Robotica (2005) volume 23, pp. 635–644. © 2005 Cambridge University Press
doi:10.1017/S0263574704001079 Printed in the United Kingdom

Adaptive fuzzy sliding control for a three-link passive robotic manipulator

Subashini Elangovan and Peng-Yung Woo*

Electrical Engineering Department, Northern Illinois University, Dekalb, IL 60115 (USA)

(Received in Final Form: September 25, 2004)

SUMMARY
An adaptive fuzzy sliding control scheme is proposed to control a passive robotic manipulator. The motivation for the design of the adaptive fuzzy sliding controller is to eliminate the chattering and the requirement of pre-knowledge on bounds of error associated with the conventional sliding control. The stability and convergence of the adaptive fuzzy sliding controller is proven both theoretically and practically by simulations. A three-link passive manipulator model with two unactuated joints is derived to be used in the simulations. Simulation results demonstrate that the proposed system is robust against structured and unstructured uncertainties.

KEYWORDS: Fuzzy control; Passive manipulator; Sliding controller.

1. INTRODUCTION
An underactuated or passive manipulator is one in which some of the joints are not actuated. This could be the result of failure of some of the actuators, for example, a robot in space that has a dysfunctional joint actuator or that of a poor design. In many cases where the underactuated mechanism is due to failure of the joint, it is hard and expensive to repair or replace the failed actuator. A passive manipulator could also be constructed by choice to reduce energy consumption, weight and cost. The control of such underactuated manipulators has been gaining a lot of attention in the recent time. Most of the work done in this area is on manipulators mounted with some kind of braking mechanism on the passive joint. Research on purely passive systems with no braking mechanism is still in an infantile stage.

Usually, the number of degrees of freedom (dof) of a manipulator is the same as the number of actuators. However, for passive manipulators, the number of degrees of freedom is more than the number of actuators. The absence of actuators for some of the joints introduces nonholonomic constraints in the system.

Control of passive manipulators poses a considerable challenge due to their highly non-linear and coupled structure. However, there have been many successful works in controlling passive manipulators using various control strategies[1–4]. The first work[1] proposed a method of controlling passive manipulators equipped with brakes by using the dynamic coupling between the joints. The condition for controllability of manipulators with passive joints, a control algorithm and optimal control strategy are presented in reference [2]. Reference [3] presents the control of a three-link manipulator with two passive joints with sliding control and also presents a control algorithm. The control of a two-link passive manipulator with sliding control is discussed in reference [4].

Sliding control is a robust control scheme whose advantages include less computation and problem order reduction. However, the design of a sliding control requires knowledge on the bounds of the system disturbances and uncertainties. Another major problem with sliding control is the presence of chattering in the control input, which has a destabilizing effect of the system. Often, to solve this problem, smooth function control has been used. While this type of control alleviates the chattering problem in sliding control, it doesn't guarantee convergence of the output to the desired value[5,6].

Fuzzy control is recently becoming a popular control scheme, especially for complicated non-linear systems including robotic systems and electrical systems. The main reason for its popularity is its model free approach and ability to incorporate human knowledge to effectively control the systems in concern. In the field of robotics as well as others, there has been considerable research with successful results on the application of fuzzy control[6–12]. While fuzzy control is intuitive, its design is often based on trial and error and is not always optimal. It is also hard to theoretically prove the stability of the system with a fuzzy control. To tackle these issues, fuzzy control has been combined with other systems such as neural networks. The control of a robotic manipulator with a neural fuzzy controller is discussed in reference [7]. Another solution is to make fuzzy control adaptive by modeling the fuzzy system as an adaptor regressor model with a definition of an update algorithm based on the Lyapunov approach[8,9]. An adaptive fuzzy control scheme is proposed in reference [10] for compensating the nonlinear gravity component of a manipulator dynamics. There has also been work done in combining adaptive fuzzy control with conventional minimum variance control such as sliding control[11]. These types of systems have the combined benefits of sliding control and adaptive fuzzy control. By making the fuzzy sliding control adaptive, the system in concern is updated according to changing parameters such as the weight of the payload and friction. Such adaptive fuzzy control schemes are universal approximators, capable of modeling any continuous system within reasonable accuracy[8,13,14]. For these reasons, adaptive fuzzy sliding control is emerging as a

* Corresponding author: Fax 1,815,753–1289.

popular control scheme for nonlinear systems. Sun et al.[15,16] apply a fuzzy system to approximate the system dynamics of a robotic manipulator. There is a complicated term in their control input to provide the robust control. The discontinuous term sgn($\boldsymbol{s}$) still exists in the control input. Xu et al.[17] apply Takagi-Sugeno type fuzzy systems to estimate the system dynamics only. As is asserted in reference [8], this type of fuzzy systems may not provide a natural framework to represent human knowledge.

In this paper, an adaptive fuzzy sliding control is proposed and applied to a three-link passive manipulator. The motivation for the design of this controller is to eliminate the chattering and the requirement of pre-knowledge on bounds of error associated with the conventional sliding control. This approach is natural considering the complicated model of the passive manipulator, the presence of disturbances and the possibility of changes in the system parameters. The idea is to model the disturbances according to rules based on human knowledge about the relation between the disturbances and some measurable states of the system. These rules are constructed by observations in different kinds of control schemes such as computed torque control or conventional sliding control. Once the value of the disturbance is calculated, it is fed back to the system to cancel the actual disturbance.

In Section 2, the mathematical model of a three-link passive robotic manipulator is derived and the control strategy is discussed. In Section 3, sliding control is briefly discussed. In Section 4, the concept of fuzzy control is introduced and an adaptive fuzzy sliding control design is proposed. The rules used for simulation and the update algorithms are also discussed. The stability of the proposed system is proved using Lyapunov theorem. In Section 5, the simulation results of the conventional sliding control and the adaptive fuzzy sliding control are presented. Section 6 concludes the paper.

2. MATHEMATICAL MODEL AND PROCEDURE TO CONTROL A THREE-LINK PASSIVE ROBOTIC MANIPULATOR

The passive robotic manipulator in concern has three planar links of lengths l_1, l_2 and l_3, respectively, as shown in Figure 1. The first joint is equipped with an actuator, which is the only driving mechanism for all the three links. The second and the third joints are equipped with brakes of masses m_1 and m_2, respectively. These brakes help to lock the corresponding links into position. An end-effector of mass m_3 is attached to the end of the third link.

The dynamic equation of the three-link passive robotic manipulator is derived by using the Lagrange's equation and the equations for kinetic and potential energies[18,19]:

$$M(q)\ddot{q} + V(q, \dot{q}) + G(q) = \tau. \tag{2.1}$$

Or:

$$\begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} + \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ 0 \\ 0 \end{bmatrix} \tag{2.2}$$

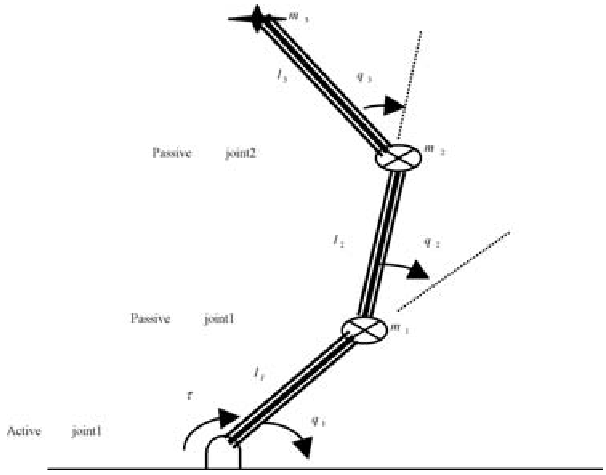

Fig. 1. A three-link passive robotic manipulator.

since the torques of the second and third joints, τ_2 and τ_3, are 0 when they are unactuated. $[c_1\ c_2\ c_3]^T$ represents the sum of the Coriolis/centripetal vector and the gravity vector. $\ddot{q}_1, \ddot{q}_2$ and $\ddot{q}_3$ represent the acceleration of the first, second and third link, respectively. Since there is only one actuator in the above manipulator, it is possible to control utmost one unactuated link at a time[2].

To control the passive link 3, the passive joint 2 should be locked at its current position, i.e.,

$$\dot{q}_2 = \ddot{q}_2 = 0. \tag{2.3}$$

The relationship between the acceleration of link 3 and the torque supplied at joint 1, with joint 2 locked into position, through mechanical coupling can be derived as follows. Substituting (2.3) into (2.2) yields

$$\ddot{q}_1 = -\frac{m_{33}}{m_{31}}\ddot{q}_3 - \frac{c_3}{m_{31}}. \tag{2.4}$$

Substituting (2.4) into (2.2) yields

$$\ddot{q}_3 = \left(m_{13} - \frac{m_{33}}{m_{31}}m_{11}\right)^{-1}\left(\tau_1 - c_1 + \frac{m_{11}}{m_{31}}c_3\right) \tag{2.5}$$

(2.5) gives the relation between the acceleration of link 3 and the torque provided by the actuator at link 1. Using (2.5), link 3 can be controlled.

To control the passive link 2, the passive joint 3 should be locked at its current position, i.e.,

$$\dot{q}_3 = \ddot{q}_3 = 0. \tag{2.6}$$

Substituting (2.6) into (2.2) yields

$$\ddot{q}_1 = -\frac{m_{22}}{m_{21}}\ddot{q}_2 - \frac{c_2}{m_{21}}. \tag{2.7}$$

Substituting (2.7) into (2.2) yields

$$\ddot{q}_2 = \left(m_{12} - \frac{m_{11}m_{22}}{m_{21}}\right)^{-1}\left(\tau_1 - c_1 + \frac{m_{11}}{m_{21}}c_2\right). \tag{2.8}$$

https://doi.org/10.1017/S0263574704001079 Published online by Cambridge University Press

Using (2.8), link 2 can be controlled by the torque provided by the actuator at link1.

To control link 1, joints 2 and 3 should be locked at their current positions, thereby giving

$$\dot{q}_2 = \ddot{q}_2 = \dot{q}_3 = \ddot{q}_3 = 0. \tag{2.9}$$

Substituting (2.9) into (2.2) yields

$$\ddot{q}_1 = \frac{\tau_1}{m_{11}} - \frac{c_1}{m_{11}}. \tag{2.10}$$

Using (2.10), link 1 can be controlled.

3. SLIDING CONTROL

Sliding control is a nonlinear control based on the remark that it is easier to control a 1st-order system than an nth-order system. It is designed to drive the system towards a sliding surface, which is a surface where the error and its derivatives are zero. By making the problem of system tracking equivalent to the system states remaining on the sliding surface, the nth-order tracking problem is reduced to a 1st-order stabilization problem[5]. Furthermore, the sliding surface is an invariant set meaning that, once the system reaches the sliding surface, it continues to remain on the sliding surface. Consider an nth-order system given by

$$x^n = f(\bar{x}) + b(\bar{x})u \tag{3.1}$$

where

$$x^n = \frac{d^n x}{dt^n} \tag{3.1.1}$$

$\bar{x} = [x^{n-1}, \ldots, x]$ represents the states of the system, and $f(\bar{x})$ and $b(\bar{x})$ are nonlinear functions that are not exactly known.

The nominal model of the system is a model of the system with approximate values taken for the unknown parameters. The nominal model for the nonlinear system in (3.1) is

$$x^n = \hat{f}(\bar{x}) + \hat{b}(\bar{x})u \tag{3.2}$$

where $\hat{f}(\bar{x})$ and $\hat{b}(\bar{x})$ are the estimates of $f(\bar{x})$ and $b(\bar{x})$, respectively. The aim of the sliding control is to make the system states $\bar{x}$ track specific time-varying states

$$\bar{x}_d = \left[x_d^{n-1}, \ldots, x_d\right]. \tag{3.3}$$

The sliding surface for an nth-order system is given by

$$S = \left(\frac{d}{dt} + \lambda\right)^{n-1} e \tag{3.4}$$

where $\lambda > 0$, e is the position error defined as

$$e = x - x_d \tag{3.4.1}$$

where x_d is the desired position. For a 2nd-order system, the sliding surface S is given by

$$S = \dot{e} + e. \tag{3.5}$$

The design of the sliding mode control (SMC) consists of two phases. The first phase is to design a nominal or equivalent control, similar to feedback linearization control or inverse control, which is a continuous control that maintains the system on the sliding surface when the system parameters are exactly known. A necessary condition for the system state trajectory to remain on the surface S is

$$\dot{S} = e^n + \sum_{i=1}^{n-1} \gamma^i \binom{n-1}{i} e^{(n-i)} = f(\bar{x}) + b(\bar{x})\,u - x_d^n + \sum_{i=1}^{n-1} \gamma^i \binom{n-1}{i} e^{(n-i)} = 0. \tag{3.6}$$

If $f(\bar{x})$ and $b(\bar{x})$ are not exactly known, then the best approximation of the equivalent control that satisfies the above condition for the nominal model given in (3.2), is given by

$$u_{eq} = \hat{b}(\bar{x})^{-1}\left(-\hat{f}(\bar{x}) + x_d^n - \sum_{i=1}^{n-1} \gamma^i \binom{n-1}{i} e^{(n-i)}\right). \tag{3.7}$$

The second phase is to design a control law that drives the system to the sliding surface in the presence of disturbances or uncertainities or when the system RP(representative point) is not on the sliding surface. This phase consists of adding a discontinuous term to the equivalent control in (3.7). The final sliding controller is then given by

$$\begin{aligned} u_{re} &= u_{eq} - \hat{b}(\bar{x})^{-1} K \operatorname{sgn}(S) \\ &= \hat{b}(\bar{x})^{-1}\left(\left(-\hat{f}(\bar{x}) + x_d^n - \sum_{i=1}^{n-1} \gamma^i \binom{n-1}{i} e^{(n-i)}\right) - K \operatorname{sgn}(S)\right) \end{aligned} \tag{3.8}$$

where

$$\operatorname{sgn}(S) = \begin{cases} 1 & \text{if} \quad S > 0 \\ 0 & \text{if} \quad S = 0 \\ -1 & \text{if} \quad S < 0 \end{cases} \tag{3.8.1}$$

and $K > 0$. This controller for the nominal system ensures the reaching condition described below

$$\frac{1}{2}\frac{d}{dt} S^2 = S^T \dot{S} \leq -\eta\,|S| \tag{3.9}$$

where $\eta > 0$. The above condition states that the squared distance to the sliding surface decreases along all system trajectories. The design of K is a critical step in sliding control. In order to ensure the reaching condition defined in (3.9) and hence the stability of the system, the value of K must be greater than the magnitude of the disturbances and uncertainties. The stability of the system after chosing K can be proven by using the Lyapunov theorem.[19]

One problem with sliding control is that often it is not possible to know the maximum magnitude of disturbances. For example, in the case of a robotic manipulator, the frictional forces are not determined and the payload of the end-effector may change in the future. If such a situation occurs and the disturbances exceed the value that K is designed for, then the system may become unstable. Another major problem is that in sliding control we set the value of K such that it is much higher than the magnitude of the disturbances and uncertainties. When K is sufficiently large, the control action defined by (3.8) is to push the system trajectories towards the sliding surface as can be seen in the simulation results. However, once the system RP reaches the sliding surface, the large K results in chattering. This is caused by the discontinuity in the sgn(S) function in combination with the large value of K. The greater is the value of K, the more is the chattering magnitude. The chattering becomes more apparent as e approaches zero and thus introduces high frequency dynamics in the system so that stability is lost.

4. ADAPTIVE FUZZY SLIDING CONTROL FOR PASSIVE ROBOTIC MANIPULATORS

4.1. Introduction to fuzzy control

Fuzzy control provides a way to incorporate the heuristic human knowledge about the behavior of a system to a controller. Fuzzy control does not require a mathematical model of the system in concern, but, instead, a practical information about the relationship between the states of the system and the parameters to be controlled. Thus, it serves as an effective control scheme for systems whose mathematical model is complicated such as a robotic manipulator. Also, from the viewpoint of implementation, fuzzy controllers enhance the application of parallel processing. A typical fuzzy controller consists of three parts, namely, Fuzzifier, Fuzzy Inference and Defuzzifier.

A fuzzifier maps an input point in the input space $U \in R^n$ to the fuzzy set A_x. Intuitively, during the fuzzification process the input is assigned a membership function based on the ambiguraity of the data. There are at least 2 types of fuzzifiers, namely, the singleton and the nonsingleton fuzzifier. The singleton fuzzifier maps a crisp input to a singleton fuzzy set. With the singleton fuzzifier an input $\mathbf{x}$ is assigned a membership value $\mu_{A_x}(\mathbf{x})$ according to the following rule

$$\mu_{A_x}(\mathbf{x}) = \begin{cases} 1 & \textit{for} \quad \mathbf{x} = x' \\ 0 & \textit{for} \quad \mathbf{x} \neq x' \end{cases} \tag{4.1}$$

where x' is the support of A_x. In nonsingleton fuzzifiers, $\mu_{A_x}(\mathbf{x}) = 1$ for $\mathbf{x} = x'$ and decreases as $\mathbf{x}$ moves away from x'. Gaussian and Triangular membership functions are commonly used in these types of fuzzifiers. Nonsingleton fuzzifiers are usually used in systems where the input is noisy [8].

The inference engine performs the mapping from fuzzy sets in the input space $U \in R^n$ to fuzzy sets in the output space $V \in R$ based on the fuzzy rule base. The fuzzy rule base is a collection of fuzzy rules which are "If-Then" rules based on heuristic knowledge about the system. A typical fuzzy rule is of the following form:

$$\begin{aligned} R_l &: \text{If } x_1 \text{ is } A_1^l, x_2 \text{ is } A_2^l, \ldots, x_n \text{ is } A_n^l \\ &\quad \text{then the output } y \text{ is } B^l \end{aligned} \tag{4.2}$$

where $\mathbf{x} = [x_1 \ldots x_n]^T \in U$ is the input to the fuzzy logic system, $y \in R$ is the output of the fuzzy logic system, A_i^l and B^l are fuzzy sets associated with the inputs and output, respectively. The superscript l refers to the l th rule out of the m rules $(1, 2, \ldots, M)$ in the rule base. Each if-then rule is an implication operation $A_1^l \times \ldots \times A_n^l \rightarrow B^l$. The implication operation in fuzzy logic is analogous to the "intersection" operation($\cap$) in the conventional logic theory. There are several ways to compute the implication, but all of them give results similar to the intersection operation in bivalued logic, though at different degrees.

The defuzzifier maps fuzzy sets obtained from the fuzzy inference engine in the output space $V \in R$ into a crisp output value. There are several choices of defuzzification methods available.[8]

4.2. Fuzzy sliding control

In this paper we consider a fuzzy system consisting of the singleton fuzzifier (4.1), the product-operation inference and the center-average defuzzifier. This combination yields the crist output of the fuzzy system to be[8]

$$y^* = \frac{\sum_{l=1}^{M} \bar{y}^l \left(\prod_{i=1}^{n} \mu_{A_i^l(x_i)} \right)}{\sum_{l=1}^{M} \left(\prod_{i=1}^{n} \mu_{A_i^l(x_i)} \right)}. \tag{4.3}$$

The rule base that is used by the fuzzy inference engine for the system under consideration has one input, namely, the function S defined in (3.4) and one output, namely, the measure of disturbance and/or uncertainity d. The fuzzy set consists of 5 membership functions for the input and the output, respectively. They are:

Input fuzzy sets (4.4)
VP for Very Postive
P for Positive
Z for Zero
N for Negative
VN for Very Negative

Output fuzzy sets (4.5)
PL for Positive Large
P for Positive
Z for Zero
N for Negative
NL for Negative Large.

The five input membership functions are chosen to be Gaussian membership functions as shown in Figure 2 and the five output membership functions are chosen to be triangular membership functions as shown in Figure 3.

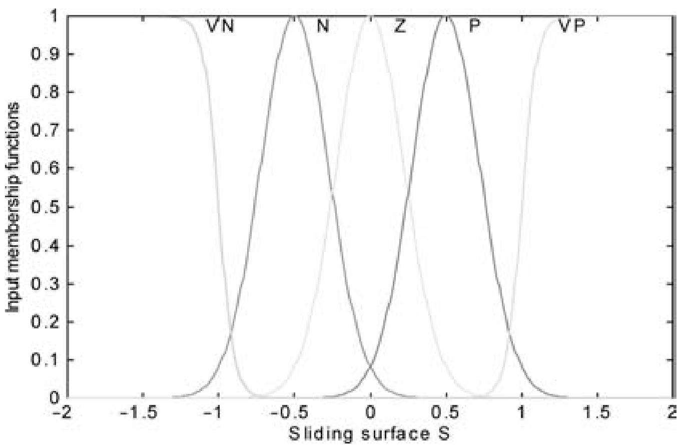

Fig. 2. Input membership functions.

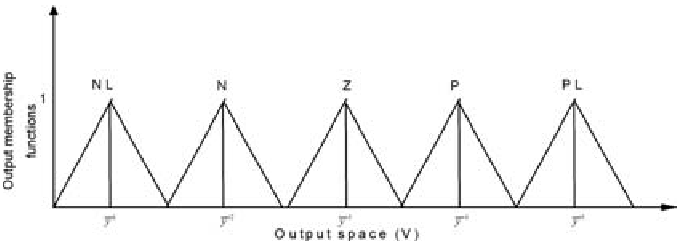

Fig. 3. Output membership functions.

The following rules of the rule base are constructed based on the observation of results obtained from the conventional sliding control.

If S is very positive, then d is positive large
If S is positive, then d is positive
If S is zero, then d is zero (4.6)
If S is negative, then d is negative
If S is very negative, then d is negative large

where S is the sliding function defined in (3.4) and $d \in R$ is the output of each rule. Since we base the rules of this fuzzy system on the sliding function S, this system is known as a fuzzy sliding system.

The shapes and supports of the membership functions are important parameters in the design of fuzzy controllers. However, in conventional fuzzy control design there is no well-defined method for them and often the design is done by trial and error. If the membership functions are not properly designed, it is difficult to get good tracking results.

4.3. Adaptive fuzzy sliding control for a passive robotic manipulator

In this section, an adaptive fuzzy sliding control scheme, which gets triggered whenever the plant deviates from the sliding surface pushing the system towards the sliding surface, is proposed. As is known, a fuzzy system can be made to be adaptive to maintain consistent performance in situations where there is large uncertainty or unknown variations in the plant parameters.

The dynamic equation of the three-link passive manipulator discussed in (2.2) is rewritten as

$$\begin{bmatrix}\ddot{q}_1\\ \ddot{q}_2\\ \ddot{q}_3\end{bmatrix} = -\begin{bmatrix} m_{11} & m_{12} & m_{13}\\ m_{21} & m_{22} & m_{23}\\ m_{31} & m_{32} & m_{33}\end{bmatrix}^{-1}\begin{bmatrix}c_1\\ c_2\\ c_3\end{bmatrix} + \begin{bmatrix} m_{11} & m_{12} & m_{13}\\ m_{21} & m_{22} & m_{23}\\ m_{31} & m_{32} & m_{33}\end{bmatrix}^{-1}\begin{bmatrix}\tau_1\\ 0\\ 0\end{bmatrix} \tag{4.7}$$

which is of the form, that is the same as in (3.1):

$$x^n = f(\bar{x}) + b(\bar{x})u \tag{4.8}$$

where

$$f(\bar{x}) = -\begin{bmatrix} m_{11} & m_{12} & m_{13}\\ m_{21} & m_{22} & m_{23}\\ m_{31} & m_{32} & m_{33}\end{bmatrix}^{-1}\begin{bmatrix}c_1\\ c_2\\ c_3\end{bmatrix} \tag{4.8.1}$$

and

$$b(\bar{x}) = \begin{bmatrix} m_{11} & m_{12} & m_{13}\\ m_{21} & m_{22} & m_{23}\\ m_{31} & m_{32} & m_{33}\end{bmatrix}^{-1} \tag{4.8.2}$$

where $q \in \mathrm{R}^{\mathrm{n}}$ represents the joint positions of the system and $u \in \mathrm{R}$ is the control input vector. Thus, the above system is a 2nd-order system and the S for the above system is given in (3.5). Here, it should be emphasized that (4.8) actually is the general form of the dynamic equation of an n-link robotic manipulator, though, in a genaral situation, u is not necessarily to be a vector with one non-zero component only as in this case expressed by (4.7). Correspondingly, the derivations hereafter are good for any n-link robotic manipulator expressed by (4.8).

Consider the system represented in (4.8) along with disturbances. The disturbances could be a combination of structured and unstructured uncertainties. Structured or parametric uncertainties represent the inaccuracies in the terms in the system model like payload mass, while unstructured uncertainties or unmodeled dynamics represent inaccuracies in the estimation of the system order. Such a system can be represented by the following differential equation

$$x^n = \hat{f}(\bar{x}) + \Delta f(\bar{x}) + \hat{b}(\bar{x})[u(t) + h(\bar{x})] \tag{4.9}$$

where $x^n = \frac{d^n x}{dt^n}$, $\hat{f}(\bar{x})$ and $\hat{b}(\bar{x})$ are the nominal values of $f(\bar{x})$ and $b(\bar{x})$, respectively. $\Delta f(\bar{x})$ represents the uncertainties in $f(\bar{x})$, $u(t)$ is the input vector and $h(\bar{x})$ represents the disturbances and unmodeled dynamics of the input. Assume that there exists $z(\bar{x})$ such that

$$\Delta f(\bar{x}) = \hat{b}(\bar{x})z(\bar{x}). \tag{4.10}$$

https://doi.org/10.1017/S0263574704001079 Published online by Cambridge University Press

Then we can rewrite (4.9) as

$$x^n = \hat{f}(\bar{x}) + \hat{b}(\bar{x})u(t) + d(\bar{x}) \tag{4.11}$$

where

$$d(\bar{x}) = \hat{b}(\bar{x})[z(\bar{x}) + h(\bar{x})] \tag{4.11.1}$$

According to reference [8], for each component $d_i(\bar{x})$ $(i = 1, \ldots, n)$ in vector $d(\bar{x})$, there exists a fuzzy system $d_{ifz}(\bar{x})$ with respect to any given $\varepsilon \geq 0$ such that

$$\forall \bar{x} \in U |d_i(\bar{x}) - d_{ifz}(\bar{x})| \leq \varepsilon \tag{4.12}$$

where ε is known as the minimum approximation error [3]. If sufficient number of rules are used, ε is proven to be small based on the universal approximation theorem in reference [12] and [13]. In the following derivations it is assumed that its value is specified. Due to (4.12), there exists a fuzzy system for the given S_i, which is the ith component of S defined in (3.5) $(i = 1, \ldots, n)$.

$$\begin{aligned} d_{ifz}(\bar{x}) &= \sum_{l=1}^{M} \theta_l \phi_l(\bar{x}) \\ &= \theta_i^T \phi_i(\bar{x}) \end{aligned} \tag{4.13}$$

such that

$$\begin{aligned} d_i(\bar{x}) &= \sum_{l=1}^{M} \theta_l \phi_l(\bar{x}) + D_i(\bar{x}) \\ &= \theta_i^T \phi_i(\bar{x}) + D_i(\bar{x}) \end{aligned} \tag{4.14}$$

where

$$|D_i(\bar{x})| < \varepsilon, \tag{4.14.1}$$

and $\theta_i^T = [\theta_1 \,..\, \theta_M]$ is a parameter vector and $\phi_i^T = [\phi_1 \,..\, \phi_M]$ is a regressor vector.

Given the sliding surface in (3.4), in the absence of disturbances and uncertainties, the equivalent control needed to satisfy the condition $\dot{S} = 0$ is given in (3.7). However, if the initial state is not on the sliding surface or if the system RP has deviated from the sliding surface due to parameter variations and/or disturbances, the controller must be designed so that it drives the system towards the sliding suface. Such a controller using sliding control is given in (3.8), but the simulation results using this controller demonstrate undesirable chattering. Another problem is that to design such a controller we need to know the upperbound of the disturbance, which in many practical situations is very difficult. Thus, a controller u_{re}, which does not have any chattering and drives the system trajectories to the reaching condition (3.9), needs to be designed.

In order to achieve the above sliding condition we propose a control scheme which combines the controller in (3.8) with a fuzzy system as given in the following

$$u_{re} = \hat{b}(\bar{x})^{-1}\left(-\hat{f}(\bar{x}) + x_d^n - \sum_{i=1}^{n-1} \gamma^i \binom{n-1}{i} e^{(n-1)} - \hat{d}_{fz}(\bar{x}) - k\,\mathrm{sgn}(S) \right) \tag{4.15}$$

where for each component $\hat{d}_{ifz}(\bar{x})(i = 1, \ldots, n)$ in vector $\hat{d}_{fz}(\bar{x})$

$$\begin{aligned} \hat{d}_{ifz}(\bar{x}) &= \sum_{l=1}^{M} \hat{\theta}_l \phi_l(\bar{x}) \\ &= \hat{\theta}_i^T \phi_i(\bar{x}), \end{aligned} \tag{4.15.1}$$

which is the fuzzy estimate of $d_{ifz}(\bar{x})$ in (4.13), is obtained from (4.3) by considering $\prod_{i=1}^{n} \mu_{A_i^l(x_i)} / \sum_{l=1}^{M} \prod_{i=1}^{n} \mu_{A_i^l(x_i)}$ as the regressor $\phi_l(\bar{x})$ and $\bar{y}^l$ as the parameter $\hat{\theta}_l$. The update rule of the parameter vector $\hat{\theta}_i$ is specified by

$$\dot{\hat{\theta}}_i = \lambda S_i \phi_i(\bar{x}) \tag{4.15.2}$$

with $\lambda > 0$

$$k = \varepsilon + \eta \tag{4.15.3}$$

where ε is the minimum approximation error, which is a very small value and is assumed to be known.

The overall system consisting of the adaptive fuzzy sliding controller and the passive robotic manipulator is given in Figure 4.

4.4. Stability proof for adaptive fuzzy sliding control

In this section, the asymptotic stability of the system is proved by LaSalle's theorem with the use of a Lyapunov function.

LaSalle's Theorem:

Given the autonomous nonlinear system

$$x^n = f(\bar{x}), \quad \bar{x}(0) = x_0 \tag{4.16}$$

with the equilibrium at the origin, then:

Asymptotic Stability: Suppose that a Lyapunov function $V(\bar{x})$ has been found such that for $\bar{x} \in N \subset R^n$, $V(\bar{x}) > 0$ and $\dot{V}(\bar{x}) \leq 0$, then the origin is asymptotically stable if and only if $\dot{V}(\bar{x}) = 0$ only at $\bar{x} = 0$.

Consider a Lyapunov function

$$V = \frac{1}{2} S^T S + \frac{1}{2\lambda} \sum_{i=1}^{n} \tilde{\theta}_i^T \tilde{\theta}_i \tag{4.17}$$

where $\tilde{\theta}_i = \hat{\theta}_i - \theta_i$ $(i = 1, \ldots, n)$ is the difference between the nominal parameter vector $\hat{\theta}_i$ in (4.15.1) and the actual parameter vector θ_i in (4.13). The derivative of (4.17) is given by

$$\dot{V} = S^T \dot{S} + \frac{1}{\lambda} \sum_{i=1}^{n} \tilde{\theta}_i^T \dot{\hat{\theta}}_i \tag{4.18}$$

since $\dot{\tilde{\theta}}_i = \dot{\hat{\theta}}_i$.

Applying (4.11) to the equation of $\dot{S}$ in (3.6) and putting the $\dot{S}$ into (4.18), we have

$$\dot{V} = S^T \left[\hat{f}(\bar{x}) + \hat{b}(\bar{x})u(t) + d(\bar{x}) - x_d^n + \sum_{i=1}^{n-1} \gamma^i \binom{n-1}{i} e^{(n-i)} \right] + \frac{1}{\lambda} \sum_{i=1}^{n} \tilde{\theta}_i^T \dot{\hat{\theta}}_i. \tag{4.19}$$

https://doi.org/10.1017/S0263574704001079 Published online by Cambridge University Press

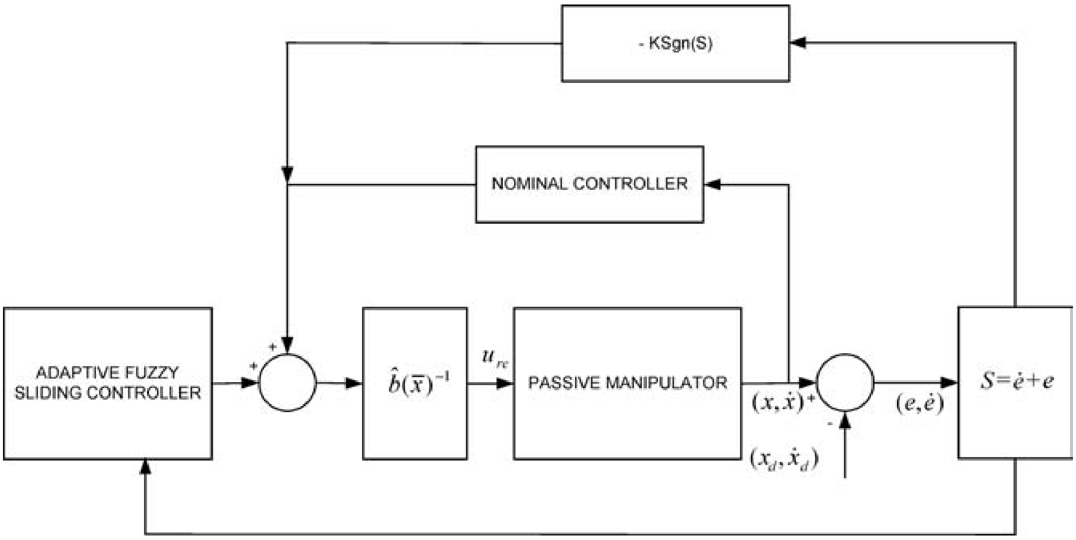

Fig. 4. Overall system architecture.

Applying the input function in (4.15) to (4.19) and simplifying the result leads to

$$\dot{V} = S^T(d(\bar{x}) - \hat{d}_{fz}(\bar{x}) - k\,\mathrm{sgn}(S)) + \frac{1}{\lambda}\sum_{i=1}^{n}\tilde{\theta}_i^T\dot{\hat{\theta}}_i. \quad (4.20)$$

Applying (4.14) and (4.15.1) to (4.20) leads to

$$\dot{V} = S^T\left[\begin{pmatrix} -\tilde{\theta}_1\phi_1(\bar{x}) \\ \cdot \\ \cdot \\ \cdot \\ -\tilde{\theta}_n\phi_n(\bar{x}) \end{pmatrix} + \begin{pmatrix} D_1(\bar{x}) \\ \cdot \\ \cdot \\ \cdot \\ D_n(\bar{x}) \end{pmatrix} - k\,\mathrm{sgn}(S)\right] + \frac{1}{\lambda}\sum_{i=1}^{n}\tilde{\theta}_i^T\dot{\hat{\theta}}_i. \quad (4.21)$$

Applying the parameter update rule in (4.15.2) to (4.21) leads to

$$\dot{V} = S^T(D(\bar{x}) - k\,\mathrm{sgn}(S)) \le |S|(|D(\bar{x})| - k). \quad (4.22)$$

Due to the definition of k in (4.15.3), we have

$$\dot{V} \le -\eta\,|S|\,. \quad (4.23)$$

Hence, it is seen that $\dot{V}$ is negative semi-definite, becoming zero only when $S=0$. From the definition of S, it is seen that $S=0$ implies that $e=0$ and $\dot{e}=0$. In reference [6] it is proven that given the results of the Lyapunov analysis above, $\lim_{t\to\infty} S = \lim_{t\to\infty}(e + \sum_{i=1}^{n-1}\lambda^i\binom{n-1}{i}e^{(n-i)}) = 0$. Thus, the overall system is asymptotically stable.

5. SIMULATIONS AND DISCUSSIONS

The adaptive fuzzy sliding control scheme proposed in Section 4 is applied to control the three-link passive manipulator with two unactuated joints shown in Figure 1, whose mathematical model is derived in Section 2. The performance and stability of the system are verified by simulations. The simulation results are validated against those obtained for a conventional sliding controller. The parameters of the length and mass of the links are

Length of link1 – 1.51 (m)
Length of link2 – 1.3 (m)
Length of link3 – 1.2 (m)
Mass of brake1 – 1 (Kg)
Mass of brake2 – .75 (Kg)
Mass of end-effector –.65 (Kg)

Both structured and unstructured uncertainties are applied to the system. The structured uncertainty consists of a payload change from 0.65 Kg to 2.5 Kg at time $t=12$ sec. The unstructured uncertainty represents dynamic friction, random disturbance and possible unmodelled high frequency components of the dynamics. The plant model with parametric inaccuracies as given in (4.11) is

$$x^n = \hat{f}(\bar{x}) + \hat{b}(\bar{x})u(t) + d(\bar{x}). \quad (5.1)$$

By assuming the actual model parameters as

$$f(\bar{x}) = (1 + 0.9\sin(2\pi t/3))\hat{f}(\bar{x}) \quad (5.2)$$

$$b(\bar{x}) = (1 + 0.2\sin(2\pi t/3))\hat{b}(\bar{x}) \quad (5.3)$$

the disturbance $d(\bar{x})$ is given by

$$d(\bar{x}) = 0.9\sin(2\pi t/3)\hat{f}(\bar{x}) + 0.2\sin(2\pi t/3)\hat{b}(\bar{x})u(t), \quad (5.4)$$

which is changing with respect to time. In our simulations, the third link is expected to track a sinusoidal trajectory in the presence of these uncertainties. An initial error of 1 rad is assumed for the third link. A 4th-order Runge Kutta integration is used to simulate the three-link passive manipulator. The software is written in Matlab.

https://doi.org/10.1017/S0263574704001079 Published online by Cambridge University Press

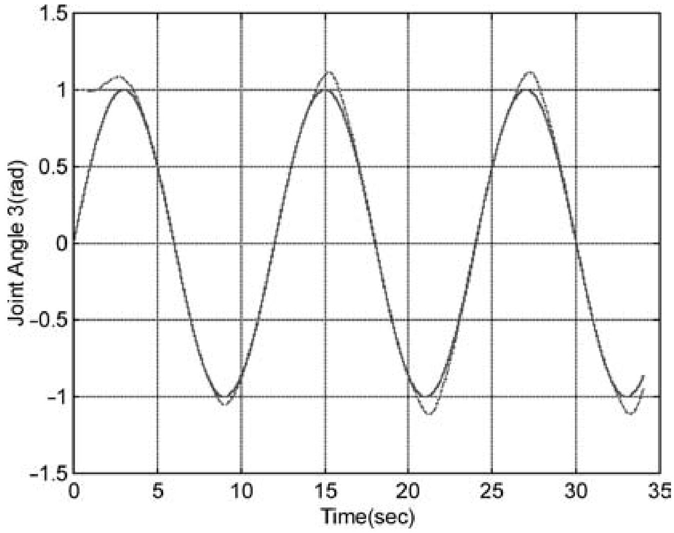

Fig. 5. Dynamic tracking of joint 3 by CSMC with payload change at time = 12 sec and unstructured uncertainty.

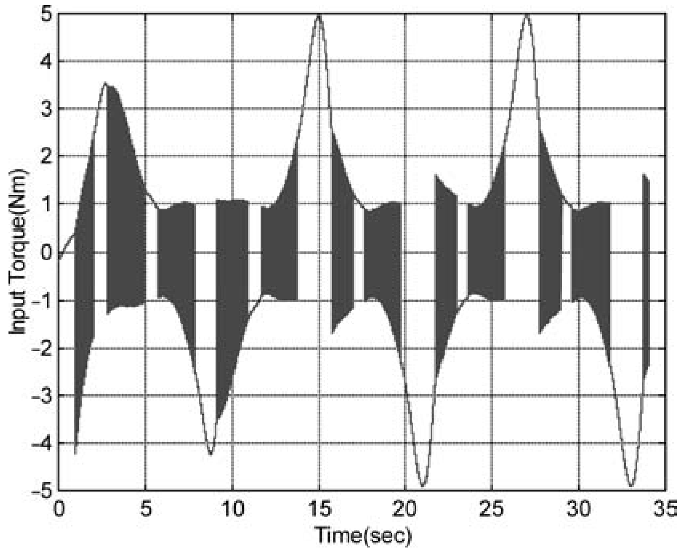

Fig. 6. Input torque for dynamic tracking of joint 3 by CSMC with payload change and unstructured uncertainty.

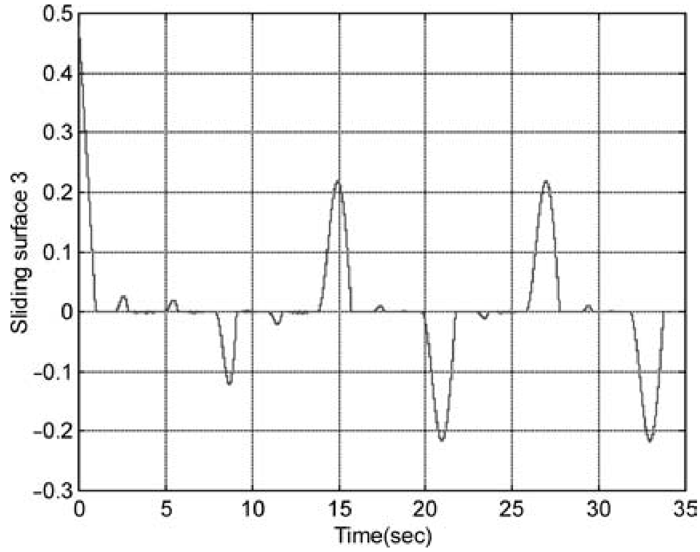

Fig. 7. Sliding surface for dynamic tracking of joint 3 by CSMC with payload change and unstructured uncertainty.

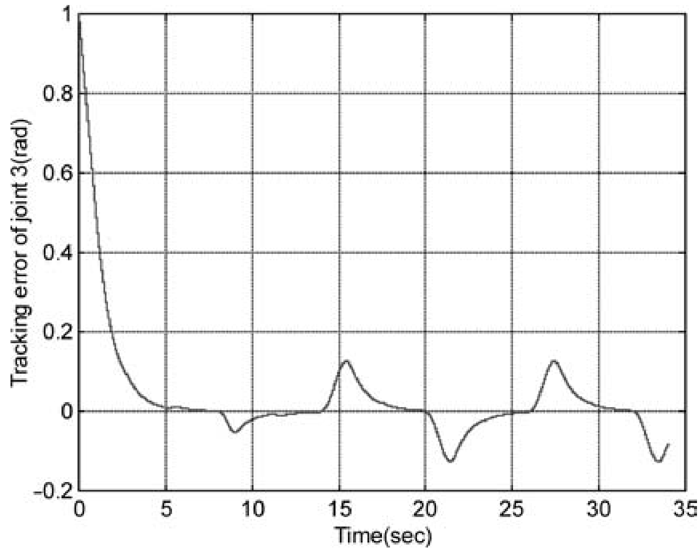

Fig. 8. Tracking error for dynamic tracking of joint 3 by CSMC with payload change and unstructured uncertainty.

First, the conventional sliding control discussed in Section 3 is applied to control the three-link passive robotic manipulator as shown in Figures 5–8. The value of the parameter K is chosen keeping as small as possible to reduce the magnitude of chattering. The input torque u(Nm) is calculated per (3.8).

Figure 8 shows a tracking error that does not seem to converge after prolonged observation. A larger value of K for the simulation is able to reduce this error. However, it is practically hard to estimate the magnitude of unstructured uncertainty and hence, it is hard to design a suitable value of K. If the value of K is not substantially high, the tracking performance degrades over time and the system becomes unstable. Furthermore, Figure 6 shows a big amount of chattering in the input torque. This kind of chattering has a lot of disadvantages and needs to be alleviated.

The adaptive fuzzy sliding control proposed in Section 4 is then applied to control the three-link passive robotic manipulator as shown in Figures 9–13. The parameter vector of the adaptive fuzzy sliding controller introduced in (4.13) representing the points where the output membership functions attain their maximum values of 1 is assumed to have the initial value $\hat{\theta}^T = [-0.3003, -0.3721, -0.0887, 1.3105, 7.2130]$, which is updated during the course of the simulation by using the update rule (4.15.2) with the update constant $\lambda = 0.07$.

Figure 9 demonstrates the adaptive fuzzy sliding controller's ability to track the desired trajectory smoothly in the presence of structured and unstructured uncertainty. The parameter vector is seen to be continuously updated, since the unstructured uncertainty is time varying. The system is found to be robust to the unstructured uncertainty. The tracking performance is better than the conventional sliding controller's, where there is a small error that does not seem to converge after prolonged observation as seen in Figure 5. The input torque in Figure 11 is smooth and its magnitude

https://doi.org/10.1017/S0263574704001079 Published online by Cambridge University Press

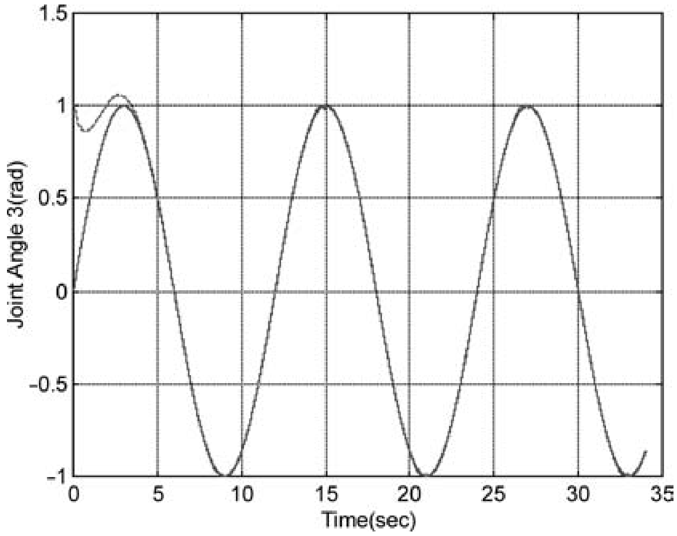

Fig. 9. Dynamic tracking of joint 3 by ASFC with payload change at time = 12 sec and unstructured uncertainty.

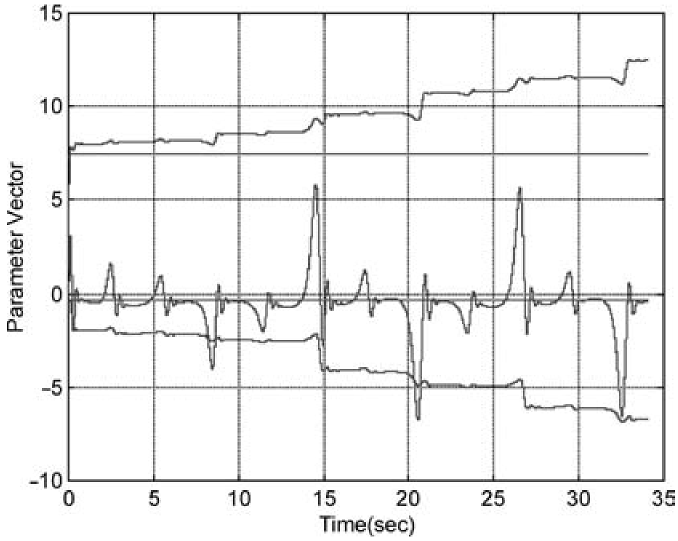

Fig. 10. Parameter vector update for dynamic tracking of joint 3 by ASFC with payload change and unstructured uncertainty.

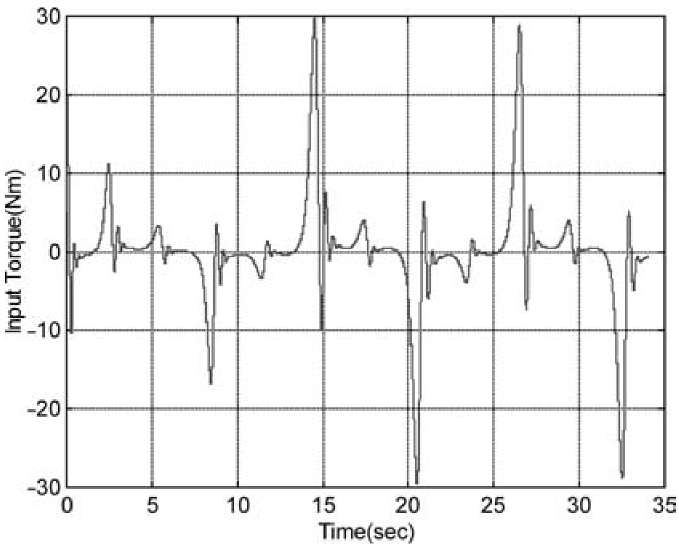

Fig. 11. Input torque for dynamic tracking of joint 3 by ASFC with payload change and unstructured uncertainty.

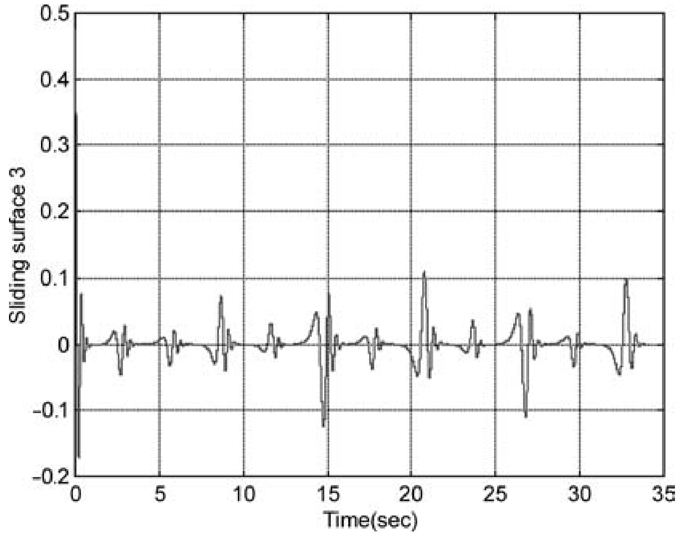

Fig. 12. Sliding surface for dynamic tracking of joint 3 by ASFC with payload change and unstructured uncertainty.

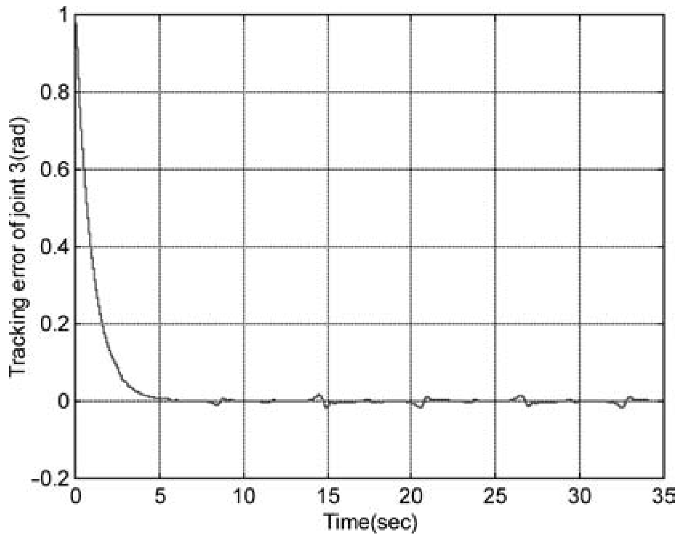

Fig. 13. Tracking error for dynamic tracking of joint 3 by ASFC with payload change and unstructured uncertainty.

lower than the conventional sliding controller's in Figure 6. It is also seen that the chattering is eliminated and a faster settling time is achieved.

During the simulation it is found that the system performs satisfactorily even when the value of k, representing the minimum approximation, is set to zero. This proves L.X. Wang's remark[8] that its value is very small and converges to zero when sufficient number of rules is used to control the input u. It is also observed, as mentioned in reference [6], that the order of parameter update is dependant on the value of S. For example, when the initial value of S is large, the center of membership function PL gets updated first, and P and Z get updated only when S is reduced. This may lead to slower convergence. However, this behavior can be modified to some extent by choosing a larger support for input membership functions Z, L and P, i.e., modifying the membership functions to include larger ranges of S.

6. CONCLUSIONS

In this paper, an adaptive fuzzy sliding control scheme is proposed to control a passive robotic manipulator. The motivation for the design of the adaptive fuzzy sliding controller is to eliminate the chattering and the requirement of pre-knowledge on bounds of error associated with the conventional sliding control. The stability and convergence of the adaptive fuzzy sliding controller is proven both theoretically and practically by simulations. A three-link passive manipulator model with two unactuated joints is derived to be used in the simulations. Simulation results demonstrate that the proposed system is robust against structured and unstructured uncertainties. Satisfactory tracking of the desired trajectory is achieved and the chattering associated with the conventional sliding control is eliminated. The contributions of this work include

a. Proposal of an adaptive fuzzy sliding control scheme to control a passive robotic manipulator. The proposed controller does not require an accurate model of the manipulator, which simplifies its implementation, and is capable of adapting itself to varying system parameters and disturbances.
b. Mathematical proof of the stability of the proposed control on a passive robotic manipulator by the Lyapunov method.
c. Practical validation of the stability and convergence of the overall system through simulation.

This work assumes that the passive joints are equipped with brakes. In many practical situations, however, the joints are not necessarily mounted with brakes. Controlling such passive manipulators is an interesting area of study. In the simulation, the actuator dynamics are not taken into consideration. Practically, actuator dynamics can be highly nonlinear and in itself constitute a major area of study. A possible direction of research includes controlling the overall system composed of an actuator with non-linear dynamics and a passive manipulator, with the controller deciding the electrical supply to the actuator.

References

1. H. Arai and S. Tachi, “Position control of a manipulator with passive joints using dynamic coupling”, *IEEE Transactions on Robotics and Automation* **7**(4), 528–534 (August, 1991).
2. M. Bergerman and Y. Xu, “Optimal control of manipulators with any number of passive joints”, *Journal of Robotic Systems* **15**(3), 115–129 (April, 1999).
3. R. Athinarayanan, B. Fallahi and R. Bansevicius, “Design and control of precise robots with active multi-degree of freedom kinematic pair”, *Mechanika* **N1**(27), 48–53, (2001).
4. P. Ro and C.-C. Yih, “Robust control of passive-jointed robot and experimental validation using sliding mode”, *Journal of Guidance, Control and Dynamics* **19**(5), 1039–1046 (September, 1996).
5. J.-J. Slotine and W. Li, *Applied Nonlinear Control* (Prentice-Hall, New York, 1991).
6. Y. Z. Guo and P.-Y. Woo, “An adaptive fuzzy sliding mode controller for robotic manipulators”, *IEEE Transactions on Systems, Man and Cybernetics (Part A)* **33**(2), 149–159 (March, 2003).
7. L. M. Peng and P.-Y. Woo, “Neural-fuzzy control system for robotic manipulators”, *IEEE Control Systems Magazine* **22**(1), 53–63 (February, 2002).
8. L. X. Wang, *A Course in Fuzzy Systems and Control* (Prentice-Hall, New York, 1997).
9. L.-X. Wang, “Stable adaptive fuzzy control of nonlinear systems”, *IEEE Transactions On Fuzzy Systems* **1**(2), 146–155 (May, 1993).
10. L. Pan and P.-Y. Woo, “PD manipulator controller with fuzzy adaptive gravity compensation”, *Journal of Robotic Systems* **17**(2), 93–106 (June, 1999).
11. T. Shaocheng, C. Tianyou and S. Cheng, “Adaptive fuzzy sliding mode control for nonlinear systems”, *Proceedings of the 1996 IEEE International Conference on Fuzzy Systems*, New Orleans, LA, USA (September, 1996) pp. 49–54.
12. T. Ross, *Fuzzy Logic with Engineering Applications* (McGraw-Hill, New York, 1995).
13. L.-X. Wang, “Fuzzy systems are universal approximators”, *Proceedings of the 1992 IEEE International Conference on Fuzzy Systems*, San Diego, Ca, USA (1992) pp. 1163–1170.
14. L.-X. Wang and J. M. Mendel, “Fuzzy basis functions, universal approximation, and orthogonal least squares learning”, *IEEE Transactions on Neural Networks* **3**(5), 807–814 (September, 1992).
15. F. C. Sun, Z. Q. Sun and G. Feng, “Design of adaptive fuzzy sliding mode controller for robot manipulators”, *Proceedings of the Fifth IEEE Int. Conf. on Fuzzy Systems* **1**, 62–67 (1996).
16. F. C. Sun, Z. Q. Sun and G. Feng, “An adaptive fuzzy controller based on sliding mode for robot manipulators”, *IEEE Trans. on Systems, Man and Cybernetics, Part B: Cybernetic* **29**, No. 4, 661–667 (August, 1999).
17. H. B. Xu, F. C. Sun and Z. Q. Sun, “The adaptive sliding mode control based on a fuzzy neural network for manipulators”, *Proceedings of the IEEE Int. Conf. on Systems, Man and Cybernetics* **3**, 1942–1946 (1996).
18. F. L. Lewis, C. T. Abdallah and D. M. Dawson, *Control of Robot Manipulators*, (Macmillan Publishing Company, New York, 1993).
19. S. Elangovan, “Adaptive Fuzzy Sliding Control for a Three-Link Passive Robotic Manipulator,” *MS Thesis* (Electrical Engineering Department, Northern Illinois University, Dekalb, Illinois, USA, 2003).