input $\tau(t)$ such that the position vector x and the velocity vector $\dot{x}$ can track the desired trajectories, if not exactly then closely. For simplicity, let (1) rewritten as:

$$M(x)\ddot{x} + f(x,\dot{x}) = \tau(t) \ , \tag{2}$$

where the vector $f(x,\dot{x}) = C(x,\dot{x})\dot{x} + G(x) + F_d\dot{x} + F_s(x) + T_d(t)$.

The following assumptions are needed to synthesis a SMC:

Assumption 1: The matrix $M(x)$ is bounded by a known positive definite matrix $\hat{M}(x)$.

Assumption 2: There exists a known estimate $\hat{f}(x,\dot{x})$ for the vector function $f(x,\dot{x})$ in eq. (2).

The tracking control problem is to force the state vector to follow desired state trajectories $x^d(t)$. Let $e(t) = x(t) - x^d(t)$ be the tracking error vector. Further, let us define the linear time-varying surface $s(t)$ [9,10,20].

$$s(t) = \dot{e}(t) - \beta(t)\ , \quad s(t) = [s_1(t), s_2(t), ..., s_n(t)]^T \ , \tag{3}$$

where $\dot{e}(t) = \dot{x}(t) - \dot{x}^d(t)$ and $\beta(t)$ is a time varying linear function. Thus from eqs. (2) and (3), we can get the equivalent control (also called ideal controller [9]) [6,7],

$$\tau_{eq}(t) = f(x,\dot{x}) + M(x)[\ddot{x}^d + \dot{\beta}] \ , \tag{4}$$

where $\tau_{eq}(t)$ is equivalently the average value of $\tau(t)$ which maintains the system's trajectories (i.e. tracking errors) on the sliding surface $s(t) = 0$. To ensure that they attain the sliding surface in a finite time and thereafter maintains the error $e(t)$ on the sliding manifold, generally the control torque $\tau(t)$ consists of a low frequency (average) component $\tau_{eq}(t)$ and a hitting (high frequency) component τ_{ht} as follows:

$$\tau(t) = \tau_{eq}(t) + \tau_{ht}(t) . \tag{5}$$

The role of $\tau_{ht}(t)$ acts to overcome the effects of the uncertainties and bend the entire system trajectories toward the sliding surface until sliding mode occurs. The hitting controller $\tau_{ht}(t)$ is taken as [7]:

$$\tau_{ht} = -M(x)K\,\mathrm{sgn}(s) \ , \tag{6}$$

where, $K = diag(k_1, \cdots, k_n)$, $k_i > 0$, $sgn(s) = [sgn(s_1), sgn(s_2), \cdots, sgn(s_n)]^T$.

To verify the control stability, let us first get an expression for $\dot{s}(t)$. Using eqs. (3-5), the first derivative of eq. (3) is:

$$\begin{aligned} \dot{s}(x,t) &= \ddot{e}(t) - \dot{\beta}(t) \\ &= \ddot{x}(t) - \ddot{x}^d(t) - \dot{\beta}(t) \\ &= M^{-1}(x)[\tau - f(x,t)] - \ddot{x}^d - \dot{\beta}(t) \\ &= M^{-1}(x)\tau_{ht} \\ &= -K\,sgn(s) . \end{aligned} \tag{7}$$

Choosing a Lyapunov function,

$$V_1 = \sum_{i=1}^{n} \frac{1}{2} s_i^2(t) \ , \tag{8}$$

and differentiating using eqs. (6) and (7), we obtain:

$$\begin{aligned} \dot{V}_1 &= \sum_{i=1}^{n} s_i(t)\dot{s}_i(t) = -\sum_{i=1}^{n} k_i s_i(t)\,\mathrm{sgn}(s_i) \\ &= -\sum_{i=1}^{n} k_i |s_i(t)| \le 0 \ , \end{aligned} \tag{9}$$

which provides an asymptotically stable system.

Since the parameters of eq. (2) depend on the manipulator structure and payload it carries, it is difficult to obtain completely accurate values for these parameters. In SMC theory, estimated values are usually used in the control context instead of the exact parameters. So that eq. (4) can be written as:

$$\tau_{eq}(t) = \hat{f}(x,\dot{x}) + \hat{M}(x)[\ddot{x}_d + \dot{\beta}] \ , \tag{10}$$

where $\hat{M}(x)$, $\hat{f}(x,\dot{x})$ are bounded estimates for $M(x)$, and $f(x,\dot{x})$ respectively. As mentioned earlier in assumption 1 and 2, they are assumed to be known in advance.

In sliding mode, the system trajectories are governed by [20]:

$$s_i(t)=0,\ \dot{s}_i(t)=0\ ,\quad i=1,\cdots,n\ . \tag{11}$$

So that, the error dynamics are determined by the function $\beta(t)$. If coefficients of $\beta(t)$ were chosen to correspond to the coefficients of a Hurwitz polynomial, it is thus implying that $\lim_{t\to\infty} e(t)=0$. This suggests $\beta(t)$ taking the following form:

$$\beta_i=-c_{1_i}e_i(t)-c_{2_i}\int e_i\mathrm{dt}\ ,\ \text{ with } c_{1_i},c_{2_i}>0\ . \tag{12}$$

So that, in a sliding manifold, the error dynamics is:

$$\ddot{e}_i(t)+c_{1_i}\dot{e}_i(t)+c_{2_i}e_i(t)=0\ , \tag{13}$$

and the desired performance is governed by the coefficients c_1 and c_2.

In summary, the sliding mode control in eqs. (5, 6) and (10) can guarantee the stability in the Lyapunov sense even under parameter variations. As a result, the system trajectories are confining to the time varying surfaces eq. (3). With this in hand, the error dynamics is decoupled i.e. each degree of freedom is dependent on its perspective error function, eq. (13). The control law eq. (10) however, shows that the coupling effects have not been eliminated since the control signal for each degree of freedom is dependent on the dynamics of the other degrees of freedom. Independency is usually preferred in practice. Furthermore, to satisfy the existence condition, a large uncertainty bound should be chosen in advance. In this case, the controller results in large implementation cost and leads to chattering efforts.

3. Decoupled robot tracking control design

In this section, we propose a fuzzy system that would approximate the equivalent control eq. (4). The main challenge facing the application of fuzzy logic is the development of fuzzy rules. To overcome this problem, an adaptive control law is developed for the on-line generation of the fuzzy rules. The input of the fuzzy system is the sliding surfaces eq. (3), and the output is a fuzzy controller, which substitutes for the equivalent eq. (4). With this choice, no bounds are needed about the system functions. Furthermore, the uncertainties are estimated and continuously compensated for, which means that the hitting controller u_{ht} eq. (6) is adaptively determined on-line.

The coming Subsection gives a brief introduction to fuzzy logic systems and characterizes them with the type, which is utilized in this contribution.

3.1. Fuzzy logic systems

A fuzzy logic system consists of a collection of L fuzzy IF-THEN rules. A one-input one-output fuzzy system has the following form:

$$\text{Rule}\,1:\text{IF } s \text{ is } A_l \text{ THEN } \tau_f \text{ is } \theta^l\ . \tag{14}$$

where $l=1,2,...,L$ is the rule number, s and τ_f are respectively, the input and output variables. A_l is the antecedent linguistic term in rule l; and θ^l, $l=1...L$ is the label of the rule conclusion, a real number called *fuzzy singleton*. The conclusion of each rule (control action), a numerical value not a fuzzy set, can be considered as *pre-defuzzified* output. Defuzzification maps output fuzzy sets defined over an output universe of discourse to a crisp output, τ_f. In this work, we have adopted singleton fuzzifier, product inference, the center-average defuzzifier which reduces the fuzzy rules eq. (14) into the following fuzzy logic system:

$$\tau_f(s,\theta)=\frac{\sum_{l=1}^{L}\theta^l\times\mu_{A_l}(s)}{\sum_{l=1}^{L}\mu_{A_l}(s)}\ , \tag{15}$$

where μ_{A_l} is the membership grade of the input s into the fuzzy set A_l. In eq. (15), if θ^l's are free (adjustable) parameters, then it can be rewritten as:

$$\tau_f(\theta, s) = \vartheta^T \xi(s) , \quad (16)$$

where $\vartheta = (\theta^1,...,\theta^L)$ is the parameter vector and $\xi(s) = [\xi^1(s),...,\xi^L(s)]^T$ is a regression vector given by,

$$\xi^l(s) = \frac{\mu_{A^l}(s)}{\sum_{l=1}^{L} \mu_{A^l}(s)} . \quad (17)$$

Generally, there are two main reasons for using the fuzzy systems in eq. (16) as building blocks for adaptive fuzzy controllers. Firstly, it has been proved that they are universal approximators [21]. Secondly, all the parameters in $\xi(s)$ can be fixed at the beginning of adaptive fuzzy systems expansion design procedure so that the only free design parameter vector is ϑ. In this case, $\tau(\theta, s)$ is linear in parameters. This approach is adopted in synthesizing the adaptive control law in this paper.

Without loss of generality, Gaussian membership functions have been selected for the input variables. A Gaussian membership function is specified by two parameters $\{c, \sigma\}$:

$$\mu_{A_j^l}(x_j) = gaussian(x_j; c, \sigma) = exp\left[-\frac{1}{2}\left(\frac{x_j - c}{\sigma}\right)^2\right],$$

where c represents the membership function's center and σ determines its width.

The fuzzy system used in this contribution is one input one output system, eq. (14). The input of the fuzzy system is normalized using L number of equally spaced Gaussian membership functions inside the universe of discourse. Slopes are identical, see fig. 1.

The described fuzzy system is used to approximate the nonlinear dynamics of robotic systems. In a decoupled manner, the control action is computed for each degree of freedom, based on the corresponding sliding surface. The control actions θ^l (output singletons) which are contained in the parameter vector θ should be known. In the coming Subsection, adaptive laws are derived to do this task. The antecedent part is fixed with Gaussian membership functions.

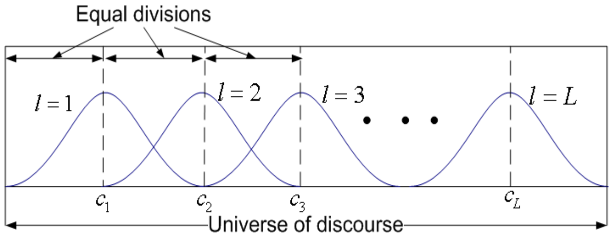


Fig. 1. Input fuzzy sets.

3.2. The adaptation mechanism

Fuzzy systems are universal function approximators. They can approximate any nonlinear function within a predefined accuracy if enough rules are used. This implies the necessity of using expert knowledge in the form of large number of rules and suitable membership functions. Usually trial and error procedure is needed to achieve the requested accuracy. Assigning parameters of the fuzzy systems (or some of them) adaptively greatly facilitates the design (e.g. reduce the number of rules) and enhances the performance (saves the computation resources).

In this Subsection, we derive an adaptive control law to determine the consequent part (control actions contained in parameter vector θ) of the fuzzy system which is used to approximate the unknown nonlinear dynamics of robotic systems. The proposed scheme saves the need to expert knowledge and tedious work needed to assign parameters of the fuzzy system. Furthermore, disturbances, approximation errors and uncertainties are determined on-line leading to a stable closed loop system. Lyapunov stability analysis is the most popular approach to prove and evaluate the convergence property of nonlinear controllers, e.g., sliding mode control, fuzzy control system. Here, Lyapunov analysis is employed to investigate the stability property of the proposed control system.

To this end, an expression for $\dot{s}(t)$ can be expressed as follows:

$$\begin{aligned}\dot{s}(x,t) &= \ddot{e}(t) - \dot{\beta}(t)\\ &= \ddot{x}\ (t) - \ddot{x}^{d}(t) - \dot{\beta}(t)\\ &= M^{-1}(x)\tau - M^{-1}(x)f(x,\dot{x}) - \ddot{x}^{d} - \dot{\beta}(t)\\ &= M^{-1}(x)[\tau_{eq} + \tau_{ht} - f(x,\dot{x})] - \ddot{x}^{d} - \dot{\beta}(t).\end{aligned} \tag{18}$$

By the universal approximation theorem [21], there exists a fuzzy controller $\tau_f(s,\theta)$ in the form of (14) such that:

$$\tau_{eq_i}(t) = \tau_{f_i}(s_i,\theta_i) = \theta_i^T \xi_i\,, \qquad i=1,\cdots,n\,. \tag{19}$$

Employing a fuzzy controller $\hat{\tau}_{f_i}(s_i,\hat{\theta}_i)$ to approximate $\tau_{eq_i}(t)$ as [22, 23],

$$\hat{\tau}_{f_i}(s_i,\hat{\theta}_i) = \hat{\theta}_i^T \xi_i + \varepsilon_i\,, \tag{20}$$

where $\hat{\theta}_i$ is the estimated value of the ideal parameter vector θ_i and ε_i is the approximation error. Now, the SMC in eq. (5) can be rewritten as:

$$\tau_i(t) = \hat{\tau}_{f_i}(s_i,\hat{\theta}_i) + \tau_{ht_i}(s_i) + \varepsilon_i\,, \tag{21}$$

where the fuzzy controller $\hat{\tau}_{f_i}(s_i,\hat{\theta}_i)$ is designed to approximate the equivalent controller $\tau_{eq_i}(t)$. Define $\tilde{\tau}_{f_i} = \tau_{eq_i} - \hat{\tau}_{f_i}(s_i,\hat{\theta}_i)$, $\tilde{\theta}_i = \theta_i - \hat{\theta}_i$, and use eq. (17), then it is obtained that:

$$\tilde{\tau}_{f_i} = \tilde{\theta}_i^T \xi_i + \varepsilon_i\,. \tag{22}$$

Substituting from eqs. (5) and (20-22) into eq. (18) yields:

$$\dot{s} = M^{-1}(x)(\tau_{ht} + \Theta + \varepsilon)\,, \tag{23}$$

where $\Theta^T = [\tilde{\theta}_1^T \xi_1, \tilde{\theta}_1^T \xi_2, \cdots, \theta_n^T \xi_n]$. Now, $M^{-1}(x)$ may be approximated by a constant positive definite diagonal matrix $\overline{M}$. Unlike constant control gain schemes (see [24] for example), this assumption has been taken into account as follows. Eq. (23) can be rewritten as:

$$\dot{s}_i = \overline{M}_{i,i}(\tau_{ht_i} + \tilde{\theta}_i^T \xi_i + E_i)\,, \qquad i=1,\cdots,n\,, \tag{24}$$

where E_i is the sum of approximation error ε_i and the uncertainty; $|\varepsilon_i| \le E_i$. A control goal would be the on-line determination of its estimate; $\hat{E}_i(t)$. The estimation error is defined by:

$$\tilde{E}_i(t) = E_i - \hat{E}_i(t)\,, \qquad i=1,\cdots,n\,. \tag{25}$$

Define a Lyapunov function as:

$$V_2\big(s(t),\tilde{\theta},\tilde{E}\big) = \sum_{i=1}^{n}\left[\frac{1}{2}s_i^2 + \overline{M}_{i,i}\frac{\tilde{\theta}_i^T\tilde{\theta}_i}{2\eta_{1i}} + \overline{M}_{i,i}\frac{\tilde{E}_i^2}{2\eta_{2i}}\right], \tag{26}$$

where η_1 and η_2 are positive constants. Differentiating eq. (26) with respect to time and using eq. (23), it is obtained that:

$$\begin{aligned}&\dot{V}_2\big(s(t),\tilde{\theta},\tilde{E}\big)\\ &= \sum_{i=1}^{n}\left[s_i\dot{s}_i + \overline{M}_{i,i}\frac{\tilde{\theta}_i^T\dot{\tilde{\theta}}_i}{\eta_{1i}} + \overline{M}_{i,i}\frac{\tilde{E}_i\dot{\tilde{E}}_i}{\eta_{2i}}\right]\\ &= \sum_{i=1}^{n}\left[s_i\overline{M}_{i,i}(\tau_{ht_i} + \tilde{\theta}_i^T\xi_i + E_i) + \overline{M}_{i,i}\frac{\tilde{\theta}_i^T\dot{\tilde{\theta}}_i}{\eta_{1i}} + \overline{M}_{i,i}\frac{\tilde{E}_i\dot{\tilde{E}}_i}{\eta_{2i}}\right]\\ &= \sum_{i=1}^{n}\left[\overline{M}_{i,i}[\tilde{\theta}_i^T(s_i\xi_i + \frac{\dot{\tilde{\theta}}_i}{\eta_{1i}})] + \overline{M}_{i,i}s_i(\tau_{ht_i} + E_i) + \overline{M}_{i,i}\frac{\tilde{E}_i\dot{\tilde{E}}_i}{\eta_{2i}}\right].\end{aligned}$$

Substituting for E_i from eq. (25),

$$\begin{aligned}&= \sum_{i=1}^{n}\left[\overline{M}_{i,i}[\tilde{\theta}_i^T(s_i\xi_i + \frac{\dot{\tilde{\theta}}_i}{\eta_{1i}})]\right.\\ &\qquad\left. + \overline{M}_{i,i}s_i(\tau_{ht_i} + \hat{E}_i) + \overline{M}_{i,i}\tilde{E}_i(s_i + \frac{\dot{\tilde{E}}_i}{\eta_{2i}})\right].\end{aligned}$$

To satisfy $\dot{V}_2 \leq 0$, the adaptive laws can be selected as:

$$\dot{\tilde{\theta}}_i = -\eta_{1i} s_i \xi_i \,, \tag{27}$$

$$\tau_{ht_i} = -\hat{E}_i \,\mathrm{sgn}(s_i)\,, \tag{28}$$

and using (25) (recalling E_i is constant)

$$\dot{\hat{E}}_i(t) = -\dot{\tilde{E}}_i = \eta_{2i}|s_i|\,. \tag{29}$$

Then, the first time derivative of (26) can be written as:

$$\begin{aligned}
&\dot{V}_2\left(s(t),\tilde{\theta},\tilde{E}\right) \\
&= \sum_{i=1}^{n}\left[\overline{M}_{i,i}s_i[-\hat{E}_i\, sgn(s_i) + \hat{E}_i] + \overline{M}_{i,i}\tilde{E}_i(s_i - |s_i|)\right] \\
&= \sum_{i+1}^{n}\left[\overline{M}_{i,i}[s_i(\hat{E}_i + \tilde{E}_i) - s_i\hat{E}_i\, sgn(s) - \tilde{E}_i|s_i|]\right] \\
&= \sum_{i+1}^{n}\overline{M}_{i,i}[s_iE_i - |s_i|(\hat{E}_i + \tilde{E}_i)] \\
&= \sum_{i+1}^{n}\overline{M}_{i,i}E_i(s_i - |s_i|) \leq 0.
\end{aligned} \tag{30}$$

Therefore, V_2 is reduced gradually and the control system is stable which means that the system trajectories converge to the sliding surfaces $s(t)$ while $\hat{\theta}$ and $\hat{E}$ remain bounded. Now, the control law eq. (18) can be rewritten as follows:

$$\tau_i(t) = \hat{\tau}_{f_i}(s_i,\hat{\theta}_i) - \hat{E}_i\,\mathrm{sgn}(s_i)\,, \qquad i = 1,\cdots,n\,. \tag{31}$$

In summary, the adaptive fuzzy sliding mode controller eq. (31) has two terms; $\hat{\tau}_{f\,i}(s,\hat{\theta})$ given in eq. (17) with the parameter $\hat{\theta}_i$ adjusted by eq. (27) and the uncertainties and approximation bound $\hat{E}_i$ adjusted by eq. (29). By applying these adaptive laws, the HAFC is model free and can be guaranteed to be stable for any nonlinear system has the form of eq. (2).

It should be noted that implementing the implies that both error algorithm and control signals have been decoupled, since each of them is dependent only on the perspective sliding surface. Unlike SMC, the proposed HAFC does not require any knowledge about the system functions nor their bounds. It adaptively determines and compensates for the unknown dynamics and external disturbances leading to a stable closed loop system. Fig. 2 shows the main elements of the control system.

5. Simulation results

To simplify the presentation, a two-link robot arm with varying loads is used to generate data in the simulation tests; fig. 3. The arm is depicted as 2-input, 2-ouput nonlinear system. The control architecture shown in fig. 2 represents the closed loop system, in which the robot is the plant to be controlled. The control input for each joint is

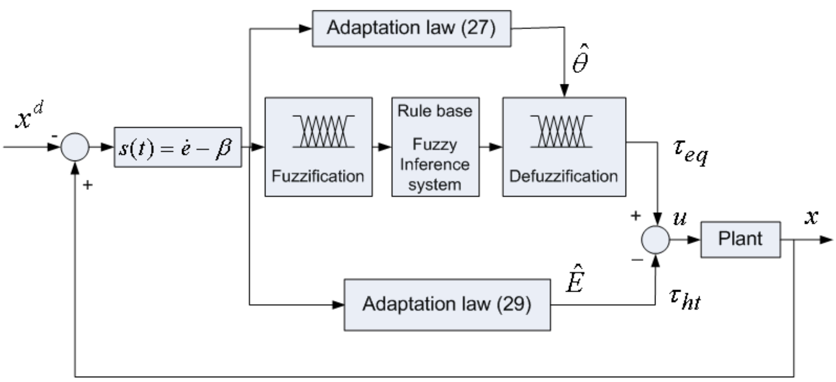


Fig. 2. The closed loop control system utilizing HAFC.

dependent on its sliding surface, parameter vector of the fuzzy controller and the estimated error; eqs. (18), (27-29) and (31).

The equations of the robot motion are given in Appendix A. The state variable vector is considered as the joint positions; i.e. $x = [x_1, x_2]^T$. They are usually available feedback signals through encoders mounted on the motor shafts.

Link parameters are given in table 1, where the mass of link one m_1 and link two m2 are randomly varied; rand (1) is a pseudo-random number ranges from $0.0 \rightarrow 1.0$. Fig. 4-a shows their time history. A random disturbance torque has been added to link two, such that $T_d = [0,7\times \text{rand}\ (1)]^T$; fig. 4-b. Dynamic and static friction torques are selected as follows:

$$F_d = \begin{bmatrix} 5\cos(\dot{x}_1) & 0 \\ 0 & 3\cos(\dot{x}_2) \end{bmatrix},\ F_s = \begin{bmatrix} 1.8\,sgn(\dot{x}_1) \\ 1.2\,sgn(\dot{x}_2) \end{bmatrix}.$$

The friction and disturbance torques are unknown to the algorithm. Random signals are generated by the *rand* function in Matlab. The desired trajectories for x_1 and x_2 are:

$x_1^d(t) = -A_1 \sin(\omega_1 t)$, $x_2^d(t) = -A_2 \sin(\omega_2 t)$, with $A_1 = 0.6$ rad, $A_2 = 0.8$ rad, $\omega_1 = 0.5\pi$ rad s^{-1}, $\omega_2 = 1.0\pi$ rad, s^{-1}.

Initially, the arm is assumed at rest, i.e. $\dot{x}_{t=0} = [0,0]^T$ rad/s, and position of links as $x_{t=0} = [-0.75\pi, 0.75\pi]^T$ rad, which resulted in initial position error $e_{(t=0)} = [-0.75\pi, 0.75\pi]^T$ rad and velocity error $\dot{e}_{(t=0)} = [0.32,\ \ -1.3635]^T$ rad, /s.

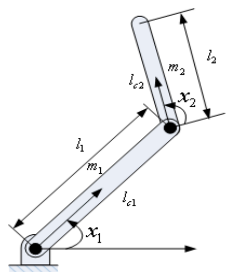


Fig. 3. A two link rigid robot.

Table 1
Parameters of the robot arms

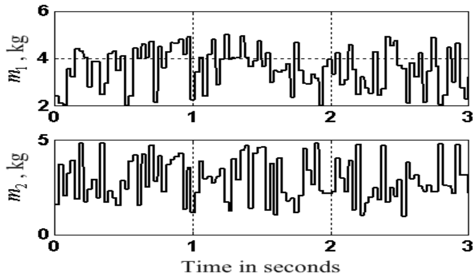
Inertia parameter	Link 1	Link 2
m mass (kg)	$2 + 3 \times rand(1)$	$1 + 4 \times rand(1)$
l length (m)	1.0	0.7
l_c position of c.g. (m)	0.5	0.35

The HAFC has been simulated under the following settings. Two rules were implemented to determine each of the two equivalent control components (i.e. $L = 2$). Each rule base has one input, s_i and one output, τ_{eq_i}, where the subscript $i = 1, 2$ denotes the joint number. This means that 4 rules are used to determine the arm's nonlinear dynamics. The universe of discourse of the input s_i is [-50, 50]. Constants of the sliding surfaces $c_1 = [40,40]^T$ and $c_2 = [0.3, 0.3]^T$. The slopes are set to $\sigma = 50$ for all membership functions. The learning rates are selected as $\eta_1^T = [15, 1.5]$ and $\eta_2^T = [45, 6]$. To reduce the chattering phenomenon, the sign function in eq. (28) is replaced by the saturation function.

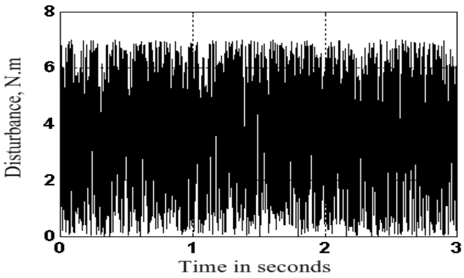
Evolution of the parameter vectors is given in fig. 5-a. Zeros were used to initiate their components. The superscripts denote the rule number, 1 and 2. Rates of adaptation for the parameter vectors are depicted in fig. 5-b. Also, the estimated errors and uncertainties were initiated from zeros and their magnitudes were adaptively tuned on-line, fig. 6. They reach the steady state after relatively short time (less than half second).

For the sake of comparison, the SMC in eqs. (5), (6) and (10) has been simulated under the following settings. Gain K of the hitting controller gain in eq. (6) was set as $K = 400I$ where I is 2×2 identity matrix. This value of K has been selected as the maximum possible one, which means maximum possible rate of convergence. Larger value results in chattering. $\hat{f}(x, \dot{x})$ and $\hat{M}(x)$ in eq. (10) were selected as follows: $\hat{M} = 5I$ i.e. time-independent (constant) matrix and

$$\hat{f} = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}\dot{x} + \begin{bmatrix} 2 \\ 1 \end{bmatrix} + \begin{bmatrix} 0.7 & 0 \\ 0.7 & 0 \end{bmatrix}\dot{x} + \begin{bmatrix} 3 \\ 2 \end{bmatrix}.$$

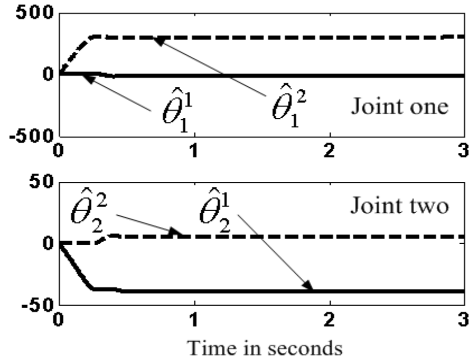


(a)

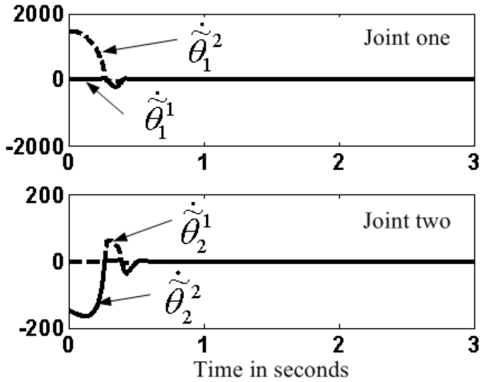


(b)

Fig. 4. Mass of links (a) and disturbance (b) profiles.



(a)



(b)

Fig. 5. Time history of (a) parameter vectors (i.e. control actions) and (b) adaptation rate.

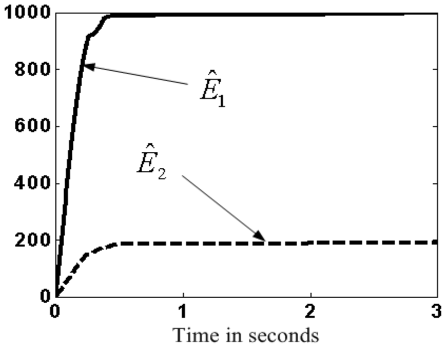


Fig. 6. Time history of the estimated errors.

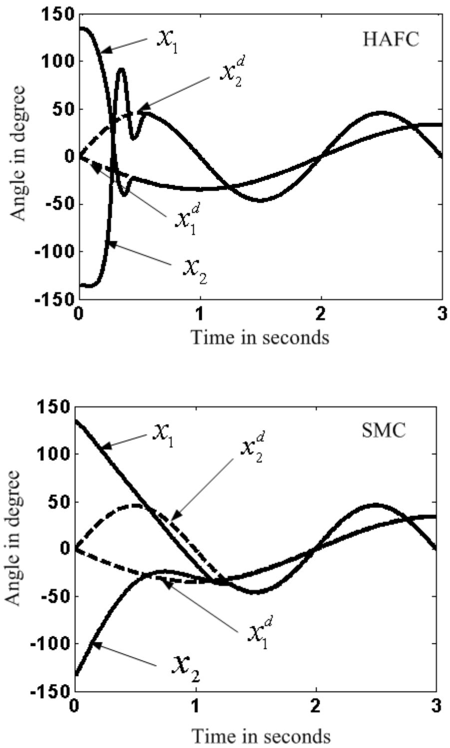


Fig. 7. The desired joint angles, x^d and actual angles x .

Similar to HAFC, the friction and disturbance torques were unknown to the algorithm. Results are shown in figs. 7 to 12. Convergence has been achieved by the two controllers. However the HAFC was faster than SMC.

It can be concluded that all signals of the control system are bounded, the states have converged to the equilibrium points and the control targets have been met.

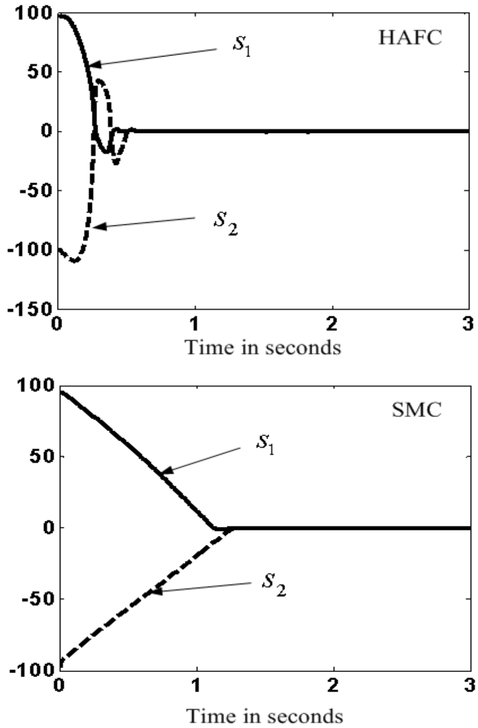


Fig. 8. Time history of the sliding surfaces.

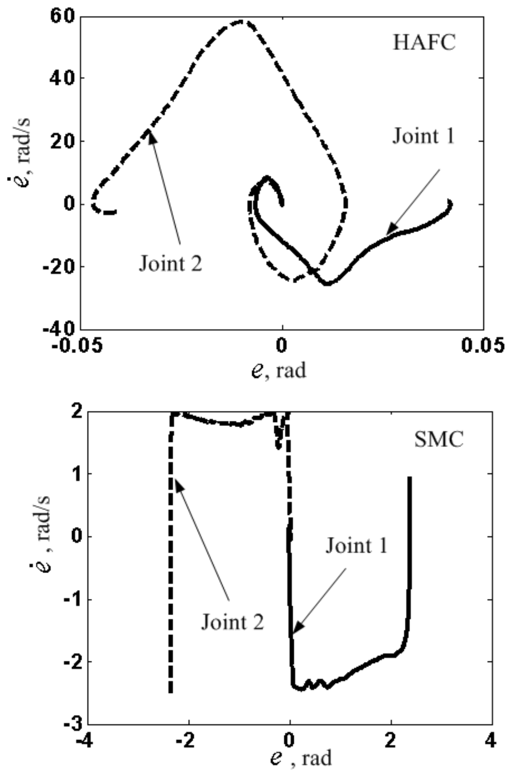


Fig. 9. Phase plots.

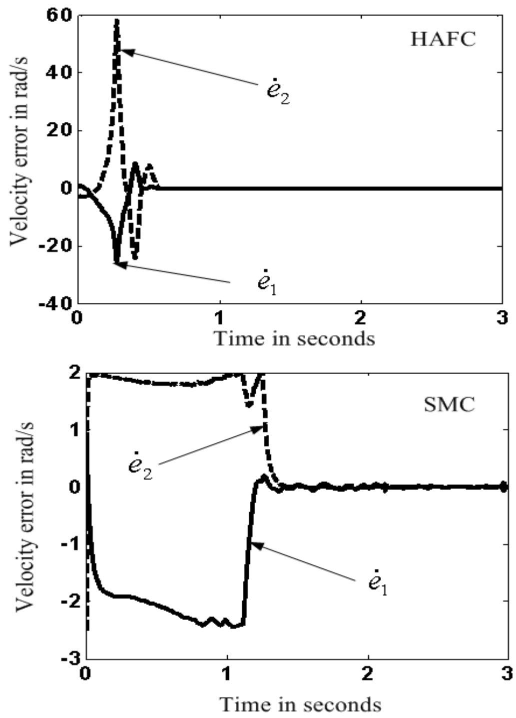


Fig. 10. Velocity tracking errors.

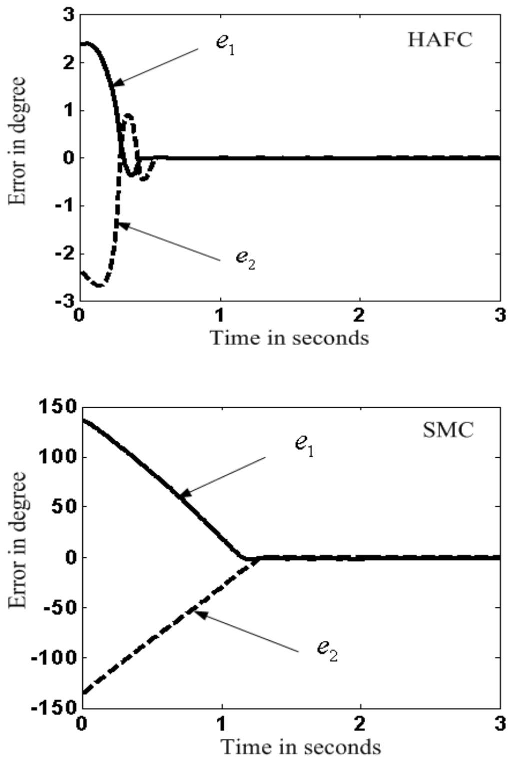


Fig. 11. Trajectory tracking errors.

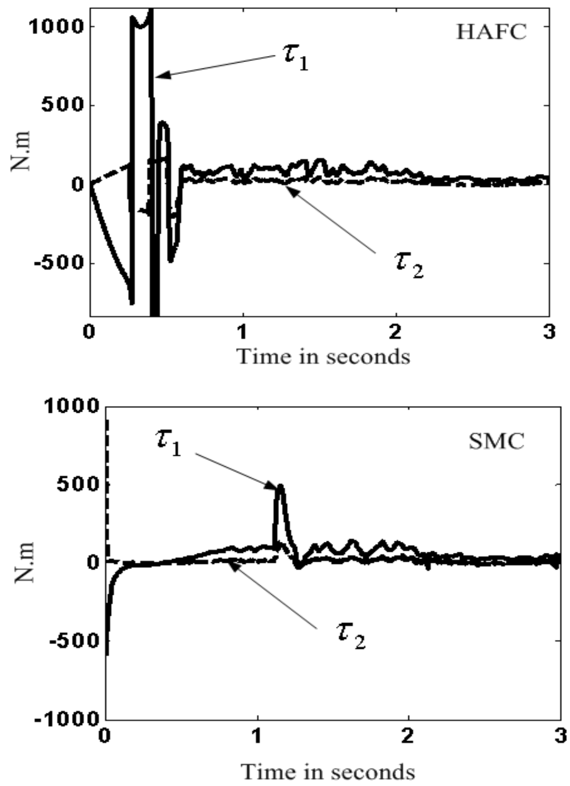


Fig. 12. The inputs.

5. Conclusions

In this article, the universal approximation property of fuzzy systems and powerfulness of SMC theory have been utilized to compose a new hybrid control algorithm for robotic systems. Optimal parameters of the fuzzy system and uncertainty bound are generated on-line. The proposed control scheme has the following advantages: (a) do not require the system model; (b) guarantees the stability of the closed loop system; (c) simple rule base (one-input one-output fuzzy system). The adaptive control law generates on-line the fuzzy rules. Furthermore, the uncertainties are learned on-line and adaptively compensated for. In comparison with SMC, the proposed scheme has eliminated the assumptions, which are usually needed to synthesize a SMC. Simulation tests have been carried out on a two link planar robot. The fuzzy system needs only two rules per joint to determine the control signal.

The approach significantly eliminates the fuzzy data base burden and reduces the computing time, thereby increasing the sampling frequency for possible implementation. It should be emphasized that, the developed adaptive laws learn the fuzzy rules and uncertainties. Zeros have been used to initiate them. Results show the effectiveness of the overall closed-loop system performance.

Appendix A

Assuming rigidity of links and joints, it can be shown that the equation of motion of the robot arms is given by:

$$M_{11}(x_2)\ddot{x}_1 + M_{12}(x_2)\ddot{x}_2 + H_1(x_1, x_2, \dot{x}_1, \dot{x}_2) = \tau_1,$$

$$M_{12}(x_2)\ddot{x}_1 + M_{22}(x_2)\ddot{x}_2 + H_2(x_1, x_2, \dot{x}_1, \dot{x}_2) = \tau_2,$$

where:

$$M_{11}(x_2) = m_1 l_{c1}^2 + I_1 + m_2 l_{c2}^2 + I_2 + m_2 l_1^2 + 2 m_2 l_1 l_{c2} \cos(x_2),$$

$$M_{12}(x_2) = m_2 l_{c2}^2 + I_2 + m_2 l_1 l_{c2} \cos(x_2),$$

$$M_{22} = m_2 l_{c2}^2 + I_2,$$

$$h = m_2 l_1 l_{c2} \sin(x_2),$$

$$H_1(x_1, x_2, \dot{x}_1, \dot{x}_2) = -2h\dot{x}_1\dot{x}_2 + m_2 g l_{c2} \cos(x_1 + x_2) + (m_1 l_{c1} + m_2 l_1) g \cos(x_1),$$

$$H_2(x_1, x_2, \dot{x}_1, \dot{x}_2) = h\dot{x}_1^2 + m_2 g l_{c2} \cos(x_1 + x_2),$$

and $g = 9.81\, m/s^2$ is the acceleration of gravity.

References

[1] A.B. Sharkawy, H. El-Awady, and K.A.F. Moustafa, “Stable Fuzzy Control for a Class of Nonlinear Systems,” Trans. of the Institute of Measurement and Control, Vol. 25 (3), pp. 265-278 (2003).

[2] T. Back, Evolutionary Algorithms in Theory and Practice, Oxford UP (1996).

[3] J. Kim and N. Kasabov, "HyFIS: Adaptive Neuro-Fuzzy Inference Systems and Their Application to Nonlinear Dynamical Systems," Neural Networks, 12, pp. 1301-1319 (1999).
[4] J.L. Buessler and J-P. Urban, "Visually Guided Movements: Learning with Modular Neural Maps in Robotics," Neural Networks, 11, pp. 1395-1415 (1998).
[5] J.J. Slotine and W. Li, Applied Nonlinear Control, Printice-Hall International, Inc. (1991).
[6] V.I. Utkin, Sliding Modes and their Applications in Variable Structure Systems, Moscow, Russia: Mir Publishers (1978).
[7] V.I. Utkin, "Variable Structure Systems with Sliding Modes," IEEE Trans. Automatic Control, Vol. AC-22, pp. 212-222 (1977).
[8] V.I. Utkin, Sliding Modes in Control and Optimization, Springer-Verlag, Berlin (1992).
[9] B. Yao, S. Chan and D. Wang, "Variable Structure Adaptive Motion and Force Control of Robot Manipulator," Automatica, Vol. 30 (9), pp. 1473-1477 (1994).
[10] B. Yao, S.P. Chan, and D. Wang, "Unified Formulation of Variable Structure Control Schemes for Robot Manipulators," IEE Trans. on Automatic Control, Vol. 39 (2), pp. 371-376 (1994).
[11] R.A. DeCarlo, S.H. Zak, and G.P. Matthews, "Variable Structure Control of Nonlinear Multivariable Systems: A Tutorial," Proceedings of the IEEE, Vol. 76 (3), pp. 212-232 (1988).
[12] K. Passino and S. Yurkovich, Fuzzy Control, Addison-Wesley Longman, Inc. (1998).
[13] R. Ordonez et al., "Adaptive Fuzzy Control: Experiments and Comparative Analysis," IEEE Transactions on Fuzzy Systems, Vol. 5 (2), pp. 167-188 (1997).
[14] R. Palm, "Robust Control by Fuzzy Sliding mode," Automatica, Vol. 30 (9), pp. 11-21 (1994).
[15] L.X. Wang, "Stable Adaptive Fuzzy Control of Nonlinear Systems," IEEE Trans. Fuzzy Systems, Vol. 1, pp. 146-155 (1993).
[16] Y.S. Lu, and J.S. Chen, "A Self Organizing Fuzzy Sliding Mode Controller Design for a Class of Nonlinear Servo Systems," IEEE Trans. Ind. Electron., Vol. 41, pp. 492-502 (1994).
[17] T. Chai, and S. Tong, "Fuzzy Direct Adaptive Control for a Class of Nonlinear Systems," Fuzzy Sets and Systems, Vol. 103, pp. 387-397 (1999).
[18] C.S. Ting, T.H.S. Li, and F.C. Kang, "Design of Fuzzy Controller for Active Suspension Systems," Mechatronica, Vol. 5 (4), pp. 365-383 (1995).
[19] Y.C. Hsu, et al., "A Fuzzy Adaptive Variable Structure Controller with Application to Robot Manipulators", IEEE Transaction on Systems, Man and Cypernetics-Part B: Cybernetics, Vol. 31 (3) (2001).
[20] Y. Stephaneko and C.Y. Su, "Variable Structure Control of Robot Manipulators with Nonlinear Sliding Manifolds," Int. J. Control, Vol. 58 (2), pp. 285-300 (1993).
[21] L.X. Wang, Adaptive Fuzzy Systems and Control, PTR Prentice Hall (1994).
[22] Y. Tang, N. Zhang, and Y. Li, "Stable Fuzzy Adaptive Control for a Class of Nonlinear Systems," Fuzzy Sets and Systems 104, pp. 279-288 (1999).
[23] Y.T Kim, and Z. Bien, "Robust Self Learning Fuzzy Controller Design for a Class of Nonlinear MIMO Systems," Fuzzy Sets and Systems 111, pp. 117-135 (2000).
[24] S.J. Huang and W.C. Lin, "Adaptive Fuzzy Controller with Sliding Surface for Vehicle Suspension Control," IEEE Trans. on Fuzzy Systems, Vol. 11 (4), pp. 550-559 (2003).

Received December 16, 2004
Accepted February 28, 2005