Gazi University Journal of Science
GU J Sci
28(1):141-148 (2015)

Trajectory Tracking Control of an Industrial Robot Manipulator Using Fuzzy SMC with RBFNN

Ayça GOKHAN AK[1,♠], Galip CANSEVER[2], Akın DELIBAŞI[2]

[1] *Marmara University Vocational School of Technical Sciences, Istanbul, 34722, TURKEY*

[2] *Yildiz Technical University, Control and Automation Engineering, Istanbul, 34220, TURKEY*

Received: 26/09/2012 Accepted:07/11/2014

ABSTRACT

One of the main problems associated with Sliding Mode Control (SMC) is that a whole knowledge of the system dynamics and system parameters is required to compute the equivalent control. Neural networks are popular tools for computing the equivalent control. In fuzzy SMC with Radial Basis Function Neural Network (RBFNN), a Lyapunov function is selected for the design of the SMC and RBFNN is proposed to compute the equivalent control. The weights of the RBFNN are adjusted according to an adaptive algorithm. Fuzzy logic is used to adjust the gain of the corrective control of the SMC. Proposed control method and a PID controller are tested on the Manutec-r15industrial robot manipulator. The real time implementations indicate that the proposed method can be applied to trajectory control applications of robot manipulators.

Key Words: *Neural Network; Fuzzy Logic; Sliding Mode Control; Robot Control.*

1. INTRODUCTION

The robot motion tracking control is one of the challenging problems due to the highly coupled nonlinear and time varying dynamics system. In addition, there are unknowns and uncertainties in the parameters of the mechanical part of the manipulators the actuating systems [1]. To design a controller for robot manipulators, it's necessary to have the exact trajectory tracking performance for reference inputs and the robustness for the external disturbances. Conventional feedback controllers are commonly used in the industry because their control architecture is very simple and easy to implement [2, 3]. However, when these conventional feedback controllers are directly applied to nonlinear systems, they suffer from poor performance and low robustness due to the unknown nonlinearities and external disturbances [4].

In the past decade the applications of intelligent control techniques (fuzzy logic or neural networks) to the motion control for robot manipulators have received considerable attention [5]. Saad et al. studied the trajectory tracking

♠Corresponding author, e-mail: aycaak@marmara.edu.tr

problem using neural network to control the nonlinear dynamic model of a robot. They utilize multilayer recurrent networks to estimate the dynamics of the system and inverse dynamic model [6]. Lee and Choi propose an adaptive neurocontroller for robot manipulators based on the radial basis function network [7]. Sun and Wang developed an adaptive fuzzy control strategy for robotic manipulators to guarantee both global stability and performance. Their control strategy employs a gradient descent algorithm to minimize a cost function which based on the error of the controller output and minimized by tuning some or all parameters of fuzzy controller [8].

Sliding Mode Control (SMC) has been recognized as a powerful control technique capable of making a control system very robust with respect to system uncertainties and external disturbances when the system trajectories belong to predetermined sliding surfaces. However, in order to compensate system uncertainties and operate effectively in the sliding surface, SMC requires instantaneous change of control input [9]. This may cause high frequency oscillations of the controller output, termed "chattering". In practical engineering systems, the chattering may cause damage to system components such as actuators. Another difficulty is the calculation the equivalent control, because a thorough knowledge of the plant dynamics is required for this purpose. Fuzzy logic and neural networks can provide an effective way to resolve these problems.

The integration of the fuzzy logic system in a sliding mode controller approach is seen many examples [1, 9, 10, 11]. Abdelhameed used chattering index to tune adaptively the switching gain of the sliding mode controller in order to shorten the duration of reaching phase and to minimize chattering of the control action of SMC [1]. Choi and Kim designed a fuzzy-sliding mode controller by fuzzifiying the sliding surfaces in order to attenuate the chattering [9]. Lin and Mon's paper presents a hybrid adaptive fuzzy control design methodology, where the nonlinear system is controlled by a state feedback controller and an adaptive fuzzy controller. In their adaptive fuzzy controller, an adaptive law is developed to tune a robust gain of the sliding-mode controller so as to cope with the uncertainties and model errors [10]. Guo and Woo used an adaptive single-input single output fuzzy system to calculate each element of the control gain vector in sliding mode controller [11].

Neural networks are used with SMC to abate of the SMC. Wai proposed a sliding–mode neural-network control system for the tracking control of an n rigid-link manipulator. Neural network controller used by him is developed to mimic an equivalent control law in the sliding mode control, and a robust controller is designed to curb the system dynamics on the sliding surface for guaranteeing the asymtotic stability property [5]. Ertugrul and Kaynak are utilized two parallel neural networks to realize a neuro-SMC. In their work equivalent control and corrective control terms of SMC are the outputs of the two layer feed-forward neural networks [12]. Especially multilayer feed forward neural network has been used to compute the equivalent control in literature.

In this paper Radial Basis Function Neural Network based fuzzy Sliding Mode Control methodology is proposed. Fuzzy logic is used to adjust the gain of the corrective control of the sliding mode controller. For the purpose of controlling the system states to hit the sliding surface and then slide along it, the weights of the RBFNN are adjusted according to an adaptive algorithm. Proposed method and PID control are tested on an Manutec-r15 industrial robot manipulator. The results obtained from the experiments are represented.

This paper is organized as follows: Fuzzy SMC with RBFNN is presented in the next section. Experimental system is described section 3. Section 4 contains the experimental results. Section 5 concludes the paper.

2. FUZZY SMC WITH RBFNN

2.1. Sliding Mode Control

Robot manipulators made of rigid links are generally described by following dynamic equation:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + F(q,\dot{q}) = u(t) \tag{1}$$

where $q, \dot{q}, \ddot{q} \in R^n$ are the joint position, velocity, and acceleration vectors, respectively; $M(q) \in R^{nxn}$ denotes the inertia matrix; $C(q,\dot{q}) \in R^{nxn}$ expresses the coriolis and centrifugal torques, $G(q) \in R^n$ is the gravity vector; $F(q,\dot{q}) \in R^{nxn}$ is the unstructured uncertainties of the dynamics including friction and other disturbances; $u(t) \in R^{nx1}$ is the actuator torque vector acting on the joints. The dynamics can be embedded in the state space representation as follows:

$$\dot{x}(t) = f(x,t) + Bu(t) \tag{2}$$

where B is the input gain matrix.

SMC has been developed for theoretical and practical studies of control engineering since 1070's by Utkin [13]. Thereafter Slotine, Edwards and Supergeon well developed the theoretical works of the Sliding Mode Controller and expanded its applications [14].

In general, the SMC can be separated into reaching mode and sliding mode. In reaching mode, trajectory starts from anywhere on the phase plane moving toward a switching surface and reaches the surface in finite time. In the SMC the trajectory asymptotically tends to the origin of the phase plane [15].

In order to derive the sliding mode control law, which forces the motion of the error to be along the sliding surface $s = 0$, the required sliding condition is defined as [16]:

$$\frac{1}{2}\frac{d}{dt}s^2 \le -\varsigma|s| \tag{3}$$

From this equation common notation for sliding mode can be obtained:

$$s\dot{s} \le -\varsigma|s| \equiv \dot{s}sign(s) \le -\varsigma \tag{4}$$

The $sign$ function is a discontinuous function;

$$sign(s) = \begin{cases} 1 & s > 0 \\ 0 & s = 0 \\ -1 & s < 0 \end{cases} \tag{5}$$

For the robotic system, the control objective is to make the robot track a given desired trajectory $q = q_d$. In this case, sliding surface is that $s = 0$ for some $\lambda > 0$.

$$s = \dot{e} + \lambda e \tag{6}$$

where λ is a positive constant, e is the tracking error equation ($e = q - q_d$).

The control should be chosen in such a way that the candidate Lyapunov function satisfies Lyapunov criteria. The positive definite Lyapunov function is selected as following,

$$V(s) = \frac{s^T s}{2} \tag{7}$$

It is aimed that the derivative of the Lyapunov function is negative definite. This can be obtainable if:

$$\frac{dV(s)}{dt} = -s^T D sign(s) \tag{8}$$

D is positive definite diagonal matrix. Taking the derivative of (7), equating (8) and using system equation, sliding mode control law can be represented as:

$$u = u_{eq} + u_c \tag{9}$$

where u_{eq} is the equivalent control law for sliding phase motion and u_c is the corrective control for the reaching phase motion. The control objective is to guarantee that the state trajectory can converge to the sliding surface. So, corrective control u_c is chosen as follows:

$$u_c = K sign(s) \tag{10}$$

where K is a positive constant. The controller in (10) results with high frequency oscillations, defined as chattering.

2.2. Design of Fuzzy SMC with RBFNN

The control law for the planned controller is as in (9). The configuration of proposed controller is shown in Fig.1. In proposed method, a RBFNN is employed to model the relationship between the sliding surface variable, s, and the equivalent control, u_{eq}. In other words, sliding variable, s, will be used as the input signal for establishing a RBFNN model to calculate the equivalent control.

RBFNN have been widely used to represent the nonlinear mappings between inputs and outputs of nonlinear control systems. The architecture of RBFNN is as shown in Fig. 2.

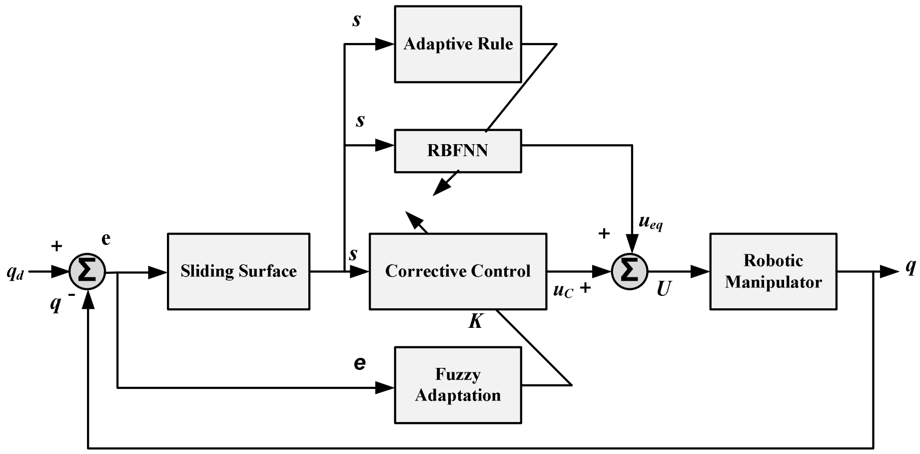


Figure 1. Block diagram of the Proposed Controller.

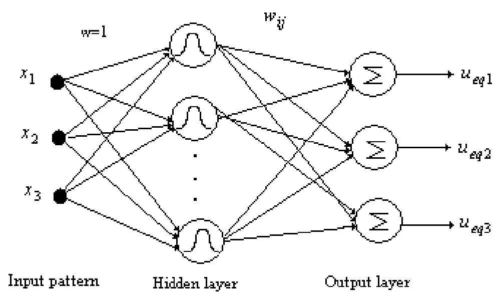


Figure 2. Radial Basis Function Neural Network (RBFNN).

The Gaussian function is used as the activation function of each neuron in the hidden layer. The network consists of three layers: an input layer, a single layer of nonlinear processing neurons, and an output layer. The excitation values of this Gaussian function are distances between the input values of the sliding surface value, *s*, and the central positions of the Gaussian functions [17].

$$g(s) = \exp\left\{-\left(\|s - c_j\| / \sigma_j\right)^2\right\} \tag{11}$$

where j is the j. neuron of the hidden layer and c_j characterizes the centres of the Gaussian function. The variable σ_j determines how the function (g) spreads over the domain of its input space. Weightings between input and hidden layer neurons are specified as constant 1. Weightings between hidden and output layer neurons (w_j) are adjusted based on adaptive rules.

The overall outputs of the structure are evaluated by a weighted sum of the neurons contained in the hidden layer and is described by (12).

$$u_{eq} = \sum_{j=1}^{n} w_j g_j(s) = w^T g(s) \tag{12}$$

The weightings of the RBFNN should be regulated based on the reaching condition, $s\dot{s} < 0$. An adaptive rule is used to adjust the weightings for searching the optimal weighting values, and obtaining the stable convergent property. The adaptive rule is derived from the steep descent rule to minimalize the value of $s\dot{s}$ with respect to w_j. The updated equation of the weighting parameter is following:

$$\dot{w}_j = -\tau \frac{\partial s(t)\dot{s}(t)}{\partial w_j(t)} \tag{13}$$

where τ is the adaptive rate parameter. Using the chain rule, (13) can be rewritten as follows:

$$\begin{aligned} \dot{w}_j &= -\tau \frac{\partial s(t)\dot{s}(t)}{\partial u_{eq}(t)} \frac{\partial u_{eq}(t)}{\partial w_j(t)} = \tau B s(t) \frac{\partial u_{eq}(t)}{\partial t} \\ &= \gamma s(t) \exp\left(-\frac{\|s - c_j\|}{\sigma_j^2}\right) = \gamma s(t) \phi_j(s) \end{aligned} \tag{14}$$

where τ and system parameter are combined as a learning parameter, γ.

The corrective control gain of the SMC is computed with fuzzy adaptation in the second proposed controller. s is the input and K is the output of the fuzzy system. s_i has membership functions: nb, nm, ns, z, ps, pm, pb (Fig 3a) and K_i has membership functions: z, ps, pm, pb (Fig 3b). Rule base is selected as follows:

IF s_i is nb, THEN K_i is b

IF s_i is nm, THEN K_i is m

IF s_i is ns, THEN K_i is s

IF s_i is z, THEN K_i is z

IF s_i is ps, THEN K_i is s

IF s_i is pm, THEN K_i is m

IF s_i is pb, THEN K_i is b

They are all Gaussian membership functions defined as follows:

$$\mu_A(s_i) = \exp\left(-\left(\frac{s_i - \alpha}{\sigma}\right)^2\right) \tag{15}$$

From our knowledge of the fuzzy systems, K_i can be written as

$$K_i = \frac{\sum_{m=1}^{M} \theta_{k_i}^m \mu_{A^m}(s_i)}{\sum_{m=1}^{M} \mu_{A^m}(s_i)} = \theta_{ki}^T \psi_{ki}(s_i) \tag{16}$$

(a) (b)

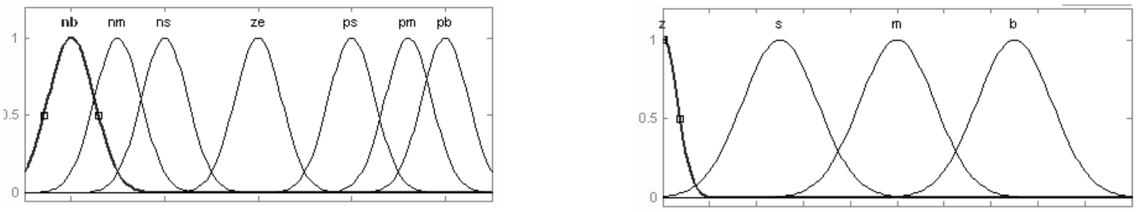


Figure 3. Membership Functions (a) Input (b) Output.

where $\theta_{ki} = \left[\theta_{ki}^{1},,...,\theta_{ki}^{m},...,\theta_{ki}^{M}\right]^{T}$ is the vector of the center of the membership functions of K_i $\psi_{ki}(s_i) = \left[\psi_{ki}^{1}(s_i),...,\psi_{ki}^{m}(s_i),...\psi_{ki}^{M}(s_i)\right]^{T}$ is the vector of the height of the membership functions of K_i in which $\psi_{ki}^{m}(s_i) = \mu_{A^m}(s_i) / \sum_{m=1}^{M} \mu_{A^m}(s_i)$, and M is the amount of the rules.

3. EXPERIMENTAL SYSTEM

The robot manipulator used in experiments is Manutec-r15 (Fig 4). Manutec-r15 is a robot with 6 degree of freedom. Manutec-r15 has 3 phase, brushless synchronous motors and incremental encoders (axis 1:600, axis 2: 1200, axis 3: 600, axis 4: 360, axis 5: 320, axis 6: 500).

Figure 4. Manutec-r15.

The malfunctioned original control and driver unit of the manutec-r15 was deactivated, and a new transformer, a new driver system, a new control system was built up to try the control methods on the robot. The block diagram of the system is shown in Fig.5.

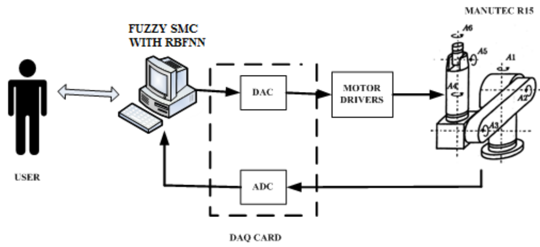


Figure 5. Block diagram of the system.

A circuit is designed to isolate the digital input-output. Simple filters were emplaced to eliminate wrong measures because of the cable length that connects the systems.

Data acquisition and control implementation were performed using the 16 Bits, 299kS/s, Quanser Q8 I/O board. Features of the computer used on the system are Pentium 4, CPU 3.2 GHz and 512 MB RAM. Proposed control methods are tested on the robot with data obtained from a DAQ card through Wincon software on a Matlab Simulink. Simulink block diagrams of the system is on Figure 6.

4. EXPERIMENTAL RESULTS

Experimental results for base two axes of the robot are presented here. The reference positions for axis 1 and 2 were 0^{o} and 157^{o} respectively. The base axis target trajectory was a sinus with 10^{o} amplitude and 0.2rad/s frequency. The target trajectory for the other axis was a sinus with 5^{o} amplitude and 0.2rad/s frequency. γ learning rate was selected as 0.01.

The desired and actual positions were close to each other for both proposed methods for axis 1 and 2. Results of proposed method are better than classical PID control as shown in Fig.7.

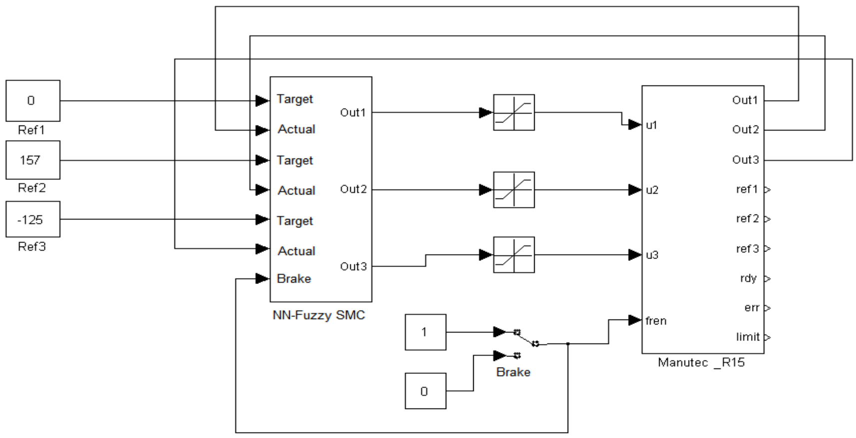
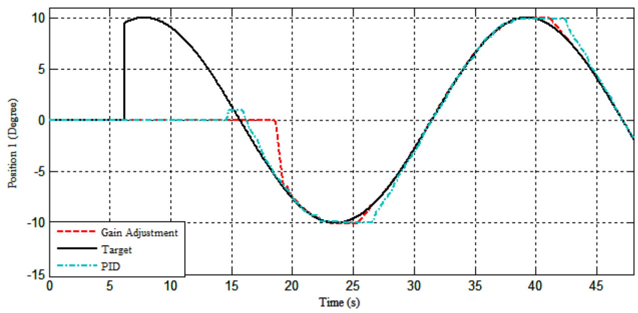
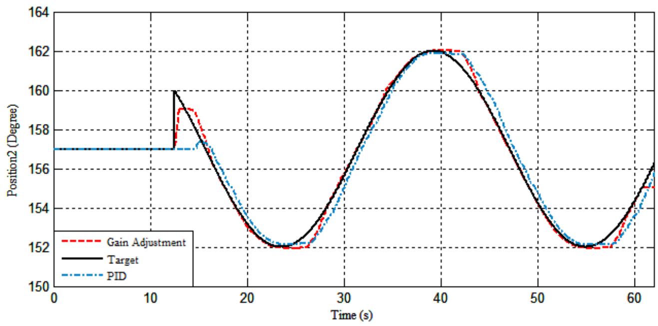


Figure 6. Simulink block diagrams of the system.



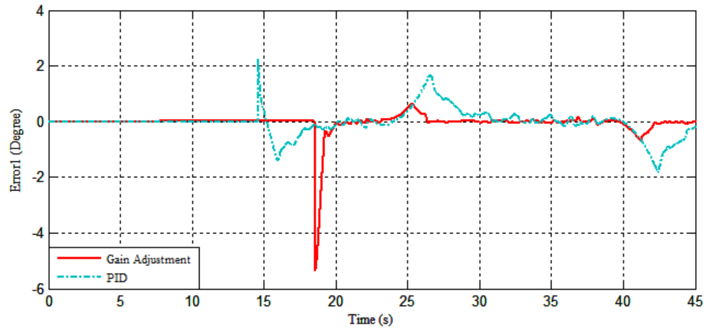
(a)



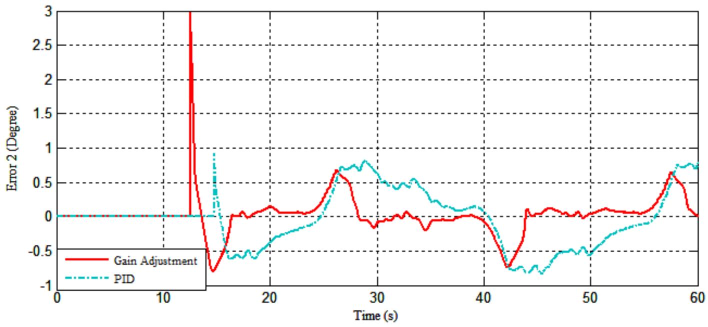
(b)

Figure 7. Target and actual states positions of two axes for Fuzzy SMC with RBFNN (a) for axis 1 (b) for axis 2.

Tracking errors of the two axes are as shown in Fig 8. Position errors are at a maximum at peak positions of the sinus. Errors are smaller on adjusting the gain with fuzzy logic. Control input torques for both methods are as shown in Fig. 9. Torques are quite close to each other on both methods for axis 1 and axis 2. There were no chattering on control signals.

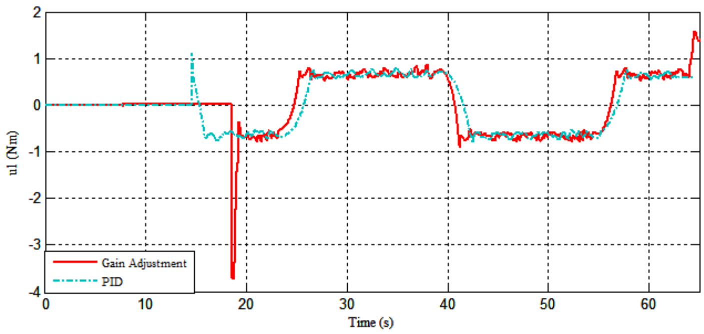


(a)

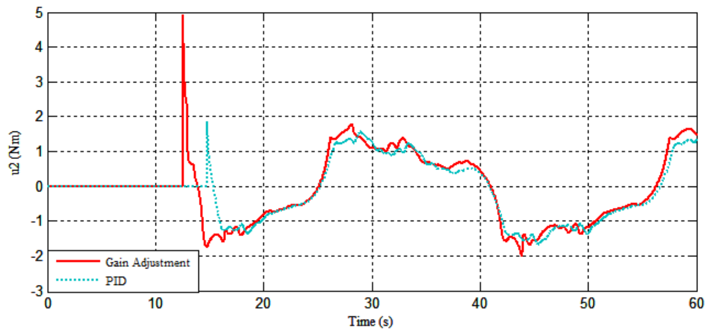


(b)

Figure 8. Position error for two axes for Fuzzy SMC with RBFNN (a) for axis 1 (b) for axis 2.



(a)



(b)

Figure 9. Control Torque Inputs for two axes for Fuzzy SMC with RBFNN (a) for axis 1 (b) for axis 2.

6. CONCLUSIONS

Many control methods about trajectory tracking applications of robotic manipulator have been put forward and reported in literature. In the classical SMC, the corrective control gain may choose larger number, which causes the chattering on the sliding surface or, corrective control gain may choose smaller number, which cause increasing of reaching time and tracking error. Here an adaptive sliding mode control scheme based on a fuzzy neural network for manipulators is presented. Using fuzzy controller to adjust the corrective control gain in SMC, system performance is improved. RBFNN is used to compute the equivalent control. So, system parameters aren't needed to compute the equivalent control.

The real time implementations show that the joint position tracking responses closely follow the desired trajectory for proposed method. Results demonstrate that the proposed fuzzy SMC with Radial Basis Function Neural Network controller in this paper, is a stable control method for robotic manipulators trajectory tracking applications

ACKNOWLEDGMENT

This research has been supported by the Yildiz Technical University Scientific Research Projects Coordination Department. Project number: 25-04-02-03.

CONFLICT OF INTEREST

No conflict of interest was declared by the authors.

REFERENCES

1. Abdelhameed, M., Adaptive Neural Network Based Controller for Robots, ***Mechatronics***, 147-162, (1999).
2. Yimin, Li., I-io, Y., and Chua, C., Model-Based PID Control of Constrained Robot in a Dynamic Environment with Uncertainty, ***IEEE International Conference on Control Applications***, Alaska, (2000).
3. Lee, MJ. and Choi, Y.K., An Adaptive Neurocontroller Using RBFN for Robot Manipulators**,** ***IEEE Transaction on Industrial Electronics,*** Vol.51, No:3, (2004).
4. Cervantes, I. and Alvarez-Ramirez, J., On the PID Tracking Control of Robot Manipulators, ***Systems & Control Letters*** 42, 37-46, (2001).
5. Wai, R., Tracking Control Based on Neural Network Strategy for Robot Manipulator, ***Neurocomputing*** 51, 425-445, (2003).
6. Saad, M., Bigras, P., Dessaint, L. and Al-Haddad, K., Adaptive Robot Control Using Neural Networks, ***IEEE Transaction on Industrial Electronics,*** Vol.41, No: 2., (1994).
7. Lee, M. and Choi, Y., An Adaptive Neurocontroller Using RBFN for Robot Manipulators, ***IEEE Transaction on Industrial Electronics***, Vol.51, (2004).
8. Sun, W. and Wang, Y., An Adaptive Fuzzy control for Robotic Manipulators, International Conference on Control, Automation, ***Robotics and Vision, Kunming China***, 1952-1956, (2004).
9. Choi, S. and Kim, J., A Fuzzy-Sliding Mode Controller for Robust Tracking of Robotic Manipulators, ***Mechatronics*** Vol.7, 199-216, (1997).
10. Lin, C. and Mon, Y, Hybrid Adaptive Fuzzy Controllers with Application to Robotic Systems, ***Fuzzy sets and systems***, 151-165, (2003).
11. Guo, Y. and Woo, P., An Adaptive Fuzzy Sliding Modem Controller for Robotic Manipulators, ***IEEE Transactions on System, Man, and Cybernetics-Part A: Systems and Humans***, Vol.33, 149-159, (2003).
12. Ertugrul, M. and Kaynak, O., Neuro sliding mode control of robotic manipulators, ***Mechatronics*** 10, 239-263, (2000).
13. Utkin, V.I., Variable Structure Systems with Sliding Modes. ***IEEE Transaction on Automatic Control*** AC-22, 212-222, (1977).
14. Edwards, C. and Spurgeon, K., Sliding Mode Control. ***Taylor&Fransis Ltd***. (1998).
15. Tsai, C., Chung, H. and Yu, F., Neuro-Sliding Mode Control with Its Applications to Seesaw Systems. ***IEEE Transaction on Neural Networks***, Vol.15, (2004).
16. Bekit, B. W., Whidborne, J.F. and Seneviratne, L.D. Fuzzy Sliding Mode Control for a Robot Manipulator, ***Computational Intelligence in Robotics and Automation***, 320–325, (1997).
17. Ham, F. M. and Kostanic, I., Neurocomputing for Science& Engineering. ***Mc Graw-Hill Inc.*** (2001).