Scientific
Research

An Adaptive Fuzzy Sliding Mode Control Scheme for Robotic Systems

Abdel Badie Sharkawy[*]**, Shaaban Ali Salman**
Mechanical Engineering Department, Faculty of Engineering, Assiut University, Assiut, Egypt
E-mail: [*]*ab.shark@aun.edu.eg*

Abstract

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

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 [1,2], genetic algorithms [3], neuro-fuzzy [4,5] and neural network [6] 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 [7].

The study of output tracking problems has a longstanding history. Sliding mode control (SMC) is often favored basic control approach, because of the insensitivity to parametric uncertainties and external disturbances [7-10]. 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 switch-

ing surfaces to obtain the desired performance of the system is very important and has been the topic of many previous works [11,12]. With the desired switching surface, we need to design a SMC such that any state outside 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 able to solve complex nonlinear control problems. They provide an effective means to capture the approximate nature of real world. Examples are numerous; see [13] for instance. While non-adaptive fuzzy control has proven its value in some applications [1,2,14], 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. It means the use of on-line data to continually "learn" the fuzzy controller, which will ensure that the performance objectives are met. This concept has proved to be a promising approach for solving complex nonlinear control problems [15,16].

Recently, adaptive fuzzy sliding mode control design has

drawn much attention of many researchers. Because, control chattering, an inherent problem associated with SMC, can evoke un-modeled and undesired high frequency dynamics, Ho *et al.* [17] have proposed an adaptive fuzzy sliding mode control with chattering elimination for nonlinear SISO systems. The adaptive laws, however, rely on the projection algorithms, which can hardly be satisfied in practical problems. In [18], the authors have established an adaptive sliding controller design based on T-S fuzzy system models. The fuzzy system used is rather complicated and the upper bound of the uncertainty is needed to synthesize the controller. A robust fuzzy tracking controller for robotic manipulator which uses sliding surfaces in the control context can be found in [19]. The control scheme, however, depends heavily on the properties of the dynamic model of robotic manipulators and similar to [17], the authors use the projection algorithms which have practical limitations.

More recently, Li and Huang [20] have designed a MIMO adaptive fuzzy terminal sliding mode controller for robotic manipulators. In the first phase of their work, the fuzzy control part relied on some expert knowledge and a trial-and-error procedure is needed to determine the output singletons. In the second phase, they designed an adaptive control scheme that determines these parameters on-line. The rule base, however is restricted to five rules per each joint and the fuzzy singletons should have values within specified ranges to enforce stability.

In this work, an adaptive fuzzy sliding mode control (AFSMC) scheme 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 needed to synthesize the SMC and better performance can be achieved.

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 AFSMC which is a model free approach. Simulation results which include comparison between AFSMC 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 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

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

(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;
$T_d \in R^n$ 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:

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

(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 (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 $s(t)$ [21],

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

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

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

(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 there-

after 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

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

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 [8,21]

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

where, $K = \mathrm{diag}(k_1, \cdots, k_n)$, $k_i > 0$, and $\mathrm{sgn}(s) = \left[\mathrm{sgn}(s_1), \mathrm{sgn}(s_2), \cdots, \mathrm{sgn}(s_n)\right]T$.

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

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

Choosing a Lyapunov function

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

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

$$\begin{aligned}
\dot{V}_1 &= \sum_{i=1}^{n} s_i(t)\dot{s}_i(t) = s_i(t)\tau_{hti} \\
&\le -\sum_{i=1}^{n} k_i s_i^2 \le 0.
\end{aligned} \qquad (9)$$

which provides an exponentially stable system.

Since the parameters of (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 (4) can be written as:

$$\tau_{eq}(t) = \hat{f}(x,\dot{x}) + \hat{M}^{-1}(x)[\ddot{x}^d + \dot{\beta}] \qquad (10)$$

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 [9]:

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

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:

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

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

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

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

In summary, the sliding mode control in (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 (3). With this in hand, the error dynamics is decoupled *i.e.* each degree of freedom is dependent on its perspective error function, (13). The control law (10) however, shows that the coupling effects have not 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 (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 (3), and the output is a fuzzy controller, which substitutes for the equivalent (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} (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:

$$\text{Rule } l : \text{IF } s \text{ is } A_l \text{THEN } \tau_f \text{ is } \theta^l \qquad (14)$$

where $l = 1, 2, \cdots, L$ 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, $l = 1, \cdots, L$ is the label of the rule conclusion, a real number called *fuzzy singleton*. The conclusion of each rule (control ac-

tion), 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 (14) into the following fuzzy logic system:

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

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

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

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

$$\xi^l(s) = \frac{\mu_{A_l}(s)}{\sum_{l=1}^{L} \mu_{A_l}(s)} \qquad (17)$$

Generally, there are two main reasons for using the fuzzy systems in (16) as building blocks for adaptive fuzzy controllers. Firstly, it has been proved that they are universal approximators [22]. 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 $\{c,\sigma\}$:

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

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, (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 **Figure 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
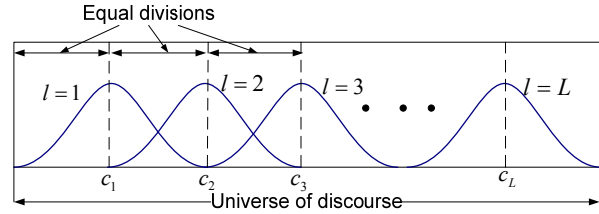

Figure 1. Input fuzzy sets.

coming Subsection, adaptive laws are derived to do this task. The antecedent part is fixed with Gaussian membership functions.

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 compensate for 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. By the universal approximation theorem [22], there exists a fuzzy controller $\tau_f(s,\theta)$ in the form of (16) such that

$$\tau_{eqi}(t) = \tau_{fi}(s_i, \theta_i) + \varepsilon_i = \theta_i^T \xi + \varepsilon_i, \quad i = 1, \cdots, n \qquad (18)$$

where ε_i is the approximation error and is bounded by $|\varepsilon_i| \leq E_i$. Employing a fuzzy controller $\hat{\tau}_f(s_i, \hat{\theta}_i)$ to approximate $\tau_{eq_i}(t)$ as

$$\hat{\tau}_f(s_i, \hat{\theta}_i) = \hat{\theta}_i^T \xi_i \qquad (19)$$

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

$$\tau_i(t) = \hat{\tau}_{f\,i}(s_i, \hat{\theta}_i) + \tau_{hti}(s_i) \tag{20}$$

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

$$\tilde{\tau}_{f} = \tilde{\theta}_i^T \xi_i + \varepsilon_i \tag{21}$$

An expression for $\dot{s}(t)$ can be expressed as follows:

$$
\begin{aligned}
\dot{s}(x,t) &= \ddot{e}(t) - \dot{\beta}(t) \\
&= \ddot{x}(t) - \ddot{x}^d(t) - \dot{\beta}(t) \\
&= M^{-1}(x)\tau - M^{-1}(x)f(x,\dot{x}) - \ddot{x}^d - \dot{\beta}(t) \\
&= M^{-1}(x)(\tau_{eq} + \tau_{ht}) - M^{-1}(x)f(x,\dot{x}) - \ddot{x}^d - \dot{\beta}(t)
\end{aligned}
\tag{22}
$$

Substituting from (19-21):

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

where $\Theta^T = \left[\tilde{\theta}_1^T \xi_1, \tilde{\theta}_2^T \xi_2, \cdots, \tilde{\theta}_n^T \xi_n \right]$. Now, assume that $M^{-1}(x)$ can be approximated by known constant positive definite diagonal matrix $\bar{M}$. Unlike constant control gain schemes (see [23,24] for example), this assumption has been taken into account as follows. Equation (23) can be rewritten as

$$\dot{s}_i = \bar{M}_{i,i}(u_{hti} - \tilde{\theta}_i^T \xi_i - E_i), \qquad i = 1, \cdots, n \tag{24}$$

where E_i is the sum of approximation errors and uncertainties. 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

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

where η_1 and η_2 are positive constants. Differentiating (25) with respect to time and using (23), it is obtained that

$$
\begin{aligned}
\dot{V}_{2i}\left(s_i, \tilde{\theta}_i, \tilde{E}_i\right) &= \sum_{i=1}^{n} \left[s_i\dot{s}_i + \bar{M}_{i,i}\frac{\tilde{\theta}_i^T \dot{\tilde{\theta}}_i}{\eta_{1i}} + \bar{M}_{i,i}\frac{\tilde{E}_i \dot{\tilde{E}}_i}{\eta_{2i}} \right] \\
&= \sum_{i=1}^{n} \left[s_i\bar{M}_{i,i}(u_{hti} - \tilde{\theta}_i^T \xi_i - E_i) + \bar{M}_{i,i}\frac{\tilde{\theta}_i^T \dot{\tilde{\theta}}_i}{\eta_{1i}} + \bar{M}_{i,i}\frac{\tilde{E}_i \dot{\tilde{E}}_i}{\eta_{2i}} \right] \\
&= \sum_{i=1}^{n} \left[\bar{M}_{i,i}[\tilde{\theta}_i^T (s_i\xi_i + \frac{\dot{\tilde{\theta}}_j}{\eta_{1i}})] + \bar{M}_{i,i}s_i(u_{hti} - E_i) + \bar{M}_{i,i}\frac{\tilde{E}_i \dot{\tilde{E}}_i}{\eta_{2i}} \right]
\end{aligned}
$$

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

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

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

Using (20)

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

then (22) can be rewritten as

$$
\begin{aligned}
\dot{V}_2\left(s(t), \tilde{\theta}, \tilde{E}\right) &= \sum_{i=1}^{n} \left[\bar{M}_{i,i}s_i[-\hat{E}_i\,\mathrm{sgn}(s_i) - E_i] - |s_i|\tilde{E}_i^2 \right] \\
&\leq -\sum_{i=1}^{m} |s_i|\bar{M}_{i,i}(\hat{E}_i - |E_i|) \leq 0
\end{aligned}
\tag{30}
$$

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, if we let

$$\Gamma(t) \equiv \sum_{i=1}^{m} |s_i|\bar{M}_{i,i}(\hat{E}_i - |E_i|) \leq -\dot{V}_2 \tag{31}$$

and integrate $\Gamma(t)$ with respect to time, then it is shown that

$$\int_0^t \Gamma(\tau)\mathrm{d}\tau \leq V_2(s(0), \tilde{\theta}, \tilde{E}) - V_2(s(t), \tilde{\theta}, \tilde{E}) \tag{32}$$

Because $V_2(s(0), \tilde{\theta}, \tilde{E})$ is bounded and $V_2(s(t), \tilde{\theta}, \tilde{E})$ is non-increasing and bounded, it implies that

$$\lim_{t\to\infty} \int_0^t \Gamma(\tau)\mathrm{d}\tau \leq \infty \tag{33}$$

Furthermore, $\dot{\Gamma}$ is bounded, so that by Barbalat's lemma [7], it can be shown that $\lim_{t\to\infty} \int_0^t \Gamma(\tau)\mathrm{d}\tau = 0$. That is, $s(t) \to 0$ as $t \to 0$. As a result, the proposed AFSMC is asymptotically stable.

Hence, the control law (18) can be rewritten as follows

$$u_i(t) = \hat{u}_{f\,i}(s_i, \hat{\theta}_i) - \hat{E}_i\,\mathrm{sgn}(s_i), \quad i = 1, \cdots, n. \tag{34}$$

In summary, the adaptive fuzzy sliding mode controller (34) has two terms; $\hat{u}_{f\,i}(s, \hat{\theta})$ given in (19) with the parameter $\hat{\theta}_i$ adjusted by (27) and the uncertainties and approximation bound $\hat{E}_i$ adjusted by (29). By applying these adaptive laws, the AFSMC is model free and can be guaranteed to be stable for any nonlinear system has the form of (2).

It should be noted that implementing the algorithm implies that the both error dynamics and control signals has been decoupled, since each of them is dependent only on the perspective sliding surface. Unlike SMC, the proposed AFSMC 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. **Figure 2** shows the main elements of the control system.

Remark 1. Since the control laws (6) and (34) contain the sign function, direct application of such control signals to the robotic system may result in chattering caused
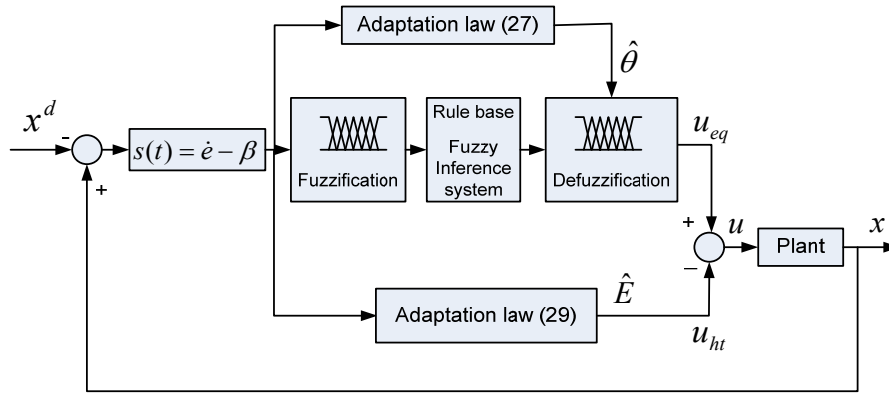
Figure 2. The closed loop control system utilizing AFSMC.

by the signal discontinuity. To overcome this problem, the control law is smoothed out within a thin boundary layer ϕ [7,21] by replacing the sign function by a saturation function defined as:

$$\text{sat}\left(\frac{s_i}{\varphi_i}\right) = \begin{cases} \text{sgn}\left(\dfrac{s_i}{\varphi_i}\right) & \left|\dfrac{s_i}{\varphi_i}\right| > 1 \\[3mm] \dfrac{s_i}{\varphi_i} & \left|\dfrac{s_i}{\varphi_i}\right| \leq 1 \end{cases}$$

4. Simulation Results

In this Section, we simulate the AFSMC and SMC on a two link robot; **Figure 3**. Simulation tests are carried out using MATLAB R2009a, version 7.8 under Windows 7 environment. A two link robot arm with varying loads is used to generate data in the simulation tests. The arm is depicted as 2-input, 2-ouput nonlinear system. The control architecture shown in **Figure 2** represents the closed loop system, in which the robot is the plant to be controlled. The detailed descriptions of the matrices $M(x)$, $C(x,\dot{x})$ and $G(x)$ in (1) for this robot are given in Appendix A. We consider the state variable vector 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 $m_1 = 2 + 2 \times rand(1)$, $m_2 = 1 + 3 \times rand(1)$, $l_1 = 1.0$ m and $l_2 = 0.7$ m, where the mass of link one m_1 and link two m_2 are randomly varied; $rand(1)$ is a pseudo-random number ranges from $0.0 \rightarrow 1.0$. **Figure 4(a)** shows their time history. A random disturbance torque has been added to the gravity torque of link two, such that $T_d = [0, 7 \times rand(1)]^T$, **Figure 4(b)**. Dynamic and static friction torques were selected as follows:

$$F_d = \begin{bmatrix} 5\cos(x_1) & 0 \\ 0 & 3\cos(x_2) \end{bmatrix}, \quad F_s = \begin{bmatrix} 1.8 \ \text{sgn}(x_1) \\ 1.2 \ \text{sgn}(x_2) \end{bmatrix}$$
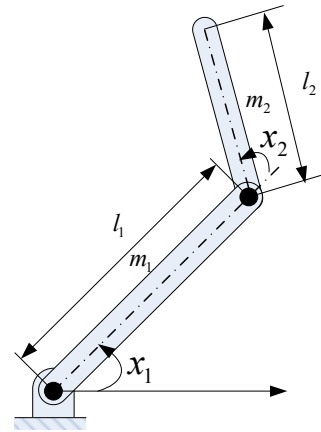

Figure 3. A two link rigid robot.

The friction and disturbance torques were unknown to the algorithm. Random signals were generated by the *rand* function in MATLAB.

The desired trajectories for x_1 and x_2 were set as:

$$x_1^d(t) = -A_1 \sin(\omega_1 t), \quad x_2^d(t) = -A_2 \sin(\omega_2 t),$$

with $A_1 = 1.2 \ rad$, $A_2 = 1.6$ rad, $\omega_1 = \pi / 2$ rad·s^{-1}, $\omega_2 = \pi$ rad·s^{-1}.

Initially, the arm is assumed at rest, *i.e.* $\dot{x}_{t=0} = [0, 0]^T$ rad·s^{-1}, and position of links as $x_{t=0} = [\pi/12, -\pi/12]^T$ rad, which resulted in initial position error $e_{t=0} = [15^\circ, -15^\circ]^T$ degree and velocity error $\dot{e}_{t=0} = [1.89, -5.03]^T$ rad·s^{-1}.

The AFSMC has been simulated under the following settings. Two rules were implemented to determine each of the two equivalent control components, *i.e.* $L = 2$ in (14). 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 a total of 4 rules were used to determine the two equivalent torques. This is relatively a quite small number of rules. In a similar study, *i.e.* adaptive fuzzy sliding mode control for nonlinear systems [25],

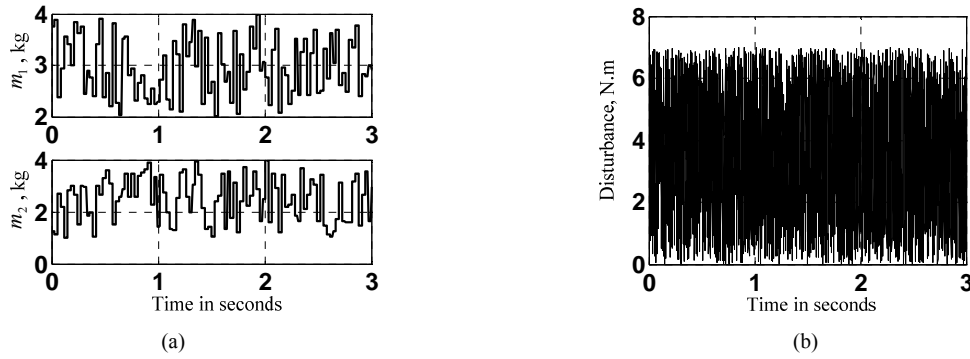　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　**ICA**

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

the rule base consists of 36 rules for a one degree of freedom system (the inverted pendulum).

Coefficients of the sliding surfaces in (12) were picked as $c_1 = [40, 40]^T$ and $c_2 = [3, 3]^T$. After few simulation tests, the learning rates were adjusted as $\eta_1^T = [15, 1.5]$ and $\eta_2^T = [45, 6]$. The estimated errors in (28) have been initiated as $\hat{E} = [-300, -100]^T$. As mentioned earlier, the sign function in (6) and (28) has been replaced by the saturation function with $\phi_i = 1$, $i = 1, 2$.

Evolution of the parameter vectors is given in **Figure 5(a)**. Zeros were used to initiate their elements. The superscripts denote the rule number, 1 and 2. The rates of adaptation for the parameter vectors are depicted in **Figure 5(b)**. As it can be noticed, the rate of adaptation of rule 1 is very close to rule 2 for the same joint. This remark was noticed by the authors from an enlarged version of **Figure 5(b)**. Time history of the estimated errors is shown in **Figure 6**.

With respect to SMC in (5), (6) and (10), we have simulated it under the following settings. The control system has been initiated with the same initial conditions (*i.e.* e and $\dot{e}$) followed by the AFSMC. Similar to what we did with respect to the AFSMC, the sign function in (6) has been replaced by the saturation function. The gain K of the hitting controller gain in (6) was set as $K = 70I$ 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. To synthesize the SMC, $\hat{M}(x)$ and $\hat{f}(x, \dot{x})$ in (10) were selected as follows: $\hat{M} = 5I$ which means that it is a time-independent matrix and

$$\hat{f} = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} \dot{x} + \begin{bmatrix} 2 \\ 1 \end{bmatrix} + F_d \dot{x} + F_s + T_d$$

where F_d, F_s and T_d are defined above.

Similar to AFSMC, the friction and disturbance torques were unknown to the control algorithm. Results are shown in **Figures 7-12**. A close look to these **Figures** shows that the AFSMC was little-bit faster than SMC. **Figure 12** depicts the control signals. In the transient phase, the maximum input torques of the SMC exhibits larger values than those of the AFSMC.

In order to quantify the performance of the two controllers, we have used the following three criteria.

1) Integral of the absolute value of error (IAE):

$$IAE = \int_0^{t_f} |e(t)| \, dt$$

2) Integral of time multiplied by the absolute value of the error (ITAE)
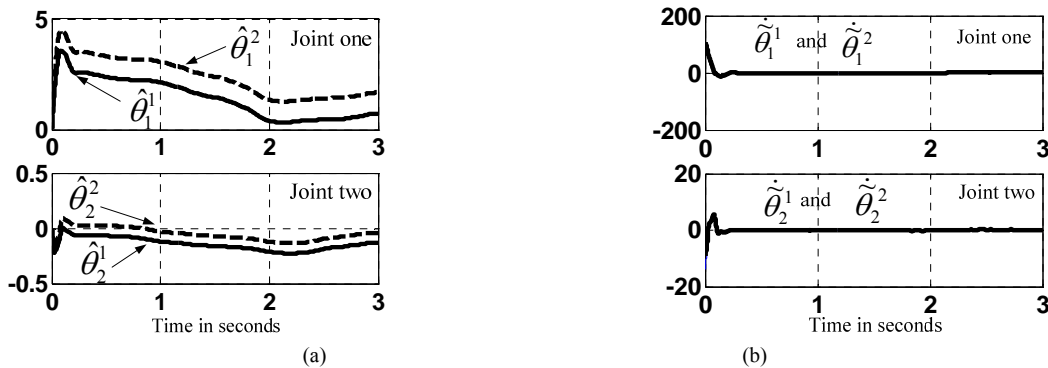
$$ITAE = \int_0^{t_f} t \cdot |e(t)| \, dt$$

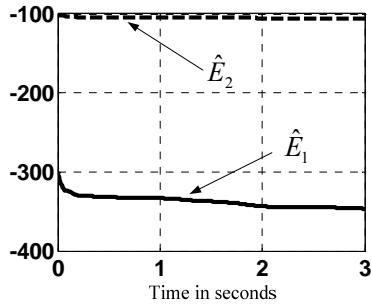Figure 5. Time history of (a) parameter vectors (*i.e.* control actions) and (b) adaptation rate.

Figure 6. Time history of the estimated errors.

3) Integral of the square value (ISV) of the control input

$$ISV = \int_0^{t_f} u^2(t)\,\mathrm{d}t$$

Both IAE and ITAE are used as objective numerical measures of tracking performance for an entire error curve, where t_f represents the total running time (3 seconds). The IAE criterion gives an intermediate result. In ITAE, time appears as a factor; it will heavily emphasize errors that occur late in time. The criterion ISV shows the consumption of energy. Results are given in **Table 1**.

These results slightly differ when we run the software more than one time under the same conditions. This is referred to the random signals involved in the simulation (masses of the links and the disturbances).

Nevertheless, one can clearly notice that the AFSMC out performs the SMC with respect to all the performance indices.

(a)

(b)
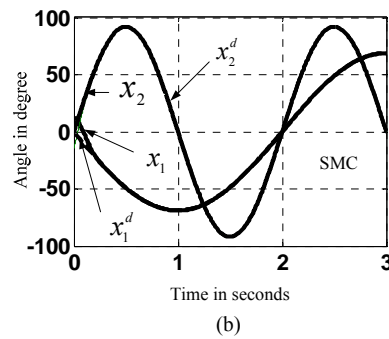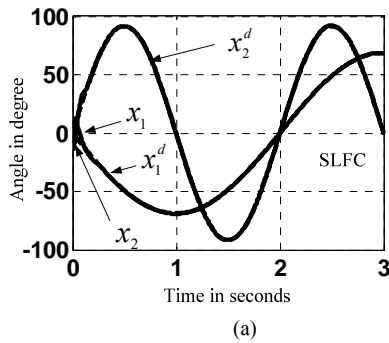
Figure 7. The desired joint angles, x^d and actual angles x.

(a)

(b)
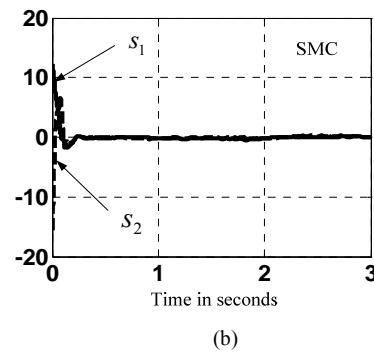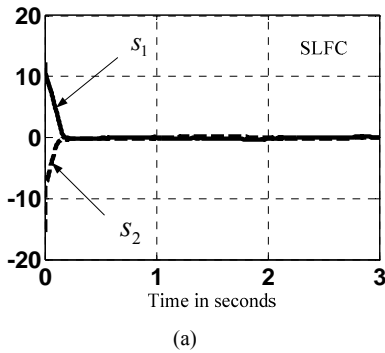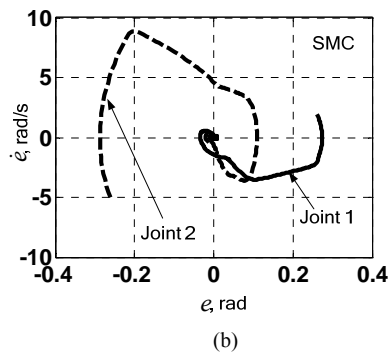
Figure 8. Time history of the sliding surfaces.
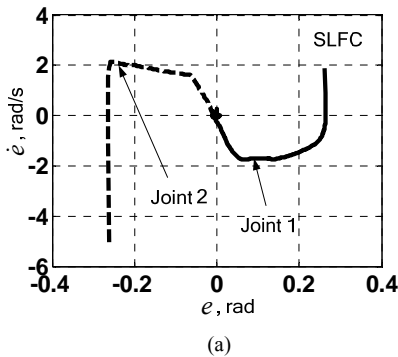

(a)
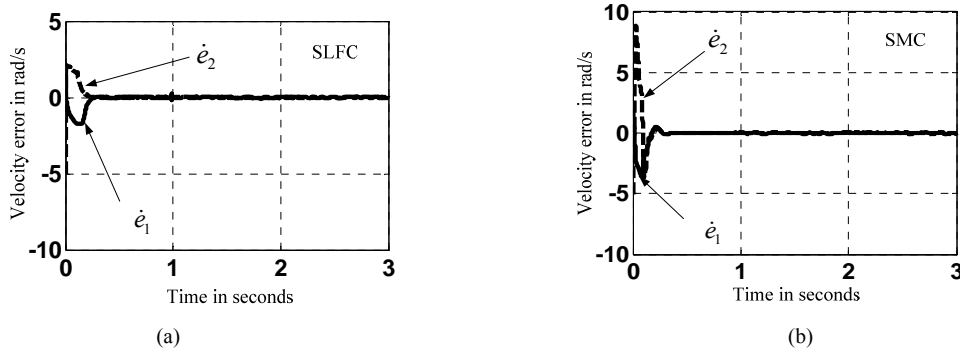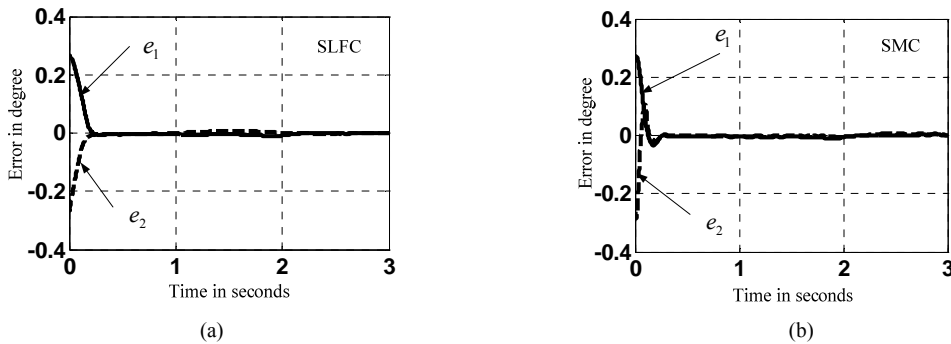

(b)

Figure 9. Phase plots.

Figure 10. Velocity tracking errors.

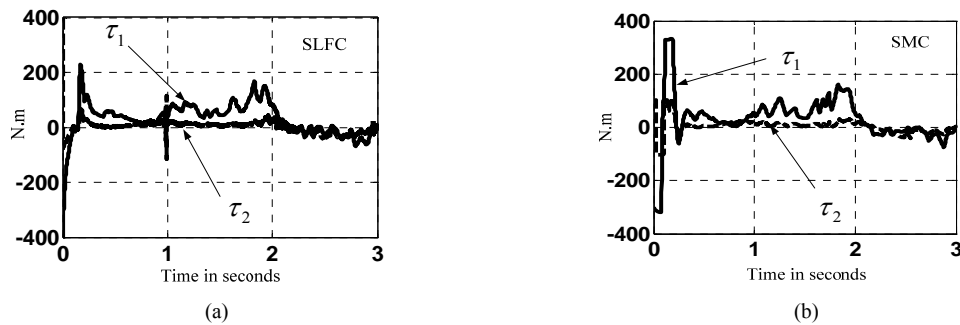Figure 11. Trajectory tracking errors.

Figure 12. The input torques.

Table 1. The performance indices.

Controller	Joint	IAE (rad) $\times 10^{-2}$	ITAE (rad·s^{-1}) $\times 10^{-2}$	ISV (N.m)$^2 \times 10^4$
AFSMC	Joint 1	2.80	1.6	1.447
	Joint 2	2.92	1.2	0.15
SMC	Joint 1	2.86	1.62	2.41
	Joint 2	4.26	2	0.37

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

5. Conclusions

In this article, we utilized the universal approximation property of fuzzy systems and powerfulness of SMC theory to compose an AFSMC scheme for robotic systems. Optimal parameters of the fuzzy system and uncertainty bound are generated on-line. The proposed control scheme has the following advantages: 1) does not require the system model; 2) guarantees the stability of the closed loop system; 3) uses a 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 control scheme is decoupled and has eliminated the assumptions, which are usually needed to synthesize a SMC.

The control scheme has been simulated 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.

6. References

[1]   A. B. Sharkawy, H. El-Awady and K. A. F. Moustafa, "Stable Fuzzy Control for a Class of Nonlinear Systems," *Tranactions of the Institute of Measurement and Control*, Vol. 25, No. 3, 2003, pp. 265-278. doi:10.1191/0142331203tm086oa

[2]   J. Lin, R.-J. Lian, C.-N. Huang and W.-T. Sie, "Enhanced Fuzzy Sliding Mode Controller for Active Suspension Systems," *Mechatronics*, Vol. 19, No. 7, 2009, pp. 1178-1190. doi:10.1016/j.mechatronics.2009.03.009

[3]   P. C. Chen, C. W. Chen and W. L. Chiang, "GA-Based Modified Adaptive Fuzzy Sliding Mode Controller for Nonlinear Systems," *Expert Systems with Applications*, Vol. 36, No. 3, 2009, pp. 5872-5879. doi:10.1016/j.eswa.2008.07.003

[4]   S. Kaitwanidvilai and M. Parnichkun, "Force Control in a Pneumatic System Using Hybrid Adaptive Neuro-Fuzzy Model Reference Control," *Mechatronics*, Vol. 15, No. 1, 2005, pp. 23-41. doi:10.1016/j.mechatronics.2004.07.003

[5]   L.-C. Hung and H.-Y. Chung, "Decoupled Sliding-Mode with Fuzzy-Neural Network Controller for Nonlinear Systems," *International Journal of Approximate Reasoning*, Vol. 46, No. 1, 2007, pp. 74-97. doi:10.1016/j.ijar.2006.08.002

[6]   Y. Zhiyong, W. Jiang and M. Jiangping, "Motor-Mechanism Dynamic Model Based Neural Network Optimized Computed Torque Control of a High Speed Parallel Manipulator," *Mechatronics*, Vol. 17, 2007, pp. 381-390. doi:10.1016/j.mechatronics.2007.04.009

[7]   J.-J. Slotine and W. Li, "Applied Nonlinear Control," Printice-Hall International, Inc., Upper Saddle River, 1991.

[8]   V. I. Utkin, "Sliding Modes and Their Applications in Variable Structure Systems," Mir Publishers, Moscow, 1978.

[9]   V. I. Utkin, "Sliding Modes in Control and Optimization," Springer-Verlag, Berlin, 1992.

[10] A. Harifi, A. Aghagolzadeh, G. Alizadel and M. Sadeghi, "Designing a Sliding Mode Controller for Slip Control of Antilock Brake Systems," *Transportation Research Part C*, Vol. 16, No. 6, 2008, pp. 731-741. doi:10.1016/j.trc.2008.02.003

[11] B. Yao, S. P. Chan and D. Wang, "Variable Structure Adaptive Motion and Force Control of Robot Manipulator," *Automatica*, Vol. 30, No. 9, 1994, pp. 1473-1477. doi:10.1016/0005-1098(94)90014-0

[12] B. Yao, S.P. Chan and D. Wang, "Unified Formulation of Variable Structure Control Schemes for Robot Manipulators," *IEEE Transactions on Automatic Control*, Vol. 39, No. 2, 1994, pp. 371-376. doi:10.1109/9.272337

[13] K. Passino and S. Yurkovich, "Fuzzy Control," Addison-Wesley Longman, Inc., Boston, 1998.

[14] R. Ordonez, J. Zumberge, J. T. Spooner and K. M. Passino, "Adaptive Fuzzy Control: Experiments and Comparative Analysis," *IEEE Transactions on Fuzzy Systems*, Vol. 5, No. 2, 1997, pp. 167-188. doi:10.1109/91.580793

[15] F. H. Hsiao, C. W. Chen, Y. W. Liang, S. D. Xu and W. L. Chiang, "T-S Fuzzy Controllers for Nonlinear Interconnected TMD Systems with Multiple Time Delays," I*EEE Transactions on Circuits and Systems*, Vol. 52, No. 9, 2005, pp. 1883-1893. doi:10.1109/TCSI.2005.852492

[16] F. H. Hsiao, J. D. Hwang, C. W. Chen and Z. R. Tsai, "Robust Stabilization of Nonlinear Multiple Time-Delay Large-Scale Systems via Decentralized Fuzzy Control," *IEEE Transactions on Fuzzy Systems*, Vol. 13, No. 1, 2005, pp. 152-163. doi:10.1109/TFUZZ.2004.836067

[17] H. F. Ho, Y. K. Wong and A. B. Rad, "Adaptive Fuzzy Sliding Mode Control with Chattering Elimination for SISO Systems," *Simulation Modeling Practice and Theory*, Vol. 17, No. 7, 2009, pp. 1199-1210. doi:10.1016/j.simpat.2009.04.004

[18] C.-C. Cheng and S.-H. Chien, "Adaptive Sliding Mode Controller Design Based on T-S Fuzzy System Models," *Automatica*, Vol. 42, No. 1, 2006, pp. 1005-1010. doi:10.1016/j.automatica.2006.02.016

[19] H. F. Ho, Y. K. Wong and A. B. Rad, "Robust Fuzzy Tracking Control for Robotic Manipulators," *Simulation Modelling Practice and Theory*, Vol. 15, No. 7, 2007, pp. 801-816. doi:10.1016/j.simpat.2007.04.008

[20] T.-H. S. Li and Y.-C. Huang, "MIMO Adaptive Fuzzy Terminal Sliding-Mode Controller for Robotic Manipulators," *Information Sciences*, Vol. 180, No. 23, 2010, pp. 4641-4660. doi:10.1016/j.ins.2010.08.009

[21] Y. Stephaneko and C.-Y. Su, "Variable Structure Control of Robot Manipulators with Nonlinear Sliding Manifolds," *International Journal of Control*, Vol. 58, No. 2, 1993, pp. 285-300. doi:10.1080/00207179308923003

[22] L. X. Wang, "Adaptive Fuzzy Systems and Control," PTR Prentice Hall, Upper Saddle River, 1994.

[23] S.-J. Huang and W.-C. Lin, "Adaptive Fuzzy Controller with Sliding Surface for Vehicle Suspension Control," *IEEE Transactions on Fuzzy Systems*, Vol. 11, No. 4, 2003, pp. 550-559. doi:10.1109/TFUZZ.2003.814845

[24] A. Poursamad and A. H. Davaie-Markazi, "Robust Adap-

tive Fuzzy Control of Unknown Chaotic Systems," *Applied Soft Computing*, Vol. 9, No. 3, 2009, pp. 970-976. doi:10.1016/j.asoc.2008.11.014

[25] M. Roopaei, M. Zolghadri and S. Meshksar, "Enhanced Adaptive Fuzzy Sliding mode Control for Uncertain Nonlinear Systems," *Communications in Nonlinear Science and Numerical Simulation*, Vol. 14, No. 9-10, 2009, pp. 3670-3681. doi:10.1016/j.cnsns.2009.01.029

Appendix A

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

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{bmatrix} + \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & 0 \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} + \begin{bmatrix} G_1 \\ G_2 \end{bmatrix}$$

where

$M_{11} = (m_1 + m_2)l_1^2 + m_2 l_2^2 + 2m_2 l_1 l_2 \cos x_2,$

$M_{12} = M_{21} = m_2 l_2^2 + m_2 l_1 l_2 \cos x_2,$

$M_{22} = m_2 l_2^2,$

$C_{11} = -2m_2 l_1 l_2 \dot{x}_2 \sin x_2,$

$C_{12} = -m_2 l_1 l_2 \dot{x}_2 \sin x_2,$

$C_{21} = m_2 l_1 l_2 \dot{x}_1 \sin x_2,$

$G_1 = (m_1 + m_2)gl_1 \cos x_1 + m_2 gl_2 \cos(x_1 + x_2),$

$G_2 = m_2 gl_2 \cos(x_1 + x_2),$

and $g = 9.8 \text{ m/sec}^2$ is the acceleration of gravity.