Robust attitude control of spacecraft using sliding mode control and productive networks

J. C. CHIOU [a] , M.-C. HWANG [a] , S. D. WUF [a] & J. Y. YANG [a]

[a] Department of Control Engineering , National Chiao-Tung University , Hsinchu, Taiwan, Republic of China Fax: E-mail:
Published online: 03 Apr 2007.

PLEASE SCROLL DOWN FOR ARTICLE

Robust attitude control of spacecraft using sliding mode control and productive networks

J. C. Chiou†, M.-C. Hwang†, S. D. Wu† and J. Y. Yang†

A new robust attitude control design of spacecraft is proposed by combining sliding mode control (SMC) and productive networks (PN). Essentially, the sliding mode control uses discontinuous control action to drive state trajectories toward a specific hyperplane in the state space, and to maintain the state trajectories sliding on the specific hyperplane. This principle provides a guideline to design a robust controller. Productive networks, which are a special type of artificial neural network, are then used to implement reaching and sliding conditions, and tackle the drawbacks of SMC such as chattering and high control gains. Attractive features of the proposed method include a systematic procedure of controller design, a reduction in chattering, robustness against model uncertainties and external disturbances. An inverted pendulum and a spacecraft attitude control problem are given to deomonstrate the effectiveness of the proposed method.

1. Introduction

Nonlinear control design has been studied for the last two decades (DeCarlo *et al.* 1988, Li and Slotine 1987, Sira-Ramirez and Dwyer 1987, Mamdani 1974, Narendra and Parthasatathy 1990). The aim of this research was to develop an analysis and design methodology that can possess robustness properties even in presence of parameter or structure uncertainties and disturbances. In the past, the application of nonlinear control methods was limited by the computational difficulty associated with nonlinear control design and analysis, such as the necessity of high speed and high precision. In recent years, however, advances in computer technology have greatly relieved this problem. Therefore, there is now considerable enthusiasm for the research and applications of nonlinear control methods.

The sliding mode control was originally developed based on the variable structure system (VSS) control (Utkin 1977, DeCarlo 1988). It is known as a robust control to deal with confined uncertainties due to the modelling error and disturbances. The sliding mode control has been applied to many practical systems, such

as flight control systems, robotics manipulator (Slotine and Sestry 1983, Bailey and Arapostathis 1987), and spacecraft attitude control (Singh and Iyer, 1989, Elmali and Olgac 1992).

The principal objective of sliding mode control is to force the trajectory of the system to a desired surface, which is known as the sliding surface. The control laws are designed so that the system state trajectory will reach the sliding surface. Once the state trajectory is on the sliding surface, it is well-known that the system is insensitive to certain parameter variations and external disturbances.

Although sliding mode control theoretically shows excellent robustness properties in dealing with parameter uncertainty and disturbances, classical sliding mode control however, presents several undesired drawbacks that limit its practical implementation. First, the most serious problem is the phenomenon of chattering, which is caused by practical switching limits, i.e. the components cannot operate at an infinite frequency. Since the method involves extremely high control activity, it may thus excite high frequency dynamics that are neglected in the course of modelling. Secondly, the sliding mode control laws are always chosen to be large enough in order to suppress those uncertainties caused by the parameter variations and external disturbances. Thus, it always makes the sliding mode control (SMC) hard to physically implement. Finally, the control design procedure and the online

calculation are tedious and complicated, especially when the dimension of system is increased or the structure of the system is highly nonlinear.

In order to overcome these drawbacks, we introduce productive networks into the sliding mode controller design. Productive networks (PN) are artificial neural networks which employ fuzzy logic inference mechanisms. Problems requiring inferencing with Boolean logic have been implemented in perceptrons or feedforward networks (Bulsari and Saxen 1991 a). However, feedforward neural networks with sigmoidal activation functions cannot accurately evaluate fuzzy logic expressions using the T-norm. Productive networks proposed by Bulsari (1992) and Bulsari and Saxen (1991 b) can accurately perform elementary fuzzy logic operations. The productive network is much simpler than a feedforward neural network since the architecture of the PN was desired to avoid extensive training; in fact, it requires no training. It was shown that fuzzy logic inferencing could be performed in productive networks by manually setting the offsets. Thus, it is reliable since the function of each of the nodes in the network is known and understood.

The objective of the present paper is to report a new robust control algorithm combined with sliding mode control and productive networks. The proposed method is aimed at effectively reducing the phenomenon of chattering and the high control gain effect due to nonlinear compensation. It can easily be implemented without a tedious design procedure and calculation.

This paper is organized as follows. Section 2 presents a review of the sliding mode control for a general nonlinear system. An introduction to productive networks is given in § 3, including the derivation of productive networks and how to perform the three basic logical operations, AND, OR and NOT. Section 4 presents a new robust control method based on the sliding mode control and productive networks. In this section, we first apply the SMC principle to the dynamic systems so that the robustness of uncertainties and disturbances will be obtained once the systems are on the sliding surface. Secondly, by defining the increase or decrease of control inputs using productive networks, a control law can be determined. Section 5 presents a single-input single-output (SISO) inverted pendulum problem and a multi-input multi-output (MIMO) spacecraft attitude control problem that illustrate the performance of the present control algorithm. Section 6 gives the final conclusions.

2. Sliding mode control

Consider a control system of the form

$$\dot{x}(t) = f(t, x, u),$$ (1)

where the state vector $x(t) \in R^n$, and the control vector $u(t) \in R^m$. In particular, for each coordinate of the control vector u, we associate a collection of real-valued continuous functions $\{u_i^+(t, x),\ u_i^-(t, x),\ s_i(x)\}$, $i = 1, 2, \ldots, m$, such that the control u is given by

$$u_i(t, x) = \begin{cases} u_i^+(t, x) & \text{for } s_i(x) > 0, \\ u_i^-(t, x) & \text{for } s_i(x) < 0, \end{cases} \quad i = 1, 2, \ldots, m,$$ (2)

where

$$s_i(x) = 0, \qquad i = 1, 2, \ldots, m$$ (3)

is the ith switching surface.

Since a sliding mode is a motion constrained on a sliding surface, the sliding surface is designed such that the system response restricted to $s(x) = 0$ has the desired behaviour, such as stability and tracking. With this in mind, let us assume that the output of (1) is of the form $y(t) = h(x(t))$ and that the problem at hand now is to find a feedback control law to force the output to track a desired signal $y_d(t)$. According to the previous discussion, the synthesis problem involves two steps: first, the selection of a switching function s whose zeros comprise a surface on which the restriction of the dynamics has the tendency to drive the error $y(t) - y_d(t)$ asymptotically to zero; secondly, the design of a switching control (corresponding to the choice of s) that makes the state trajectory reach and slide on the sliding surface. A key issue in SMC design is decided by the existence of sliding mode on the switching surface. Thus, designing the proper switching surface is the complementary key problem in SMC design.

Although general nonlinear sliding surfaces are possible, linear ones are more prevalent in SMC design. After the switching surface has been designed, the next important aspect of SMC design is guaranteeing the existence of a sliding mode. The Lyapunov stability criterion is then used to derive a sliding condition so that the sliding surface can be reached within a finite time. A suitable Lyapunov function, v, can be chosen as

$$v\tfrac{1}{2}s^T s \geq 0.$$ (4)

If the following condition is satisfied with the designed control input, a sliding mode does exist:

$$\dot{v} = s^T \dot{s} < 0, \qquad \text{for all time.}$$ (5)

In Li and Slotine (1987), the MIMO sliding condition is given by

$$s_i \dot{s}_i = -\eta |s_i| \qquad \eta > 0 \quad i = 1, 2, \ldots, m$$ (6)

which guarantees the transient time from the initial state to the sliding manifold. Equation (6) is more restrictive

than (5) which only requires

$$\dot{v} = s^{T}\dot{s} = \sum_{i=1}^{m} s_i \dot{s}_i < 0. \qquad (7)$$

Note that (6) is a special case where

$$s_i \dot{s}_i < 0 \qquad \text{for } i = 1, 2, \ldots, m. \qquad (8)$$

That is, all sliding subconditions in (6) with respect to sliding variables $s_i(x)$. $i = 1, 2, \ldots, m$, are required to be strictly satisfied at any time instant.

From the above derivations, sliding mode control algorithms have been successfully developed. However, there exist the following drawbacks. First, the sliding mode controllers inherently contain a high frequency behaviour, i.e. chattering, which may excite high frequency dynamics neglected in the course of modelling. Secondly, the magnitudes of the SMC control laws are always chosen to be large enough in order to suppress those uncertainties caused by the parameter variations and external disturbances. Thirdly, the control design procedure of the system and the online calculation are tedious and complicated, especially, when the dimension of the system is increased or the behaviour of system is highly nonlinear.

3. Productive networks

Since the aforementioned SMC algorithm suffers various drawbacks in computer implementation, these have motivated us to look for an alternative procedure that overcomes these difficulties. This alternative uses the productive networks to implement reaching and sliding conditions of SMC, and thus reduces the above-mentioned drawbacks. In productive networks, if we assume all logical operations can be represented as the combinations of AND ($\wedge$), OR ($\vee$) and NOT ($\sim$) operations, and the truth value of events A, B and C are represented by $t(A)$, $t(B)$ and $t(C)$ respectively, then the truth values of AND, OR and NOT can be carried out as follows:

$$t(A \wedge B) = t(A)t(B) \qquad (9)$$

$$t(A \vee B) = t(A) + t(B) - t(A)t(B) \qquad (10)$$

$$t(\sim A) = 1 - t(A). \qquad (11)$$

The OR operation can be modified to a suitable form by

$$A \vee B = \sim(\sim A \wedge \sim B) \qquad (12)$$

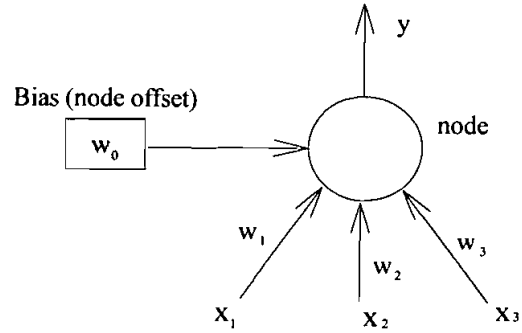$$t(A \vee B) = 1 - (1 - t(A))(1 - t(B)), \qquad (13)$$

Figure 1. A node in a productive network.

which is equivalent to the operation shown in (10). Similarly, we obtain the following operations:

$$\left. \begin{array}{l} t(A \wedge B \wedge C) = t(A)t(B)t(C) \\ t(A \vee B \vee C) = 1 - (1 - t(A))(1 - t(B))(1 - t(C)) \end{array} \right\} \qquad (14)$$

for three operands. These operations contribute toward building the productive networks.

A productive network, as defined here, collects an offset product of inputs and a further offset by a bias. This can be illustrated by activation of the node as shown in Fig. 1, which can be written as

$$a = \omega_0 - |(\omega_1 - x_1)(\omega_2 - x_2)(\omega_3 - x_3)| \qquad (15)$$

where the weights $(\omega_1, \omega_2, \omega_3)$ and bias(ω_0) are zero or one, x_i, $i = 1, 2, 3$ are the inputs of the node. The output of the node is the absolute value of the activation, a, i.e.

$$y = |a|. \qquad (16)$$

In general, one can express the activation in the following abbreviated form:

$$a = \omega_0 - \left| \prod_j (\omega_j - x_j)^{[1 + (1/2)\omega_j(1 - \omega_j)]} \right|. \qquad (17)$$

The productive network has multiple nodes with one or more inputs which should be positive numbers equal to or less than 1. Each of the nodes has a bias (alternatively called the offset of the node). Obviously, the output is decided by the number between 0 and 1. Productive networks are so named because of the multiplication of inputs at each node.

To show that the productive networks can be organized to compute any complicated logical operation, it is sufficient to show that the three basic logical operations can be implemented in this framework.

The simplest operation, NOT, can be represented by the networks shown in Fig. 2.

The AND operation as given in Fig. 3 is also implemented in a facile manner in this framework. Its
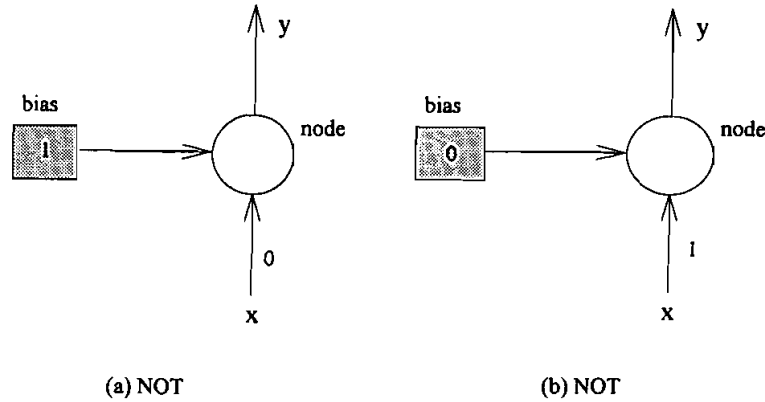
Figure 2. Inverting an input in a productive network (NOT): (a) has the same function as (b).
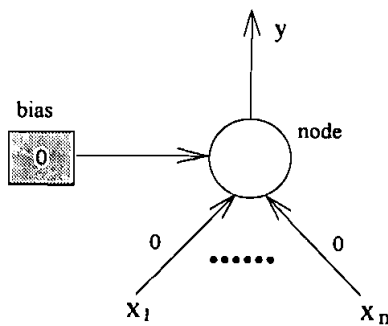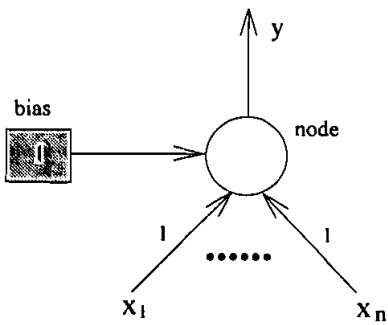

Figure 3. AND.

Figure 4. OR.

value of y is the product of the truth values of its arguments; i.e. both the weights and the bias have values of zero.

On the other hand, the OR operation requires all weights and bias equal to 1. Figure 4 illustrates an OR operation with n inputs in a productive network.

4. Proposed nonlinear control algorithm

A MIMO nonlinear control system is taken as:

$$\left.\begin{aligned}
\dot{x}_1 &= f_1(x_1, x_2, t) \\
\dot{x}_2 &= f_2(x_1, x_2, t) + G(x_1, x_2, t)u \\
y &= h(x_1)
\end{aligned}\right\} \qquad (18)$$

where $x_1(t) \in R^m$, $x_2(t) \in R^{n-m}$ are the plant state vectors, and $u(t) \in R^m$, $y \in R^m$ are system input and output vectors respectively. The desired output is $y_d(t) \in R^m$.

As described in § 2, the first step of SMC design entails sliding surfaces so that the system dynamics follow a desired performance once the system state trajectory is on the sliding surfaces. It can easily be accomplished by choosing

$$s = \dot{e} + ce = 0, \quad c = \begin{bmatrix} c_1 & 0 & \cdots & 0 \\ 0 & c_2 & \cdots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ 0 & \cdots & 0 & c_m \end{bmatrix}, \quad c_i > 0,$$

$$i = 1, \ldots, m \quad (19)$$

where $e = y - y_d$ is the tracking error and $c \in R^{m \times m}$ is a diagonal positive matrix. Once the state trajectory is on the sliding surface, the desired output will be satisfied. Next, the objective of the sliding mode control is to design a control law such that the sliding condition is satisfied i.e. equation (7) is satisfied. In this regard, we differentiate (19) with respect to time once and introduce (1) to obtain

$$\dot{s} = \ddot{e} + c\dot{e}$$

$$= \frac{d}{dt}\left[\left(\frac{\partial h}{\partial x_1}\right) \cdot f_1(x_1, x_2, t)\right] - \cdots$$

$$= \left(\frac{\partial h}{\partial x_1}\right) \cdot \left(\frac{\partial f_1}{\partial x_2}\right) \cdot G(x_1, x_2, t) \cdot u + \cdots$$

$$\equiv B(x_1, x_2, t)u + [\text{O.T.}] \qquad (20)$$

where

$$B(x_1, x_2, t) = \left(\frac{\partial h}{\partial x_1}\right) \cdot \left(\frac{\partial f_1}{\partial x_2}\right) \cdot G(x_1, x_2, t)$$

and [O.T] denotes the other terms. In the present development, equation (8) is used to perform a reaching and sliding condition because with equation (8) it is easier

to obtain the control rules than with equation (7), i.e. the condition

$$s_i \dot{s}_i = s_i \left\{ \sum_{j=1}^{m} B_{ij} u_j + [\text{O.T}]_i \right\} < 0 \qquad i = 1, 2, \ldots, m,$$

(21)

should be satisfied for each sliding surface. Now we can discuss the causal nexus between u_j and s_i, $\dot{s}_i$ to construct control laws.

In equation (20), it is assumed that $\dot{s}_i$ increases as u_j increases and vice versa. Assuming B_{ij} is greater than zero, equation (21) provides the information that if s_i is greater than zero and $\dot{s}_i$ is greater than zero, then decreasing u_j will result in decreasing $s_i \dot{s}_i$. If s_i is less than zero and $\dot{s}_i$ is less than zero, then increasing u_j will result in decreasing $s_i \dot{s}_i$. The inputs of the proposed productive networks controller with respect to the ith sliding surface are s_i and $\dot{s}_i$. The output in this development is Δu_{ij} which is the change of u_j with respect to the ith sliding surface. In implementation, $\dot{s}_i$ is approximated with $(s^k - s^{k-1})/T$ where T is the sampling time. If we define the truth values of $s_i > 0$, $\dot{s}_i > 0$ as in Fig. 5, then the truth value of (increasing ΔU_{ij}) is equal to $t[(s_i < 0) \cap (\dot{s}_i < 0)]$ and the truth value of (decreasing ΔU_{ij}) is equal to $t[(s_i > 0) \cap (\dot{s}_i > 0)]$. It is natural to choose the difference between t(increasing ΔU_{ij}) and t(decreasing ΔU_{ij}) as actual output ΔU_{ij}. In mathematical descriptions, we have

$$t(\text{decreasing } \Delta U_{ij}) = t[(s_i > 0) \cap (\dot{s}_i > 0)]$$

$$= t[(s_i > 0)] \cdot t[(\dot{s}_i > 0)] \qquad (22)$$

$$t(\text{increasing } \Delta U_{ij}) = t[(s_i < 0) \cap (\dot{s}_i < 0)]$$

$$= t[\sim(s_i > 0)] \cdot t[\sim(\dot{s}_i > 0)] \qquad (23)$$

$$\Delta U_{ij} = t(\text{increasing } \Delta U_{ij}) - t(\text{decreasing } \Delta U_{ij}). \qquad (24)$$

If B_{ij} is less than zero, according to the above discussion, the sign of ΔU_{ij} should be changed. Then we come to the conclusion that

$$\Delta U_{ij} = \text{sign} (B_{ij}) \cdot [t(\text{increasing } \Delta U_{ij})$$
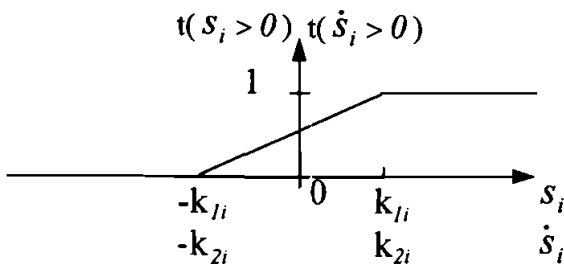
$$- t(\text{decreasing } \Delta U_{ij})]. \qquad (25)$$

$$t(S_i > 0) \quad t(\dot{S}_i > 0)$$

Figure 5. Truth value of $A > 0$.

The final value of ΔU_j is obtained by the total summation of ΔU_{ij} with respect to each sliding surface, i.e.

$$\Delta U_j = \sum_{i=1}^{m} \Delta U_{ij} \qquad j = 1, 2, \ldots, m \qquad (26)$$

and

$$\Delta u_j = k_{3j} \cdot \Delta U_j \qquad j = 1, 2, \ldots, m \qquad (27)$$

where k_js are prescribed given constants.

A systematic procedure is now available for robust control of nonlinear dynamic systems. The proposed nonlinear control algorithm is summarized as follows.

Step 1. Design sliding surfaces, s, which are given in (19), such that the system response is restricted to $s = 0$, which guarantees a desired behaviour such as stability or tracking. Initially, $s^0 = 0$ and $u^0 = 0$ where upper index 0 denotes the initial time step.

Step 2. At a time step t_p, compute s^p, and $\dot{s}^p$.

Step 3. Evaluate these truth values of $s^p > 0$, $\dot{s}^p > 0$ with two factor vectors k_1, k_2 respectively.

Step 4. Use equations (25) and (26) to compute the value of change of the ith control input with respect to each sliding surface.

Step 5. To evaluate the output of the controller, equation (27) is used and the control action is updated by

$$u^k = u^{k-1} + \Delta u.$$

Step 6. Add control action in Step 5 to the nonlinear dynamic system (1).

Step 7. Return to Step 2.

5. Numerical examples

5.1. SISO Systems

An inverted pendulum system shown in Fig. 6 is used as a SISO example. A movable pole is joined to a vehicle by a pivot. The pole, which serves as an inverted pendulum, can be kept standing by moving the vehicle appropriately. Let ω_0, x_1 and y_1 denote the coordinates of the pole, and θ is the angular position of the pole deviated from the equilibrium position. The pendulum has mass m and length $2L$, and the moment of inertia is $I = mL^2/3$. Let x_2 and y_2 denote the coordinates of the vehicle with mass M. The control force, u, is applied to the vehicle. The equation of motion is:

$$\ddot{\theta} = \frac{\left[mgL \sin \theta - \dfrac{m^2 L^2}{M + m} \cos \theta \sin \theta \cdot \dot{\theta}^2 - \dfrac{mL \cos \theta}{M + m} \cdot u \right]}{\left[I + mL^2 - \dfrac{m^2 L^2}{M + m} \cos^2 \theta \right]}.$$
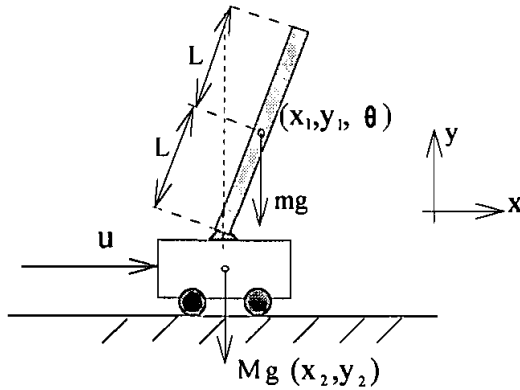
Figure 6. Diagram of an inverted pendulum.

Since the goal of an inverted pendulum is to keep the pole standing, the actual output of the system and the desired output, y_d, are chosen to be

$$y(t) = \theta,$$

$$y_d(t) = 0.$$

The sliding surface, s, is chosen to be

$$s = \dot{\theta} + 10\theta = 0.$$

These two factors for $t(s > 0)$ and $t(\dot{s} > 0)$ are k_1 and k_2 respectively.

Two pendulums with different lengths and weights are used for simulation. Their specifications are as follows:

pendulum No. 1 : 3·5 g, 15 cm,

pendulum No. 2 : 50 g, 50 cm.

The mass of the vehicle is 1 kg, which is the same for the two different pendulums. The resulting control was simulated for the following parameter values:

$$c = 10, \qquad k_1 = 10,$$

$$k_2 = 300, \qquad k_3 = 40.$$

Time responses of the angular position for both pendulums, from the given initial condition of 60° from the equilibrium position, are shown in Fig. 7. The two response are almost identical in spite of the difference in system parameters, i.e. the system uncertainties. Note that there is no overshooting observed for both cases. The simulation results in Fig. 8 show the time responses of s.

In order to demonstrate the external disturbance rejection capability of the proposed method, a sinusoidal external force is applied on the vehicle in the horizontal direction. It is chosen as
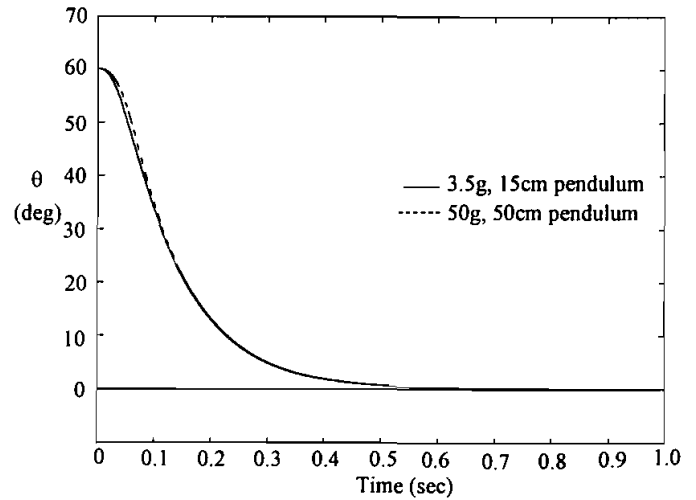
$$T_d = 10 \sin (200t).$$

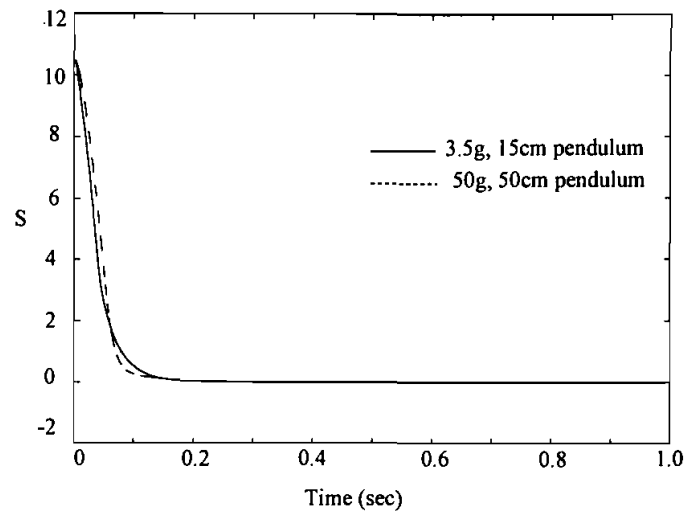Figure 7. Time responses of $\theta(t)$ of two different pendulums.

Figure 8. Time responses of $s(t)$ of two different pendulums.

Figures 9–11 show the simulation results. Pendulum No. 2 is used for both cases in order to make a comparison. The angular position response shown in Fig. 9 is almost unaffected by the external disturbance, although the time response of s as shown in Fig. 10 exhibits a severe oscillation along the sliding surface.

Chattering is a serious drawback of sliding mode control and has been an important problem in the field of SMC. Figure 12 shows the state trajectory. It is clear that the phenomenon of chattering is greatly reduced.

In order to show how the proposed method performs better than that of SMC combined with fuzzy logic control (FSMC), we compare it with the method proposed by Hwang and Lin (1992). The system parameters are the same. Figures 13 and 14 show the simulation results. FSMC uses a 'two-degree nested control window' structure, in which the same set of control rules is used recursively and different scaling
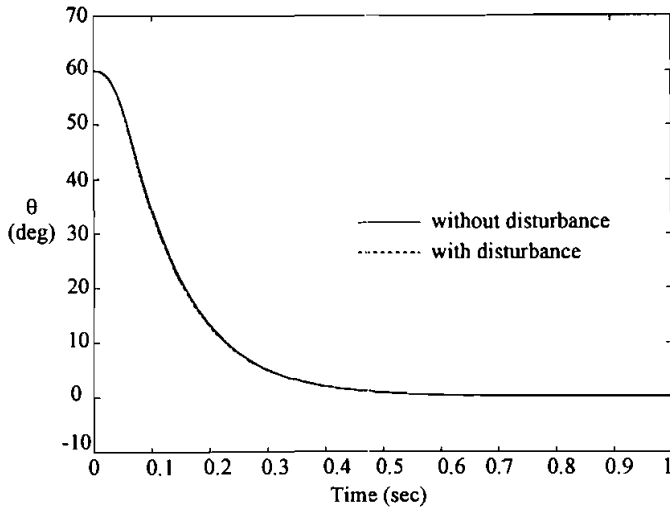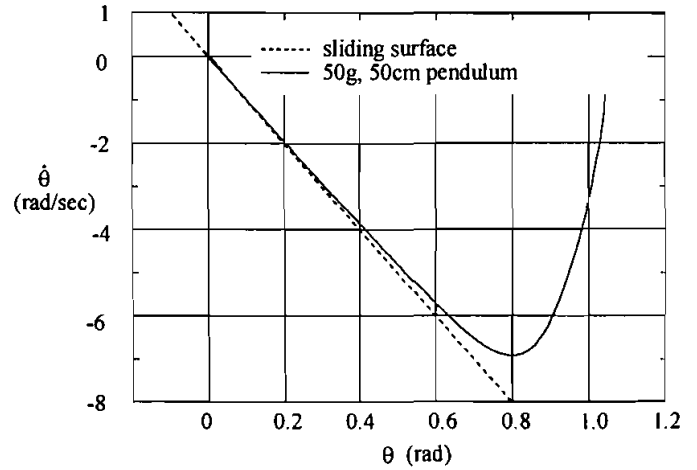
Figure 9. Time response of θ(*t*) with disturbance.
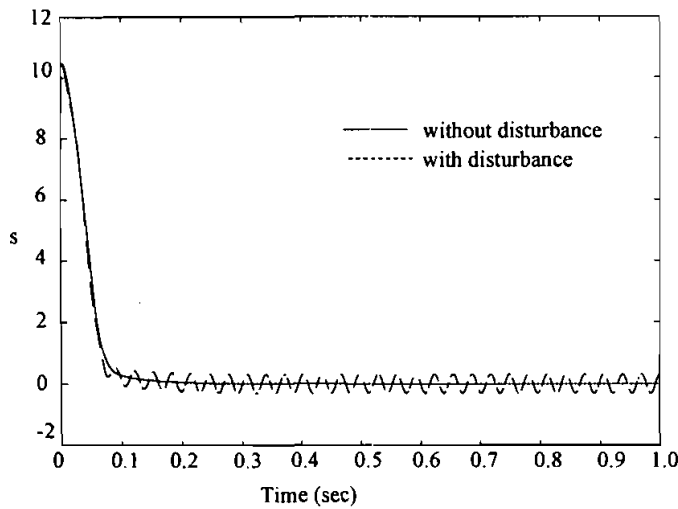

Figure 12. State trajectory of pendulum No. 2.

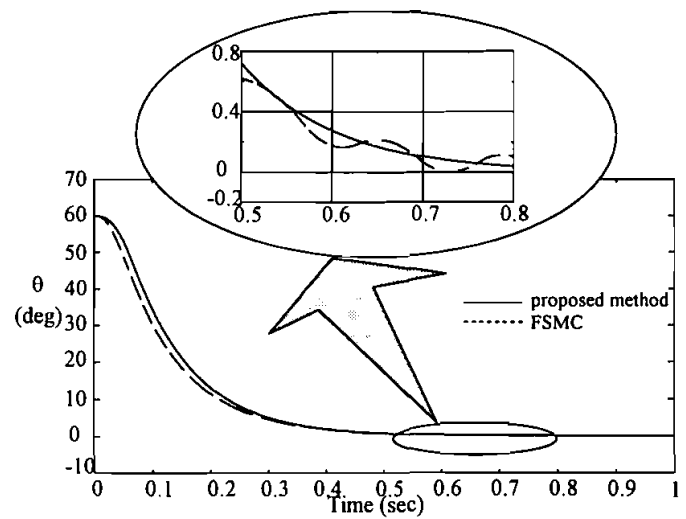**Figure 10.** Time response of *s*(*t*) with disturbance.

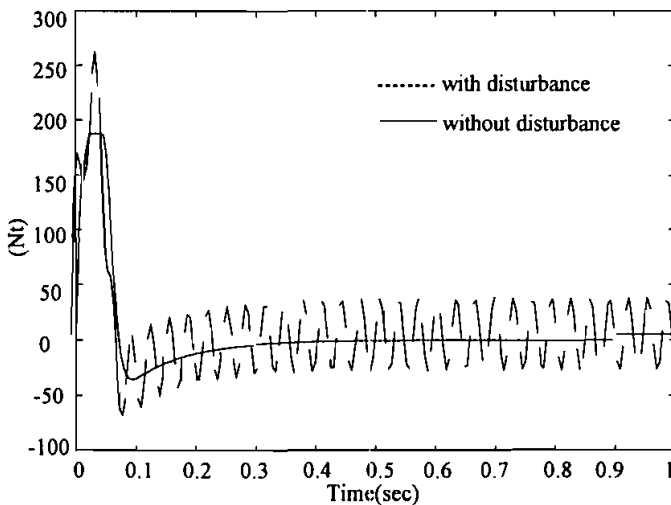**Figure 13.** Time responses of θ(*t*) of two different methods.

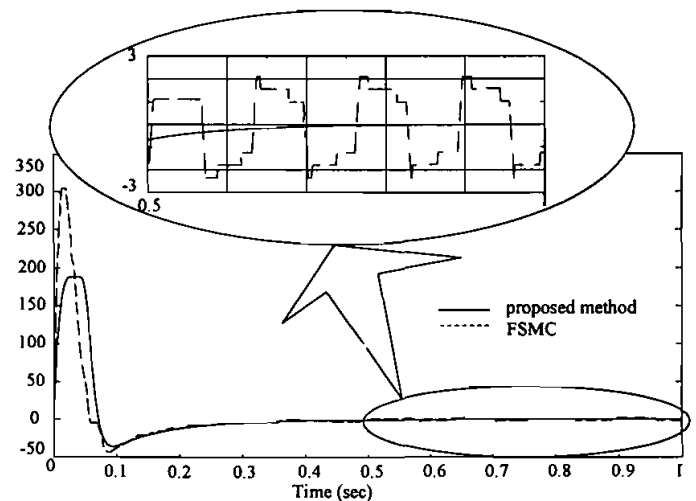**Figure 11.** Time response of *u* with disturbance.

Figure 14. Control inputs of two different methods.

factors are used for each degree of nest. The scaling factors for the first degree of nest are chosen as the same as the proposed method: $k_1 = 10$, $k_2 = 300$ and $k_3 = 40$; the scaling factors for the second degree of nest are $k_1 = 0.5$, $k_2 = 20$ and $k_3 = 4$. It is more tedious than the proposed method which only needs one set of the scaling factors. In order to avoid the time consumption on defuzzification of FSMC, it needs a look-up table which directly relates the controller inputs s and $\dot{s}$ with the controller output change Δu. This will cause control input to oscillate as shown in Fig. 14. On the other hand, the control input of the proposed method yields no oscillatory effect at the closing stage of the simulation.

5.2. MIMO systems

The attitude control problem of a spacecraft is considered to be an example of MIMO nonlinear systems. Typically, dynamics and associated characteristics are taken from Singh and Iyer (1989) and Elmali and Olgac (1992). Assume the spacecraft is on a circular orbit in an inverse square gravitational field, and the attitude of the space vehicle has no effect on the orbit. The equations of motion of the system are

$$\dot{x}_1 = f_1(x)$$

$$x_2 = f_2(x) + Gu + T_d$$

where $x = [x_1 \ x_2] = [\theta^T \ \omega^T]$ is the state vector, and

$$f_1 = \begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \end{bmatrix} = \begin{bmatrix} \cos\theta_3 \sec\theta_2\omega_2 - \sin\theta_3 \sec\theta_2\omega_3 - \omega_0 \\ \sin\theta_3\omega_2 + \cos\theta_3\omega_3 \\ \omega_1 + \tan\theta_2(\sin\theta_3\omega_3 - \cos\theta_3\omega_2) \end{bmatrix}$$

$$f_2 = \begin{bmatrix} \dfrac{I_2 - I_3}{I_1}(\omega_2\omega_3 - 3\omega_0^2\xi_2\xi_3) \\ \dfrac{I_3 - I_1}{I_2}(\omega_3\omega_1 - 3\omega_0^2\xi_3\xi_1) \\ \dfrac{I_1 - I_2}{I_3}(\omega_1\omega_2 - 3\omega_0^2\xi_1\xi_2) \end{bmatrix}$$

$$G = \begin{bmatrix} \dfrac{1}{I_1} & 0 & 0 \\ 0 & \dfrac{1}{I_2} & 0 \\ 0 & 0 & \dfrac{1}{I_3} \end{bmatrix}, \quad T_d = \begin{bmatrix} \dfrac{T_{d1}}{I_1} \\ \dfrac{T_{d2}}{I_2} \\ \dfrac{T_{d3}}{I_3} \end{bmatrix}$$

$$\xi_c = \begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{bmatrix} = \begin{bmatrix} -\sin\theta_1 \cos\theta_2 \\ \cos\theta_1 \sin\theta_3 + \sin\theta_1 \sin\theta_2 \cos\theta_3 \\ \cos\theta_1 \cos\theta_3 - \sin\theta_1 \sin\theta_2 \sin\theta_3 \end{bmatrix}.$$

u is the control torque vector and T_d is the external disturbance vector. $\theta = [\theta_1 \ \theta_2 \ \theta_3]^T$ denotes the roll, pitch and yaw angles about the body fixed axes respectively. $\omega[\omega_1 \ \omega_2 \ \omega_2]^T$ is the angular velocity with respect to the inertial frame, and ω_0, a constant, denotes the orbital angular velocity of the mass centre of the spacecraft. $(I_1 \ I_2 \ I_3)$ are the moments of inertia about the body fixed axes. The objective of this control strategy is to track $y_d = [\theta_{1d} \ \theta_{2d} \ \theta_{3d}]^T$ in the presence of modelling uncertainties and external disturbances. From Singh and Iyer (1989 and Elmali and Olgac (1992) we get

$$y = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \theta = h(\theta) = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix}.$$

$$y_d = [1 - e^{-0.353t}(\sin 0.353t + \cos 0.353t)]r$$

where $r = [180°, 45°, 90°]^T$.

The sliding surface is chosen to be

$$s = \dot{e} + ce$$

$$= (\dot{\theta} - \dot{y}_d) + \begin{bmatrix} c_1 & 0 & 0 \\ 0 & c_2 & 0 \\ 0 & 0 & c_3 \end{bmatrix}(\theta - y_d)$$

$$= 0.$$

The two factors required for $t(s > 0)$ and $t(\dot{s} > 0)$ are

$$k_1 = [k_{11} \ k_{12} \ k_{13}]^T$$

$$k_2 = [k_{21} \ k_{22} \ k_{23}]^T$$

respectively. Note that B cannot be exactly obtained because the modelling uncertainties appear in the moments of inertia, $\{I_i, i = 1, 2, 3\}$. Since we know the values of $\{I_i, i = 1, 2, 3\}$ are all positive, it means that the sign of B_{ij} remains unchanged, i.e.

$$\text{sign}(B_{ij}) = \text{sign}\left(\frac{\partial f_{1i}}{\partial \omega_j}\right)$$

$$\Delta U = \begin{bmatrix} 0 & \cos\theta_3 \sec\theta_2 & -\sin\theta_3 \sec\theta_2 \\ 0 & \sin\theta_3 & \cos\theta_3 \\ 1 & -\tan\theta_2 \cos\theta_3 & \tan\theta_2 \sin\theta_3 \end{bmatrix}$$

$$\cdot \begin{bmatrix} t[(s_1 < 0) \cap (\dot{s}_1 < 0)] - t[(s_1 > 0) \cap (\dot{s}_1 > 0)] \\ t[(s_2 < 0) \cap (\dot{s}_2 < 0)] - t[(s_2 > 0) \cap (\dot{s}_2 > 0)] \\ t[(s_3 < 0) \cap (\dot{s}_3 < 0)] - t[(s_3 > 0) \cap (\dot{s}_3 > 0)] \end{bmatrix}$$

For numerical simulation, the following values of the parameters of the spacecraft are assumed. The orbital angular speed, ω_0, is 7.29×10^{-5} rad s². The modelling

uncertainties appear in the form of $I_i + \Delta I_i$, $i = 1, 2, 3$, where $I_1 = 874.6$, $I_2 = 888.2$ and $I_3 = 97.6 \text{ kg m}^2$. The variations in the inertia are taken as

$$\begin{bmatrix} \Delta I_1 \\ \Delta I_2 \\ \Delta I_3 \end{bmatrix} = 0.5[1 + \sin{(0.1t)}] \cdot \begin{bmatrix} v_1 I_1 \\ v_2 I_2 \\ v_3 I_3 \end{bmatrix}$$

where $v_1 = 0.1$, $v_2 = 0.2$, $v_3 = 0.3$. The external disturbances are chosen as

$$T_{d1} = 40 \sin{(2\pi t)}$$

$$T_{d2} = 40 \cos{(2\pi t)}$$

$$T_{d3} = 20 \sin{(2\pi t)}.$$

The initial orientation of the spacecraft is taken as $\theta_0 = [-3 \ 3 \ 5]^T$ degrees. Simulation of the closed-loop system with control design parameters is chosen as follows: $c = \text{diag}(1, 1, 1)$, $k_1 = [0.1 \ 0.1 \ 0.1]^T$, $k_2 = [0.1 \ 0.1 \ 0.1]^T$ and $k_3 = [1 \ 1 \ 1]^T$. The responses of spacecraft are shown in Figs 15 and 16. The transition of the error functions with tracking conditions are stabilized in nearly 20 s, as shown in Figs 19 and 20, whereas, the time differentiation of sliding surfaces, $\dot{s}$, experienced high frequency chattering behaviour as illustrated in Figs 17 and 18. In order to improve the behaviour, we further choose saturation function B_{ij} as

$$\text{sat}(B_{ij}) = \begin{cases} 1 & \text{if } B_{ij} > \text{Bound}_{ij} \\ B_{ij} & \text{if } B_{ij} < \text{Bound}_{ij} \\ -1 & \text{if } B_{ij} < -\text{Bound}_{ij} \end{cases}$$

where

$$\text{Bound}_{11} = |1|$$

$$\text{Bound}_{12} = |\cos{\theta_3} \sec{\theta_2}|$$

$$\text{Bound}_{13} = |\sin{\theta_3} \sec{\theta_2}|$$

$$\text{Bound}_{21} = |1|$$

$$\text{Bound}_{22} = |\sin{\theta_3}|$$

$$\text{Bound}_{23} = |\cos{\theta_3}|$$

$$\text{Bound}_{31} = |1|$$

$$\text{Bound}_{32} = |\tan{\theta_2} \cos{\theta_3}|$$

$$\text{Bound}_{33} = |\tan{\theta_2} \sin{\theta_3}|.$$

Simulation results on the attitude tracking manoeuvres are shown in Figs 21 and 22. In accordance with the sliding-mode theory, Figs 23 and 24 illustrate that the sliding vectors s and $\dot{s}$ are driven to the sliding surface and then stayed oscillatory because of the desired tracking variables y_d. All of the control torques applied in Figs 19, 20, 25 and 26 show that the present scheme requires lesser torques in comparison with both Singh
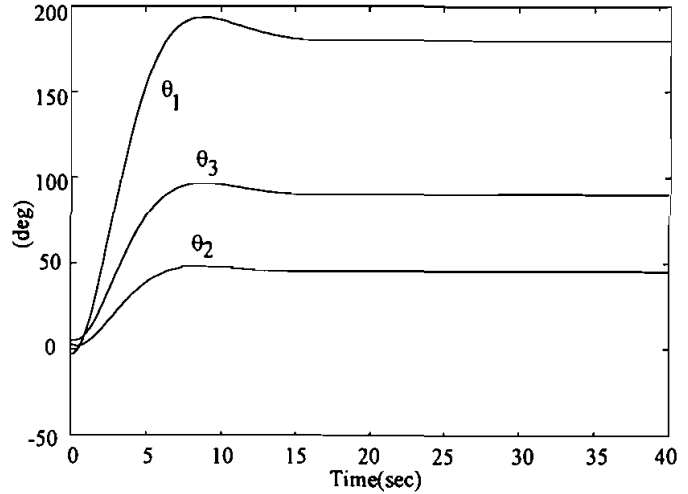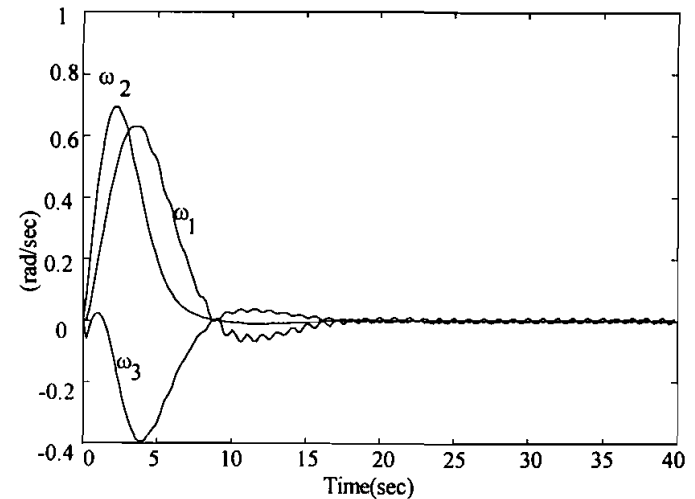

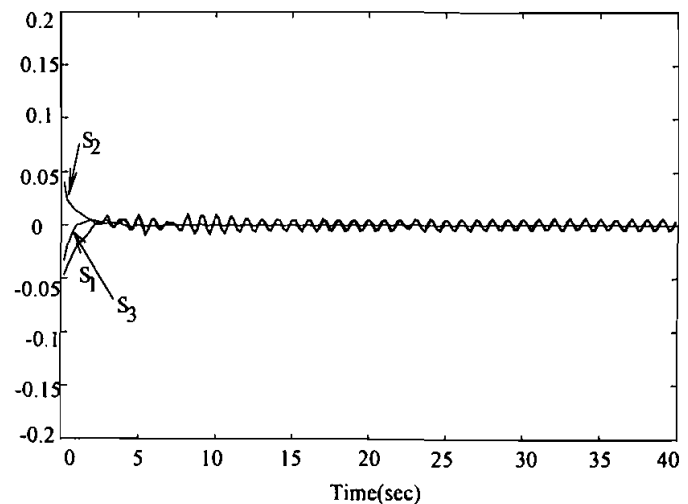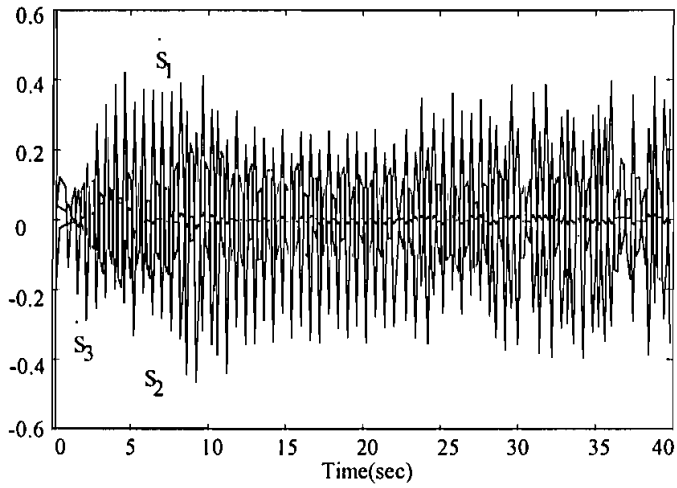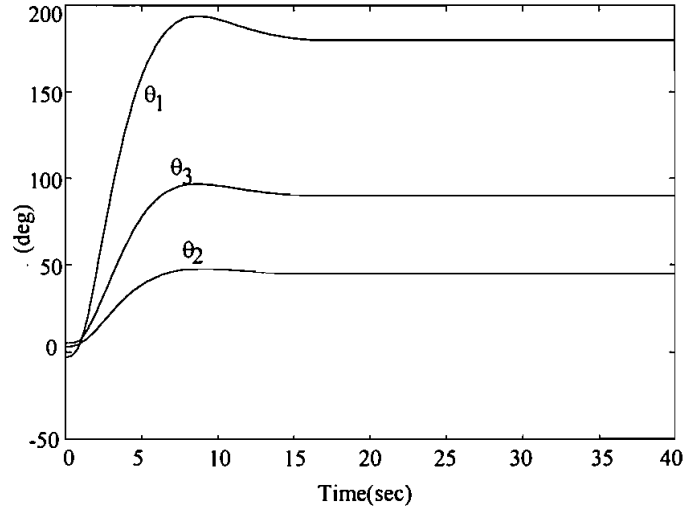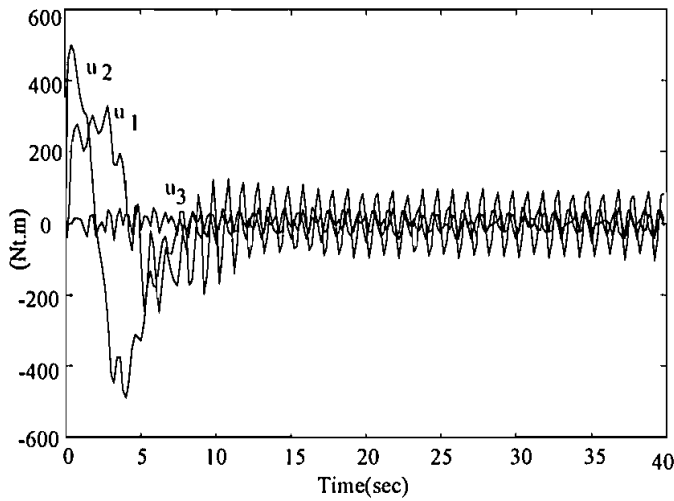Figure 15. Time responses of θ.

Figure 16. Time responses of ω.

Figure 17. Time responses of S.

J. C. Chiou et al.

Figure 18.   Time responses of *ṡ*.

Figure 21.   Time responses of θ.
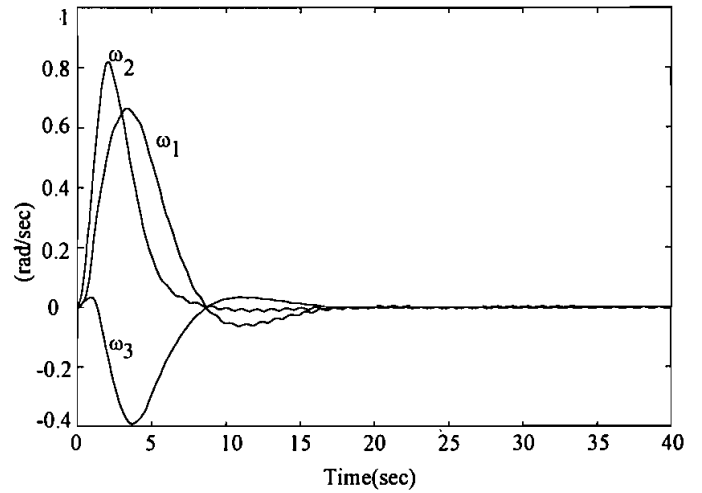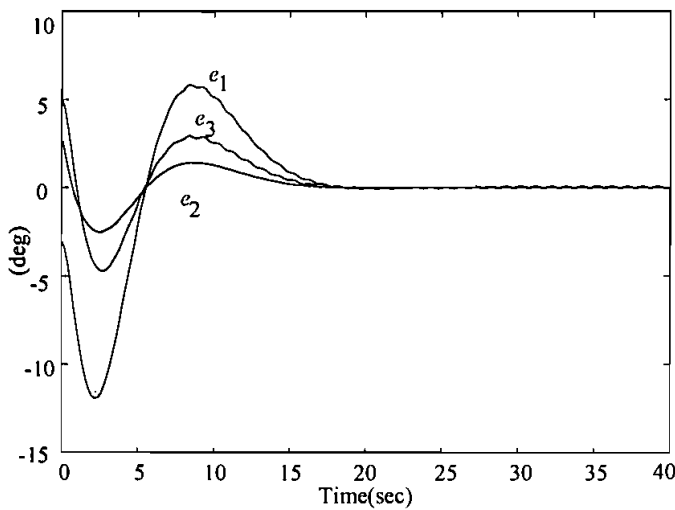

Figure 19.   Time responses of *u*.

Figure 22.   Time responses of ω.
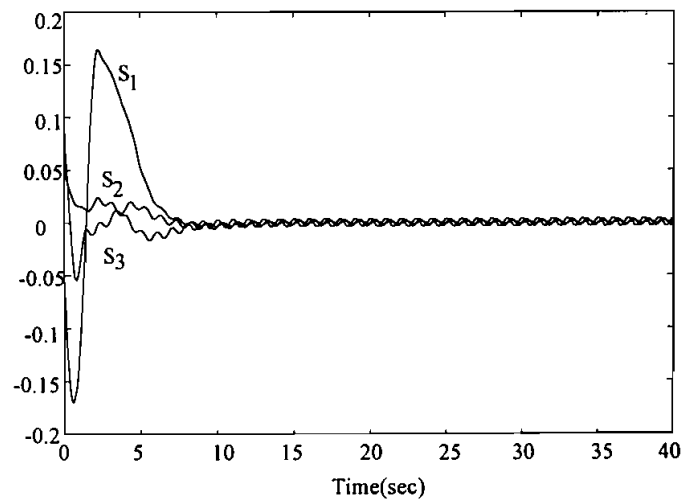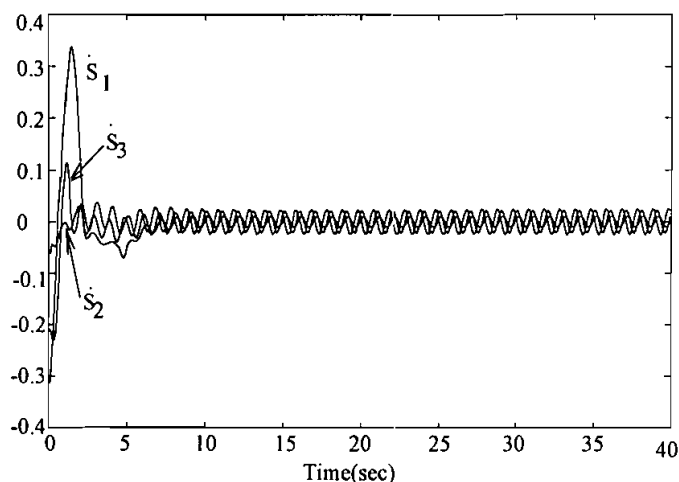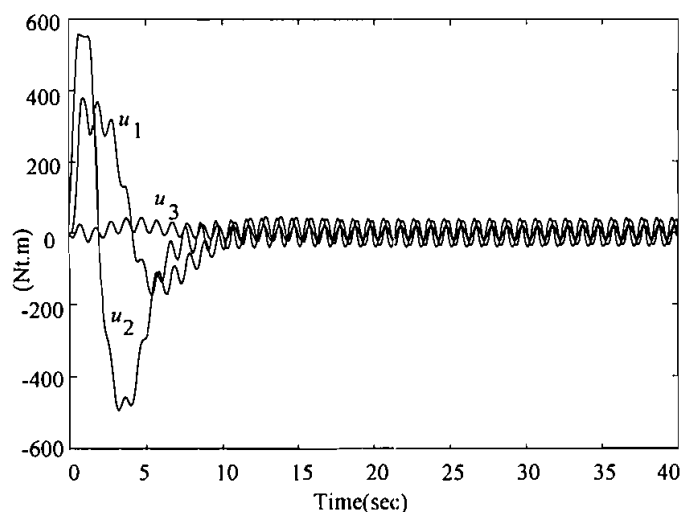

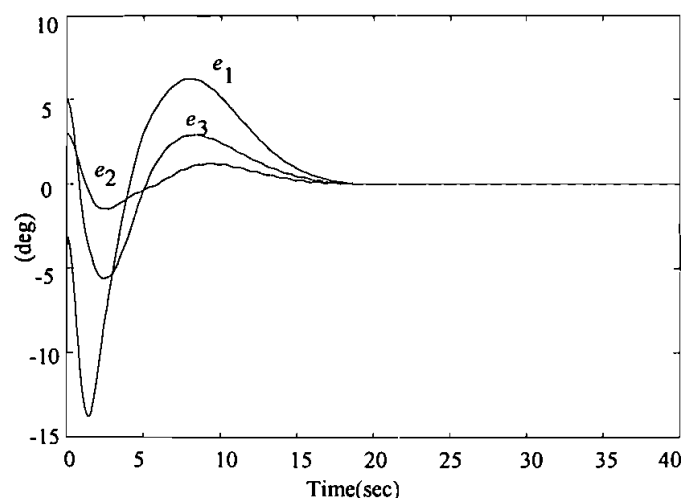**Figure 20.   Time responses of *e*.**

Figure 23.   Time responses of *S*.

Figure 24. Time responses of ṡ.

Figure 25. Time responses of u.

Figure 26. Time responses of e.

and Elmali's results. The successful trace of the roll, pitch and yaw angles to the corresponding desired angles can be seen from Figs 19, 20, 25 and 26. Hence, the robustness of unmodelled dynamics and external disturbances can be justified. In view of the simulation results, the present scheme outperforms both Singh and Elmali's schemes.

6. Conclusions

In this paper, a robust control scheme has been developed to overcome some of the difficulties encountered in the design and implementation of sliding mode control. It has been shown that the proposed scheme possesses the following attractive features.

(1) It eases the controller design for both SISO and MIMO nonlinear dynamic systems.

(2) Robustness against the modelling uncertainties and external disturbances is obtained.

(3) The phenomenon of chattering, inherent in sliding mode control can be effectively reduced.

(4) The controller design procedure of the system is facile and the online calculation is simple, thus a real-time simulation is feasible in the physical implementation of the example problems.

References

BAILEY, E., and ARAPOSTATHIS, A., 1987, Simple sliding mode control scheme applied to robot manipulators, *International Journal of Control*, 45, 1197–1209.

BULSARI, A., 1992, Training artificial neural networks for fuzzy logic. *Complex Systems*, 6, 443–457.

BULSARI, A., and SAXEN, H., 1991a, Implementation of chemical reactor selection expert system in a feedforward neural networks. *Proceedings of the Australian Conference on Neural Networks*, pp. 227–229; 1991b, Fuzzy logic inferencing using a specially designed neural network architecture. *Proceeding of the International Symposium on Artificial Intelligent Applications and Neural Networks*, pp. 57–60.

DeCARLO, R. A., ZAK, S. H., and MATTHEWS, G. P., 1988, Variable structure control of nonlinear multivariable systems: a Tutorial. *Proceedings of the Institute of Electrical and Electronics Engineers*, 76, 212–232.

ELMALI, H., and OLGAC, N., 1992, Robust output tracking control of nonlinear MIMO systems via sliding mode technique. *Automatica*, 28, 145–151.

HWANG, G. C., and LIN, S. C., 1992, A stability approach to fuzzy control design for nonlinear systems. *Fuzzy Sets and Systems*, 48, 279–287.

LI, D., and SLOTINE, J. J., 1987, On sliding control for multi-input multi-output nonlinear systems. *Proceedings of the America Control Conference*, pp. 874–879.

MAMDANI, E. H., 1974, Applications of fuzzy algorithms for control of simple dynamic plant. *Proceedings of the Institute of Electrical Engineers*, 121, 1585–1588.

NARENDRA, K. S., and PARTHASATATHY, K., 1990, Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks*, 1, 4–27.

SINGH, S. N., and IYER, A., 1989, Nonlinear decoupling sliding mode control and attitude control of spacecraft. *IEEE Transactions on Aerospace Electronic Systems*, **25**, 621–633.

SIRA-RAMIREZ, H., DWYER, III, T. A. W., 1987, Variable structure controller design for spacecraft nutation damping. *IEEE Transactions on Automatic Control*, **32**, 435–438.

SLOTINE, J. J., SASTRY, S. S., 1983, Tracking control of nonlinear systems using sliding surfaces, with application to robot manipulators. *International Journal of Control*, **38**, 465–492.

UTKIN, V. I., 1977, Variable structure systems with sliding modes. *IEEE Transactions on Automatic Control*, **22**, 212–222.