Iranian Journal of
Marine Technologies
Marine Science University of Imam Khomein

Robust Adaptive Fuzzy Sliding Mode Control of Permanent Magnet Stepper Motor with Unknown Parameters and Load Torque

H. Malekizade

h.malekizadeh@yahoo.com

Imam Khomeini Maritime Sciences University, Nowshahr, Mazandaran, Iran.

Abstract: In this paper, robust adaptive fuzzy sliding mode control is designed to control the Permanent Magnet (PM) stepper motor in the presence of model uncertainties and disturbances. In doing so, the nonlinear model is converted to canonical form, then, for designing the controller, the robust sliding mode control is designed to decrease the effects of uncertainties and disturbances. A class of fuzzy systems to approximate the ideal input is designed. Stability of closed loop control is also guaranteed by the Lyapunov function employed in controller design. Compared to previous studies that have used position, velocity, and currents phases in the feedback loop to produce and track control signal, the proposed algorithm only needs to measure position and velocity of the shaft for controlling performance of the motor. All the other parameters of motor and load torque are assumed to be unknown. The simulation results show that the control strategy is effective to the motor position tracking error and robust to overcome uncertainties and disturbances.

Keywords: Robust adaptive fuzzy sliding mode control, PM stepper motor, Stability of closed loop control.

Received Date : 1398/10/28

Accepted Date : 1399/01/07

1. Introduction

Position systems have been traditionally implemented using DC motors due to the relative ease in controlling them. The ease of control is due to the fact that the system equations describing a DC motor is linear. However, there are still disadvantages in using such motors for positioning systems due to their low torque to inertia ratio. Position systems are now being implemented using permanent magnet (PM) stepper motors. These motors have higher torque to inertia ratio than DC motors and are cheaper and brushless. Originally, stepper motors have been designed to provide precise positioning control within an integer number of steps. That is, they are open-loop stable [1].

A PM stepper motor converts electronic pulse into proportionate mechanical movement. Each revolution of the stepper motor's shaft is made up of series of discrete individual steps. A step is defined as the angular rotation produced by the output of the shaft each time the motor receives a step pulse. These types of motors are very popular in digital control circuits, such as robotics, because they are ideally suited for receiving digital pulses for step control. Each step causes the shaft to rotate a certain number of degrees [2]. Many control methods are proposed to control the position of PM stepper motors, but not all of them are suitable for tracking control. Nonlinear control system design has traditionally relied on linearized approximations of models treated by linear control methods. However, when the system in question is inherently nonlinear, linear approximations may be uninformative and designs based on them will exhibit unsatisfactory performance [3].

In the last decade, adaptive control has become popular. Adaptive tracking controller can be designed for PM stepper motor driving mechanical load [1,2]. In this method, extra control terms were defined. Also, to gain control signal, many variables should be measured that make complicated control system and the controlling process will be costly and time-consuming due to the number of sensors and devices and much computation. Some researchers developed sliding mode control [2-4]. [4] proposed static and dynamic sliding mode control that makes some intense variations in control signal. Both [3], [4] also have problems mentioned in [1,2]. Some scholars developed robust adaptive control for systems that have noise [4-8]. There are some studies [9-12] that used the linearization method and combination with neural network to have smooth variation in system output. In order to identify complicated system, neural network is used because of its potential to map inputs and outputs of the complicated systems [10-13]. Neural networks as a controller are more complicated than fuzzy systems. One of the main goals of this paper is to design a controller that needs fewer variables and less memory to use less sensor and cost, respectively. When neural networks are employed as online controller, they need time till the weight coefficients are adapted. So, another controller should be used till the training is completed or offline training is done. In some cases, to train neural networks, the designer needs to be aware of input variables in past tenses, so much memory is used [14]. Fuzzy system can be used online and it doesn't need to training algorithm. Also, the structure of fuzzy system is much easier. As a model free design method, fuzzy systems have been successfully applied to control complex or identified process whose mathematical models are difficult to obtain [13-17]. It is the well-known fact that fuzzy system, as well as neural networks, can approximate certain classes of functions to a given accuracy [16-21]. Adaptive fuzzy control (AFC) can be defined as control methodology. Successful applications of AFC and sliding mode control have been widely reported in the literature [19- 26]. However, in previous methods [1-4],[7,12], for tracking control, position, velocity, and currents phases wer used in the feedback loop to produce control signal. In addition, control signal in previous method had two or more dimensions.

In this paper, robust adaptive fuzzy sliding mode control is used to design a controller for PM stepper motor. Fixed structure fuzzy system with triangular membership functions is designed. The state errors are used as fuzzy inputs. So, practically, only position and velocity of shaft that are defined as state variables are needed for measurement. The proposed model only uses position and velocity of the shaft in order to create input control. Numerical simulation is used to consider the ability of the system to track the reference signal. Results show the system's efficient tracking in the presence of a large load torque.

The paper is organized as follows. In Section 2, we show motor model and convert it to canonical form. The direct adaptive fuzzy sliding mode control scheme for a class of nonlinear second order canonical SISO system is introduced in Section 3. Numerical simulations are shown in Section 4. Finally, the conclusion is given in Section 5.

2 PM Stepper Motor's Model

The equations describing the stepper motor are given as follows [1-3]:

$$\begin{cases} \dot{I}_a = \dfrac{1}{L}(V_a - RI_a + k_m \omega sin(N_r\theta_r)) \\ \dot{I}_b = \dfrac{1}{L}(V_b - RI_b - k_m \omega cos(N_r\theta_r)) \\ \dot{\omega} = \dfrac{1}{J}(-k_m I_a\, sin(N_r\theta_r) + k_m I_b\, cos(N_r\theta_r) - B\,\omega - T_L) \\ \dot{\theta}_r = \omega \end{cases} \tag{1}$$

where I_a is the current in winding A, I_b is the current in winding B, θ_r is the angular displacement of the shaft of the motor, ω is the angular velocity of the shaft of the motor, V_a is the voltage across the winding of phase A, V_b is the voltage across the winding of phase B, N_r is the number of rotor teeth, J is motor inertia, B is viscose friction coefficient, L and R are the inductance and the resistance of each phase winding, and k_m is the motor torque constant.

Equations (1) used to describe the stepper motor model are nonlinear. A nonlinear transformation, known as the Direct-Quadrature (DQ) transformation, can be used to transform these equations into a form, which is more suitable for designing nonlinear controllers. The DQ transformation is defined as:

$$\begin{bmatrix} I_d \\ I_q \\ \omega \\ \theta_r \end{bmatrix} = \begin{bmatrix} cos(N_r\theta_r) & sin(N_r\theta_r) & 0 & 0 \\ -sin(N_r\theta_r) & cos(N_r\theta_r) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ \omega \\ \theta_r \end{bmatrix} \quad (2)$$

And the new input voltages are defined as:

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = \begin{bmatrix} cos(N_r\theta_r) & sin(N_r\theta_r) \\ -sin(N_r\theta_r) & cos(N_r\theta_r) \end{bmatrix} \begin{bmatrix} V_a \\ V_b \end{bmatrix} \quad (3)$$

where V_d, I_d are the direct voltage and current and V_q, I_q are the quadrature voltage and current. By using the DQ transformation shown in equations (2) and (3), the equations (1) can be written as:

$$\begin{cases} \dot{I}_d = \frac{1}{L}(V_d - RI_d + N_r\omega L I_q) \\ \dot{I}_q = \frac{1}{L}(V_q - RI_q - N_r\omega L I_d - k_m\omega) \\ \dot{\omega} = \frac{1}{J}(k_m I_q - B\omega - T_L) \\ \dot{\theta}_r = \omega \end{cases} \quad (4)$$

We define new parameters D_1, D_2, …, D_5, u_q, u_d as follows:

$$\begin{cases} D_1 = \frac{R}{L}, \ D_2 = \frac{k_m}{L}, \ D_3 = \frac{k_m}{J}, \\ D_4 = \frac{B}{J}, \ D_5 = N_r, \ u_q = \frac{V_q}{L}, \ u_d = \frac{V_d}{L} \end{cases} \quad (5)$$

Substituting these new parameters in equation (4) results in:

$$\begin{cases} \dot{\theta}_r = \omega \\ \dot{\omega} = D_3 I_q - D_4\omega - \frac{T_L}{J} \\ \dot{I}_q = -D_1 I_q - D_5\omega I_d - D_2\omega + u_q \\ \dot{I}_d = -D_1 I_d + D_5\omega I_q + u_d \end{cases} \quad (6)$$

From equations (6) we have:

$$\begin{cases} I_q = -\frac{\dot{I}_q}{D_1} - \frac{D_5}{D_1}\omega I_d - \frac{D_2\omega}{D_1} + \frac{1}{D_1}u_q \\ I_d = -\frac{\dot{I}_d}{D_1} + \frac{D_5\omega}{D_1} I_q + \frac{1}{D_1}u_d \end{cases} \quad (7)$$

By replacing equations (7) into equations (6), the following second order nonlinear system is gained:

$$\begin{cases} \dot{\theta}_r = \omega \\ \dot{\omega} = F_1 + \frac{D_3}{D_1}(u_q - \frac{D_5\omega}{D_1}u_d) \end{cases} \quad (8)$$

where:

$$F_1 = -\frac{D_3}{D_1}\dot{I}_q - \frac{D_3D_2}{D_1}\omega + \frac{D_3D_5\omega}{D_1^2}\dot{I}_d - \frac{D_3D_5^2\omega^2}{D_1^2}I_q - D_4\omega - \frac{T_L}{J} = F_2 - \frac{T_L}{J} \quad (9)$$

The load torque is defined as follows [1]):

$$T_L = ml^2\ddot{\theta}_r + mgl\sin(\theta_r) \quad (10)$$

Equation (8) can be written as:

$$\begin{cases} \dot{\theta}_r = \omega \\ \dot{\omega} = F + \frac{D_3}{D_1\bar{J}}(u_q - \frac{D_5\omega}{D_1}u_d) \\ F = \frac{F_2 - \frac{mgl\sin(\theta_r)}{J}}{\bar{J}} \qquad \bar{J} = \frac{J + ml^2}{J} \end{cases} \quad (11)$$

Let define state vector $\boldsymbol{x} = [x_1, x_2]^T = [\theta_r, \dot{\theta}_r]^T$, equation (8) in state space form can be written as:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = F + \frac{D_3}{D_1}(u_q - \frac{D_5 x_2}{D_1}u_d) \end{cases} \quad (12)$$

To calculate $g(\boldsymbol{x})$ (input gain), at first, we define the positive constant parameter λ which is chosen by designer. Let, define input signals as follow:

$$u_q = \lambda u_d = \lambda u_c \quad (13)$$

where u_c is the control signal. From equations (12), (13), we have:

$$\frac{D_3}{D_1\bar{J}}(u_q - \frac{D_5x_2}{D_1}u_d) = \frac{D_3}{D_1\bar{J}}(\lambda - \frac{D_5x_2}{D_1})u_c = g(\boldsymbol{x})u_c \quad (14)$$

To design the controller, the sign of $g(\boldsymbol{x})$ should be constant for $\forall x \in \Omega_x$, in which Ω_x is the controllability region. In this paper, λ is designed till $g(\boldsymbol{x}) > 0$, $\forall x \in \Omega_x$.

From above equations we have:

$$\frac{D_3}{D_1\bar{J}}(\lambda - \frac{D_2x_2}{D_1}) > 0 \Rightarrow \lambda - \frac{D_2x_2}{D_1} > 0 \Rightarrow \lambda - \frac{N_r\omega L}{R} > 0 \Rightarrow \lambda > \frac{N_r\omega L}{R} \quad (15)$$

In real world λ can be simply chosen large enough to satisfy the above equation. State space model of the system can be written as:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = F + g(\boldsymbol{x})u_c + d \end{cases} \quad (16)$$

where d is all disturbances in the system. To design the controller, it is assumed that $F, g(\boldsymbol{x})$ are unknown. We only know the real value of the position and velocity of shaft. So, other parameters such as load torque, resistance, etc. are unknown. Fig. 1 shows the model of the system and control loop. As it is shown, the state vector of the system is compared with reference vector and makes error vector. By using error vector as fuzzy inputs and also applying the design sliding surface, controller can create efficient control signal that only need coefficient λ to compute u_d, u_q. These control signals are used to make phase voltages.

Nonlinear model of the system converts to DQ form and the DQ form converts to canonical form and will be used to design controller that is presented in the next section.

3 Adaptive Fuzzy Sliding Mode Control

3.1. Description of Fuzzy Logic Controllers

A zero-order Takagi-Sugeno fuzzy system with point fuzzification method, Mamdani product type inference, and center-average defuzzification approach are used in the proposed method. Let us define $M(a, b; \boldsymbol{X})$ as a nonzero membership function in the interval $\boldsymbol{X} \in (a, b)$, and zero in $\boldsymbol{X} \notin (a, b)$. Consider i^{th} rule of the proposed fuzzy system as follows [26]:

$R^{(i)}$: IF X_1 is A_1^i, and X_2 is A_2^i, THEN y is θ_i

In this paper, regarding Fig. 1, $\boldsymbol{X} = [X_1, X_2]^T = \boldsymbol{e} = [e, \dot{e}]^T \in U \subset R^n$ is the crisp input and $y \in V \subset R$ is crisp output, A^i_js are fuzzy sets with membership function $\mu_{A_j^i}(X_j) = M(a^i_{j1}, a^i_{j2}; X_j)$ for some $a^i_{j1} < a^i_{j2}$, $i = 1, 2, \ldots, m$, and $j = 1, 2, \ldots, n$, $m = 4$ is the number of rules, $n = 2$ is the number of input of fuzzy system, and $\boldsymbol{\theta} = [\theta_1, \theta_2, \theta_3, \theta_4]^T$ is the outputs of the rules.

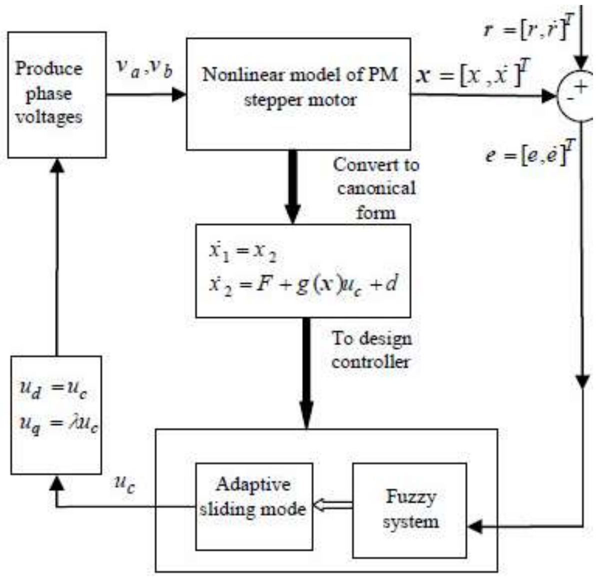

Fig. 1 Description of the system model and control loop.

Then, the output of a Takagi-Sugeno fuzzy system is defined as $\hat{y} := \hat{f}(\boldsymbol{x}, \boldsymbol{\theta})$, and obtained as a weighted average of the rule outputs by the following equation:

$$\hat{y} := \hat{f}(\boldsymbol{X}, \boldsymbol{\theta}) = \frac{\sum_{i=1}^{4} \theta_i \mu_i(\boldsymbol{X})}{\sum_{i=1}^{4} \mu_i(\boldsymbol{X})} = \sum_{i=1}^{4} \theta_i \xi_i(\boldsymbol{X}) \tag{17}$$

$$\mu_i(\boldsymbol{X}) = \prod_{j=1}^{2} \mu_{A_j^i}(X_j) \text{ and } \xi_i(\boldsymbol{X}) = \frac{\mu_i(\boldsymbol{X})}{\sum_{i=1}^{4} \mu_i(\boldsymbol{X})}$$

In literature, it has been demonstrated that this class of fuzzy system has universal approximation properties [24].

3. 2. Design of Adaptive Sliding Mode Control

In this section, the purpose is to synthesize a direct adaptive fuzzy controller for the system (16). A sliding surface can be defined in the error state as follows:

$$S = -\dot{e} - k_1 e \tag{18}$$

in which $\boldsymbol{e} = [e, \dot{e}]^T$, $e = r - y$, and k_1 is positive constant. By using (18), the time derivation of the switching surface S can be written as:

$$\dot{S} = -\ddot{e} - k_1 \dot{e} \tag{19}$$

Assumption1. The disturbance d is unknown but bounded, i.e. $d < d_{max}$

Assumption2. For the system (16) there exists a function $\beta(\boldsymbol{x}) > 0$ such that $\dot{S} = -\beta(\boldsymbol{x}) S$, [17].

As it is clear, for system (16), if the feedback linearization teqnique is used, the ideal feedback control law can be written as [24, 25]:

$$u^* = \frac{1}{g(\boldsymbol{x})}(-F + \boldsymbol{L}^{\mathrm{T}}\boldsymbol{e} + \ddot{r} - d) \tag{20}$$

By applying Eq. (20) into Eq. (16), we have:

$$\ddot{e} = -\boldsymbol{L}^T \boldsymbol{e} = -L_1 e - L_2 \dot{e} \tag{21}$$

where $\boldsymbol{L} = [L_1, L_2]^T$ is positive definite constant and is chosen such that Eq. (21) is Hurwitz stable, i.e. $\lim_{t\to\infty} e = 0$. Fuzzy system (17) is used to approximate the ideal input u^*. To guarantee stability of closed loop system, a supplementary signal u_s is added, hence, by using the definition (17), the proposed control signal is given by:

$$u_c = u_f + u_s,\ u_f = \boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{\xi}(\boldsymbol{X}),\ u_s = -\frac{S}{\rho^2} \tag{22}$$

Where $\boldsymbol{\xi}(\boldsymbol{X}) = [\xi_1(X), \xi_2(X), \ldots, \xi_m(X)]^T$ and ρ is positive constant representing the attenuation level of the effects of both approximation error and the external disturbances. We can define $\boldsymbol{\theta}^* = [\theta_1^*, \theta_2^*, \ldots, \theta_m^*]^T$, that is the desired value of θ. From (16), we have:

$$-\ddot{e} = F + g(\boldsymbol{x})u + d - \ddot{r} \tag{23}$$

$$\dot{S} = F + g(\boldsymbol{x})u + d - \ddot{r} - k_1 \dot{e} \tag{24}$$

By using (17), (20), (22), we can write [21]:

$$g(\boldsymbol{x})u - g(\boldsymbol{x})u^* = \sum_{i=1}^{n-1} c^i(\theta_i - \theta_i^*)\xi_i(\boldsymbol{X}) + \delta \quad (25)$$

where c^i is a positive constant. δ is obtained as follows

$$\delta = \varepsilon - \frac{S}{\rho^2}\sum_{i=1}^{m} c^i \quad (26)$$

where ε is the approximation error. Equation (26) is acceptable, because when $\boldsymbol{e} \to 0$, then $S \to 0$, as a result, $\varepsilon \to \delta$ We can write:

$$g(\boldsymbol{x})u - g(\boldsymbol{x})u^* = \sum_{i=1}^{m} c^i(\theta_i - \theta_i^*)\xi_i(\boldsymbol{X}) + \varepsilon - \frac{S}{\rho^2}\sum_{i=1}^{m} c^i \quad (27)$$

If we define some parameters such as:
$C = diag([c^1, c^2, c^3, c^4])$, $\psi^2 = \sum_{i=1}^{m} c^i > 0$ where ψ is a positive constant and $\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta} - \boldsymbol{\theta}^*$ then we have:

$$g(\boldsymbol{x})u - g(\boldsymbol{x})u^* = \tilde{\boldsymbol{\theta}}^T C\boldsymbol{\xi}(\boldsymbol{X}) + \varepsilon - \frac{S}{\rho^2}\psi^2 \quad (28)$$

From Assumption 1, Eq. (24) and replacing u by u^*, we have:

$$\begin{aligned} &F + g(\boldsymbol{x})u^* + v = 0 \\ &v = \beta(\boldsymbol{x})S - \ddot{r} - k_1\dot{e} + d \end{aligned} \quad (29)$$

By substituting (28) into (29), we have:

$$\begin{cases} g(\boldsymbol{x})u = -F - v + \tilde{\boldsymbol{\theta}}^T C\boldsymbol{\xi}(\boldsymbol{X}) + \varepsilon - \psi^2\dfrac{S}{\rho^2} \\ \Rightarrow F + g(\boldsymbol{x})u = -v + \tilde{\boldsymbol{\theta}}^T C\boldsymbol{\xi}(\boldsymbol{X}) + \varepsilon - \psi^2\dfrac{S}{\rho^2} \end{cases} \quad (30)$$

So, $\dot{S}$ can be rewritten as:

$$\begin{cases} \dot{S} = -v + \tilde{\boldsymbol{\theta}}^T C\boldsymbol{\xi}(\boldsymbol{X}) + \varepsilon + d - \psi^2\dfrac{S}{\rho^2} - \ddot{r} - k_1\dot{e} \\ \dot{S} = -\beta(\boldsymbol{x})S + \tilde{\boldsymbol{\theta}}^T C\boldsymbol{\xi}(\boldsymbol{X}) + \varepsilon - \psi^2\dfrac{S}{\rho^2} \end{cases} \quad (31)$$

For better application θ_i is tuned by following σ-modification robust adaptive law:

$$\dot{\theta}_i = -\gamma(S\xi_i(\boldsymbol{X}) - \sigma\theta_i) \quad (32)$$

Proof. Consider the Lyapunov function as following:

$$V = \frac{1}{2}S^2 + \frac{1}{2\gamma}\tilde{\boldsymbol{\theta}}^T C\tilde{\boldsymbol{\theta}} \quad (33)$$

By using this fact $\dot{\tilde{\boldsymbol{\theta}}} = \dot{\boldsymbol{\theta}}$, differentiating (33) along (31) yields:

$$\begin{cases} \dot{V} = \dot{S}S + \dfrac{1}{\gamma}\tilde{\boldsymbol{\theta}}^T C\dot{\tilde{\boldsymbol{\theta}}} \\ \Rightarrow \dot{V} = S(-\beta(\boldsymbol{x})S + \tilde{\boldsymbol{\theta}}^T C\boldsymbol{\xi}(\boldsymbol{X}) \\ \qquad + \varepsilon - \psi^2\dfrac{S}{\rho^2}) + \dfrac{1}{\gamma}\tilde{\boldsymbol{\theta}}^T C\dot{\tilde{\boldsymbol{\theta}}} \end{cases} \quad (34)$$

$$\begin{cases} \dot{V} = -\beta(\boldsymbol{x})S^2 + \tilde{\boldsymbol{\theta}}^T CS\boldsymbol{\xi}(\boldsymbol{X}) \\ \qquad + S\varepsilon - \dfrac{\psi^2 S^2}{\rho^2} + \dfrac{1}{\gamma}\tilde{\boldsymbol{\theta}}^T C\dot{\tilde{\boldsymbol{\theta}}} \\ \Rightarrow \dot{V} \le -\dfrac{\psi^2 S^2}{\rho^2} + S\varepsilon \\ \qquad + \dfrac{1}{\gamma}\tilde{\boldsymbol{\theta}}^T C(\dot{\boldsymbol{\theta}} + \gamma S\boldsymbol{\xi}(\boldsymbol{X})) \end{cases} \quad (35)$$

By substituting (32) into (35) and using following fact,

$$\begin{aligned} &\tilde{\boldsymbol{\theta}}^T C\boldsymbol{\theta} = \frac{1}{2}\boldsymbol{\theta}^{*T} C\boldsymbol{\theta}^* - \frac{1}{2}\tilde{\boldsymbol{\theta}}^T C\tilde{\boldsymbol{\theta}} - \frac{1}{2}\boldsymbol{\theta}^T C\boldsymbol{\theta} \le \\ &\frac{1}{2}\boldsymbol{\theta}^{*T} C\boldsymbol{\theta}^* \end{aligned} \quad (36)$$

we have:

$$\begin{cases} \dot{V} \le -\dfrac{\psi^2 s^2}{\rho^2} + S\varepsilon + \tilde{\boldsymbol{\theta}}^T C\sigma\boldsymbol{\theta} \\ \dot{V} \le -\dfrac{\psi^2 s^2}{2\rho^2} - \dfrac{\psi^2 s^2}{2\rho^2} + S\varepsilon + \dfrac{1}{2}\boldsymbol{\theta}^{*T} C\sigma\boldsymbol{\theta}^* \end{cases} \quad (37)$$

$$\begin{cases} \dot{V} \le -\dfrac{\psi^2 S^2}{2\rho^2} - \dfrac{1}{2}[\dfrac{\psi^2 S^2}{\rho^2} - 2S\varepsilon + \dfrac{\rho^2}{\psi^2}\varepsilon^2 - \\ \dfrac{\rho^2}{\psi^2}\varepsilon^2] + \dfrac{1}{2}\boldsymbol{\theta}^{*T} C\sigma\boldsymbol{\theta}^* \\ \dot{V} \le -\dfrac{\psi^2 S^2}{2\rho^2} - \dfrac{1}{2}(\dfrac{\psi S}{\rho} - \dfrac{\rho}{\psi}\varepsilon)^2 + \dfrac{\rho^2}{2\psi^2}\varepsilon^2 + \\ \dfrac{1}{2}\boldsymbol{\theta}^{*T} C\sigma\boldsymbol{\theta}^* \end{cases} \quad (38)$$

$$\dot{V} \le -\frac{\psi^2 S^2}{2\rho^2} + \frac{\rho^2}{2\psi^2}\varepsilon^2 + \frac{1}{2}\boldsymbol{\theta}^{*T} C\sigma\boldsymbol{\theta}^* \quad (39)$$

Integrating from above inequality from $t = 0 \to T$, we have:

$$\begin{cases} V(T) - V(0) \le \\ -\displaystyle\int_0^T \frac{\psi^2 s^2}{2\rho^2}dt + \frac{\rho^2}{2\psi^2}\int_0^T \varepsilon^2 dt + \frac{\sigma}{2}\int_0^T \boldsymbol{\theta}^{*T} C\boldsymbol{\theta}^* dt \end{cases} \quad (40)$$

$$\begin{cases} \displaystyle\int_0^T \frac{\psi^2 s^2}{2\rho^2}dt \le \\ -V(T) + V(0) + \dfrac{\rho^2}{2\psi^2}\displaystyle\int_0^T \varepsilon^2 dt + \frac{\sigma}{2}\int_0^T \boldsymbol{\theta}^{*T} C\boldsymbol{\theta}^* dt \end{cases} \quad (41)$$

Using this fact that $V(T) \ge 0$, above inequality can be simplified as:

$$\begin{cases} \int_0^T \frac{\psi^2 s^2}{2\rho^2} dt \le \\ V(0) + \frac{\rho^2}{2\psi^2}\int_0^T \varepsilon^2 dt + \frac{\sigma}{2}\int_0^T \boldsymbol{\theta}^{*T} C \boldsymbol{\theta}^* dt \end{cases} \tag{42}$$

Equation (42) guarantees that $S \in L_\infty$. Because all of the variables are in the right-hand side and are bounded, $\dot{s} \in L_\infty$. Since the right-hand side of (42) is also bounded $S \in L_2$, using Barbalat lemma, we have $S \rightarrow 0$ when $t \rightarrow \infty$. Therefore, the tracking error converges to the origin, i.e, $\lim_{t\rightarrow\infty} \boldsymbol{e} = 0$ [23].

In this paper, regarding Fig. 1, in order to design controller, the model of system is converted to the canonical form. Unlike other proposed method [1-4], [7,12], only the position and velocity of shaft are measured. The errors of shaft's position and velocity are used as fuzzy inputs and to design slide surface. Then, in adaptive sliding mode, the output of each rule is tuned by adaptive law. Then, the control signal is gained by (17), (22). Also, changing the model of system to canonical form results in tuning parameters for one input and another input is gained by multiplying a constant gain by first input. The results contribute to simplifying the designing process. The results of simulations in the next section show the outperformance of controller and proposed method.

4. Simulation Results

For simulation, motor's data is used as following, and also the results are compared with [1].

$R = 0.5\Omega, N_r = 50, K_m = 0.1 \;\; Nm/A, L = 0.00125H$

$J = 0.000027 kg.m^2,\ B = 0.000537 Nm/rad/s, K_D = 0$

The control objective is to bring motor position from initial time instant $\theta_d(t_i) = \theta_i = 0$ to final time $\theta_d(t_f) = \theta_f$. The additional constraints are chosen:

$$\dot{\theta}_d(t_i) = \dot{\theta}_d(t_f) = 0$$

$$\theta_d(t) = \theta_i + (\theta_f - \theta_i)(10\Delta_t^3 - 15\Delta_t^4 + 6\Delta_t^5)$$

Where $\Delta_t = \dfrac{t - t_i}{t_f - t_i},\ t_i = 0s$. If we define A_{11}, A_{12} as membership functions for first fuzzy set with input e_1 as shaft position error, A_{21}, A_{22} as membership functions for second fuzzy set with input e_2 as shaft velocity error and $\mu_{A_{jk}^i}(e_j)$ as membership value, k = 1, 2, k is number of membership functions in a fuzzy set, the rules of fuzzy system can be written as:

if e_1 is A_{11}^1 and e_2 is A_{21}^1 then $\mu_1 = \mu_{A_{11}^1}(e_1)\ g\ \mu_{A_{21}^1}(e_2)$

if e_1 is A_{12}^2 and e_2 is A_{21}^2 then $\mu_2 = \mu_{A_{12}^2}(e_1)\ g\ \mu_{A_{21}^2}(e_2)$

if e_1 is A_{11}^3 and e_2 is A_{22}^3 then $\mu_3 = \mu_{A_{11}^3}(e_1)\ g\ \mu_{A_{22}^3}(e_2)$

if e_1 is A_{12}^4 and e_2 is A_{22}^4 then $\mu_4 = \mu_{A_{12}^4}(e_1)\ g\ \mu_{A_{22}^4}(e_2)$

Load torque's data and controller parameters are given as:

$$mf_{e_1,e_2} = \begin{bmatrix} Left & center & right \\ -1 & -0.1 & 0.5 \\ -0.5 & 0.1 & 1 \end{bmatrix},\quad \rho = 0.01$$

$\gamma = 10^6, \quad S = -\dot{e} - 0.01e, \quad \lambda = 3$

$l = 0.1m, \quad m = 0.1, 0.6kg, \quad g = 9.81kgm/s^2$

The triangular membership function is used in fuzzy sets. This membership functions are the same for two fuzzy sets. To solve the differential equations, sampling time of 0.0001*s* is used. To simulate the different loads experienced by the stepper motor, different weights are attached to the motor shaft. At first, system simulation is done without noise and *m*=0.1 *kg*, θ_f =0.3 *rad*. In Fig 2, the tracking error of both adaptive method (Ad) [1] and proposed adaptive fuzzy sliding (AFS) are demonstrated. As shown, the position error of the proposed method is smaller than the adaptive method. The error after 0.8 s is approximately 2×10^{-4}.

Note: In [1], the simulation shows that the adaptive controller needs at least 5 s till the position error becomes approximately zero. But, in this paper, through simulation the rise time is decreased to 0.5s to show the outperformance of the proposed method.

To consider the performance of the controllers in different situation, m=0.6 kg is used. The tracking error is shown in Fig 3. The final error for the proposed method after 0.8 s is approximately 0.002 rad. This response shows the stability of the AFS in the presence of the weight variations. In next step, the final value of the reference is changed to θ_f =2 *rad*. The tracking error is shown in Fig 4. The final error after 0.8 s is approximately 0.003 rad.

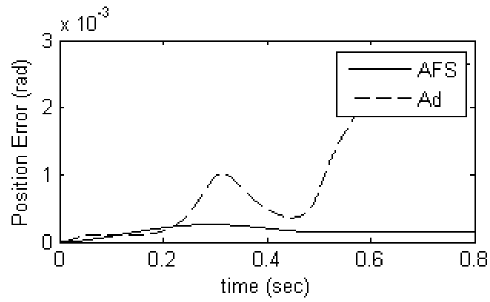

Fig. 2 The tracking error for adaptive and ASF, m=0.1 kg.

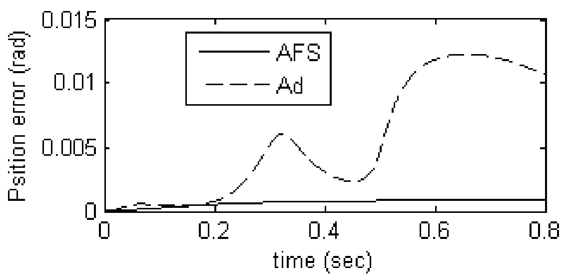

Fig. 3 The tracking error for adaptive and AFS, m=0.6kg.

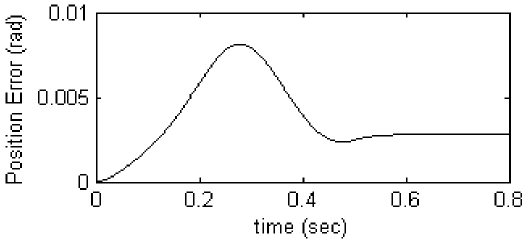

Fig. 4 The tracking error for AFS, m=0.6 kg maximum domain= 2 rad.

All these figures show that the proposed method is completely stable when the reference or some parameters change.
Note: Because the adaptive response is unstable, the error cannot be demonstrated in Fig 4.
Also, the adaptive controller is very sensitive to its values of the parameters and it needs to tune the values of the parameters for each variation. But, suitable values cannot be obtained till it becomes stable for new reference.
Now, the proposed method is tested in the presence of the input noise. The tracking error is shown in Fig 5. As shown, the maximum error is less than 0.001 rad. Fig.s 6 and 7 show the variations of the input voltages. The noise has a large domain while in Fig 5 this input noise is very small. These figures show that not only is the proposed method stable in different situations but it also can reject the noise in input signals. The variations of the input currents are shown in Fig 8. It is shown that during tracking the reference to final value, the voltages and currents have some variations. But, these variations become constant as soon as the output of the system obtains the final value.
In Fig (9), Eq. (15) is considered to prove the stability of the controller. It can be proved that the sign of $g(\boldsymbol{x})$ is positive all the time. In Eq. (15), it can be demonstrated that $\lambda - \frac{N_r \omega L}{R} > 0$, if we define $C = \lambda - \frac{N_r \omega L}{R}$, Fig.9 shows the variation of C during simulation.
The value of C is positive all the time because the sign of $g(\boldsymbol{x})$ is constant during simulation. This Fig. also shows that Eq. (15) is satisfied and closed loop system is stable.

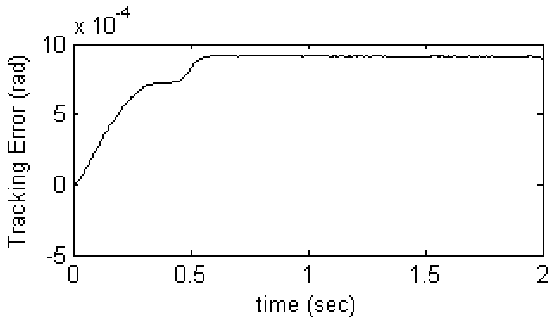

Fig. 5 The tracking error in presence if input noise m=0.6, output domain =03 rad.

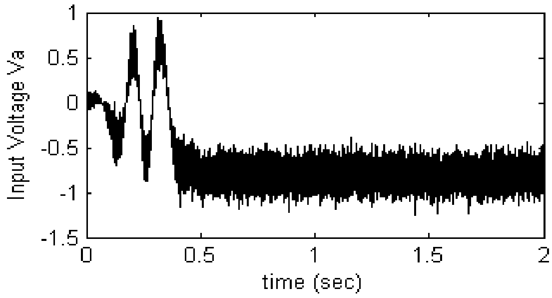

Fig. 6 The variation of Va in presence of noise.

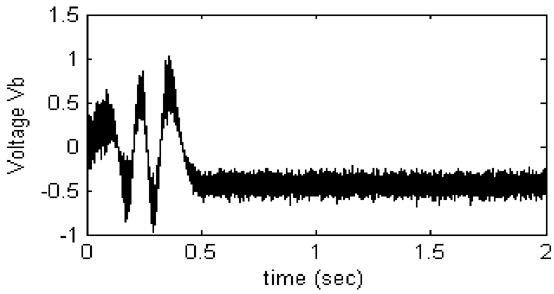

Fig. 7 The variation Vb in presence of noise.

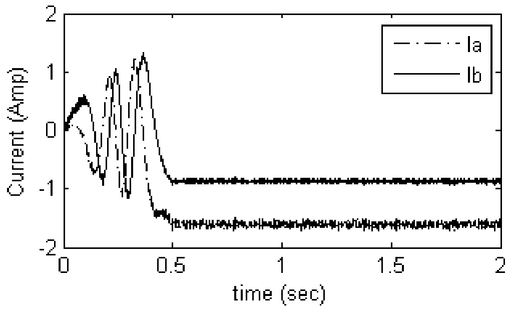

Fig 8. The variation of the currents in presence of noise.

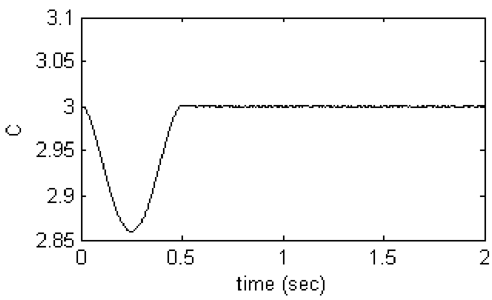

Fig. 9 The variation of the C.

All Figures show the efficiency of the controller in tracking the control signal and noise rejection.

5. Conclusion

A robust adaptive fuzzy sliding mode control method for the nonlinear model of the PM stepper motor in the presence of model uncertainties and input noise is proposed in this paper. The control command is created by proposing an adaptive fuzzy system with a fixed number of rules. In previous methods [1-4], [7,12], for tracking control, position, velocity and currents phases were used in the feedback loop to produce control signal. Also, control signal used in previous method had two or more dimension. In this paper, by using only position and velocity of the shaft, efficient control signal is created which could make a desired tracking in the presence of large load torque and noise. Also, only one input is adjusted by adaptive fuzzy sliding mode controller and another input is tuned by multiplying proportional gain by first input. It has been demonstrated that the proposed controller can be effective in different situations if the controller's parameters are tuned for the first time. By applying the suggested controller, the tracking of error in presence of the input noise is significantly reduced.

Reference

[1] Speagle, R. C., Dawson, D. M., “Adaptive Tracking Control of Permanent Magnet Stepper Motor Driving a mechanical Load”, IEEE Conference, 1993,

[2] Harb, A. M., Zaher, A. A., “Nonlinear Control of Permanent Magnet Stepper Motors”, Communications in Nonlinear Science and Numerical Simulation”, Vol.9, No. 4, pp.443-458, 2004.

[3] Nollet, F., Floquet, T., Perruquetti, W., “Observer based Second Order Sliding Mode Control Laws for Stepper Motors”, Control Engineering Practice, Vol. 16, No. 4, pp. 429-443, 2008.

[4] Zribi, M., Ramirez, H. S., Ngai, A., “Static and Dynamic Sliding Mode Control Schemes for a Permanent Magnet Stepper Motor”, International Journal of Control, Vol.74, No.2, pp.103-117 2001.

[5] Ge, S. S., Hang, C. C., Zhang, T., “A Direct Method for Robust Adaptive Nonlinear Control with Guaranteed Transient Performance”, Systems and Control Letters, Vol.37, No.5, pp.275-284, 1999.

[6] Ploh, A., Annaswmy, A. M. and Skantze, F., “Adaptation in the Presence of a General Nonlinear Parameterization :An Error Model Approach”, IEEE Transaction on Automatic Control, Vol.44, No.9, pp.1634-1652, 1999.

[7] Nealis, J. M., Smith, R. C., “Nonlinear Adaptive Parameter Estimation Algorithms for Hysteresis Models of Magnetostrictive Actuators”, Proceeding of the SPIE, Smart Structure and Materials, pp.25-36, 2002.

[8] Marino, R., Peresada, S., Tomei, P., “Nonlinear Adaptive Control of Permanent Magnet Step Motors”, Automatica, Vol.31, No.11, pp.1595-1604, 1995.

[9] Zribi, M., Chiasson, J., “Position Control of a pm Stepper Motor by Exact Linearization”, IEEE Trans on Automatic Control, Vol.36, No.5, pp.620-625, 1991.

[10] Minkov, M. D., Rodgerson, J., Harly, R. G., “Adaptive Neural Speed Controller of a DC Motor”, Electric Power Systems Research, Vol.47, No.2, pp.123-132, 1998.

[11] Nouri, K., Dhaouadi, R., Braiek, N. B., “Adaptive Control of a Nonlinear DC Motor Drive using Recurrent Neural Networks”, Applied Soft Computing, Vol.8, No.1, pp.371-382, 2007.

[12] Fallahi, M., Azadi, S., “Adaptive Control of a DC Motor using Neural Networks Sliding Mode Control”, Proceedings of the International Multi Conference of Engineers and Computer Scientists, 2009.

[13] Feng, G., “Position Control of a PM Stepper Motor using Neural Networks”, Proceeding of the IEEE Conference on Decision and Control, Vol.2, pp.1766-1769, 2000.

[14] Gomi, H., Kawato, M., “Neural Network Control for a Closed Loop System using Feedback Error Learning”, Neural Networks, Vol.6, No.7, pp.933-946, 1993.

[15] King, P. J. and Mamdani, E. H., “The Application of Fuzzy Control Systems to Industrial Process”, Automatica, Vol.13, No.3, pp.235-242, 1977.

[16] Takagi, T. and Sugeno, M., “Fuzzy Identification of Systems and its Applications to Modeling and Control”, IEEE Trans. on Systems, Man and Cybernetics, Vol.15, No.1, pp.116-132, 1985.

[17] Essounbouli, N. and Hamzaoui, A., “Direct and Indirect Robust Adaptive Fuzzy Controllers for a Class of Nonlinear Systems”, International Journal of Control, Automation and Systems, Vol.4, No.2, pp.146-154, 2006.

[18] Song, L. P., Zhang, R., Luo, Y., “Direct Torque Controlling of Permanent Magnet Synchronous Motor based on the Adaptive Fuzzy Controller”, Fuzzy Information and Engineering Volume 2 Advances in Soft Computing, Vol.62, pp.393-401, 2009.

[19] Wang. L. X. and Mendel. J. M., “Fuzzy Basis Functions Iniversal Approximation and Orthogonal Least-Squares Learning”, IEEE Transaction on Neural Network, Vol.3, No.5, pp.807-814, 1992.

[20] Castro, J. L., “Fuzzy Logic Controllers are Universal Approximators”. IEEE Transaction on Systems, Man and Cybernetics, Vol.25, No.4, pp.629-635, 1995.

[21] Hornik, K., “Multilayer Feedforward Networks are Universal Approximators”, Neural Networks, Vol.2, No.5, pp.359-366, 1989.

[22] Chang, W., “Design of Robust Fuzzy Model-based Controller with Sliding Mode Control for SISO Non-Linear Systems”, Fuzzy Sets and Systems, Vol.125, No.1, pp.1-22, 2002.

[23] Zhang, T., Ge, S. S., and Hang, C. C., “Direct Adaptive Control of Non-Affine Nonlinear Systems using Multilayer Neural Networks”, Proceeding of the American Control Conference. Philadelphia, pp.515-519, 1998.

[24] Phan, P .A., Gale, T .J., “Direct Adaptive Fuzzy Control with a Self-Structuring Algorithm”, Fuzzy Sets and Systems, Vol.159, pp.871-899, 2008.

[25] Chang, W.D., Hwang, R.C., Hsieh, J.G., “A Self-Tuning PID Control for a Class of Nonlinear Systems based on the Lyapunov Approach”, Journal of Process Control, Vol.12, No.2, pp.233-242, 2002.

[26] Moradi, M., Kazemi, M. H., Ershadi, E., “Direct Adaptive Fuzzy Control with Membership Function Tuning”, Asian Journal of Control, Vol.15, No.3, pp.1-10, 2013.