Araştırma Makalesi

www.ejosat.com ISSN:2148-2683

Research Article

Neural Network Based Sliding Mode Controller with Genetic Algorithm for Two Link Robot Manipulator*

Melikcan Ölgün[1], Umut Tilki[2 **]

[1] Suleyman Demirel University, Faculty of Engineering, Departmant of Electronics and Communication Engineering, Isparta, Türkiye (ORCID: 0000-0002-2892-9319)

[2,*] Suleyman Demirel University, Faculty of Engineering, Departmant of Electrical and Electronics Engineering, Isparta, Türkiye (ORCID: 0000-0002-8988-787X)

(Conference Date: 5-7 March 2020)

(**DOI:** 10.31590/ejosat.araconf16)

ATIF/REFERENCE: Ölgün, M. & Tilki, U. (2020). Neural Network Based Sliding Mode Controller with Genetic Algorithm for Two Link Robot Manipulator. *Avrupa Bilim ve Teknoloji Dergisi*, (Special Issue), 120-129.

Abstract

In this paper, a novel control method is proposed in order to control the trajectory of two link robotic manipulator. The model of the manipulator has some unknown parameters because of the elasticity of the used materials for robot manipulator. Besides the robotic manipulators are generally open for the external disturbances. Radial basis function neural networks are employed in order to model these unknown parameters. The controller structure for trajectory tracking for the robot manipulator is based on robust and adaptive Sliding Mode Control (SMC). The coefficients of the SMC is calculated by the help of evolutionary algorithm, namely genetic algorithm. The proposed algorithm guarantees that the tracking error converges to zero in a finite time. The stability of the closed loop system is ensured by Lyapunov Theory. Numerical simulations have been conducted in Matlab/Simulink environment to demonstrate the validity of the proposed controller. Moreover, effectiveness and validity of the proposed control method is confirmed by comparative simulation results. The conducted experiments are demonstrated that proposed controller has disturbance rejection, chattering free, robust and fast property when we compared with other approaches in the literature.

Keywords: Sliding Mode Control, Genetic Algorithm, Robot Manipulator, Trajectory Tracking, RBFNN.

İki Linkli Manipülatör için Genetik Algoritma ile Yapay Sinir Ağı Tabanlı Kayan Kipli Kontrolcü

Öz

Bu makalede, iki eklemli robot manipülatörün yörünge kontrolü için yeni bir kontrol yöntemi önerilmektedir. Manipülatör modeli, robot manipülatör için kullanılan malzemelerin esnekliği nedeniyle bazı bilinmeyen parametrelere sahiptir. Ayrıca robot manipülatörler genellikle dış gürültülerden etkilenir. Bu bilinmeyen parametreleri modellemek için radyal bazlı fonksiyon tabanlı yapay sinir ağları kullanılır. Robot manipülatörün yörünge takibi için kontrolcü yapısı, gürbüz ve adaptif kayan kipli kontrolcüye (SMC) dayanmaktadır. SMC'nin katsayıları evrimsel algoritma, yani genetik algoritma yardımıyla hesaplanır. Önerilen algoritma, izleme hatasının belli bir sürede sıfıra yakınlaşmasını garanti eder. Kapalı döngü sisteminin kararlılığı Lyapunov Teorisi ile sağlanmaktadır. Matlab / Simulink ortamında önerilen denetleyicinin geçerliliğini göstermek için sayısal simülasyonlar yapılmıştır. Ayrıca, önerilen kontrol yönteminin etkinliği ve geçerliliği karşılaştırmalı simülasyon sonuçları ile teyit edilmiştir. Yapılan deneyler, önerilen denetleyici yapısının literatürdeki diğer yaklaşımlarla karşılaştırdığımızda gürültülere dayanıklı, tırlama etkisinden uzak, sağlam ve hızlı özelliklere sahip olduğunu göstermektedir.

Anahtar Kelimeler: Kayan Kipli Denetleyici, Genetik Algoritma, Robot Manipülatör, Yörünge Takibi, RBFNN.

---

*This paper was presented at the ***International Conference on Access to Recent Advances in Engineering and Digitalization (ARACONF 2020)***.

** Corresponding author: Suleyman Demirel University, Faculty of Engineering, Departmant of Electrical and Electronics Engineering, Isparta, Türkiye (ORCID: 0000-0002-8988-787X), umuttilki@sdu.edu.tr

1. Introduction

Nowadays, due to the advancement of robot technology and the widespread usage of manipulators, trajectory tracking control of these systems is the focus of studies in robotics. As the requirement of high-performance requests such as fast response trajectory tracking with high precision increases, the new control strategies for manipulators develop.

The main purpose of the trajectory tracking problem is to improve the stability of the overall system and to minimize the trajectory tracking errors. For this aim, in this work a novel controller method, which is Neural Network Based Adaptive Sliding Mode Controller with Genetic Algorithm (NNSMACWGA), is developed. In this controller model uncertainties such as link flexibility and friction terms in the nonlinear two link robot manipulator equations are modeled by using Radial basis function neural networks. Desired trajectory is tracked by the help of adaptive sliding mode approach (ASMC). The coefficients in this ASMC controller is calculated with evolutionary optimization technique, namely genetic algorithm (GA) in order to get the fastest responses with minimum trajectory tracking errors.

Control strategies for industrial robotic manipulators depend on model of the manipulator is exact or unknown. Because of the simple mathematical model, PID controllers are widely employed various applications for trajectory control of robot manipulators (Choi & Chung, 2004; Kumar et.al. , 2011). Although this controller gives satisfactory results according to trajectory tracking performance, the stability and convergence speed highly depends on the assigning proportional, integral and derivative coefficients. Moreover, manipulator dynamics has some unknown parameters due to the fact that having non rigid materials and open to external disturbances. Under varying conditions of the model with PID controller gives better result if the controller coefficients are dynamic (Loucif et.al. , 2020). Since robotic manipulators are used for highly precision trajectory tracking tasks, instead of PID controller, sliding mode controllers, which have variable control structure, give robust response for trajectory tracking against aforementioned model uncertainties and external disturbances (Tilki & Olgun, 2019). The principle of SMC is composed of two subsections. In the first part, proper sliding surfaces, which are needed to be followed by the determined state variables in order to reach desired trajectory, are constructed. In second part, switching robust control laws enabling the state variables to reach these sliding surfaces constructed in the first stage (Nguyen et.al. , 2018). The disadvantage of SMC, namely chattering phenomena, which is switching the control signal fast to handle disturbance and uncertain model effects, occurs during the sliding phase. Trajectory tracking control of 3-DOF manipulator is ensured with adaptive fuzzy sliding mode control technique (Amer et.al. , 2011). A supervisory fuzzy logic control was employed to replace switching terms and switching gains in the robust control input for SMC. Further study of this approach integrated adaptive laws were generated in order to eliminate chattering effects (He et.al. , 2016). Recently, to overcome the finite time convergence problem of traditional SMC, Terminal Sliding Mode Control approach was considered for robust trajectory tracking against model uncertainties (Tran & Kang, 2017). In TSMC method, sliding surfaces are modeled with nonlinear functions, which provides TSMC to enable fast convergence without extensive control inputs (Vijay & Jena, 2018). When compared with traditional SMC, it is stated that the response of TSMC method is rapidly converge the desired trajectory.

In this work, model free system is preferred and the complexity of predicting the exact model of the robot manipulator is prevented. Neural Network technique is widely employed in order to deal with model uncertainties in the literature (Vijay & Jena, 2018; Sun et.al. 2011). Because of this advantage, for modeling the uncertain terms in model free approach of robotic manipulator radial basis function of neural networks are employed.

The rest of this paper is organized as follows: Mathematical preliminaries of the trajectory control of robot manipulator are given in Section 2. Two different controller design techniques namely neural network based sliding mode controller method and neural network based adaptive sliding mode controller with genetic algorithm method are also given in this section. The simulation results of these controllers are given and compared in Section 3. Concluding remarks are discussed in Section 4.

2. Material and Method

2.1. Mathematical Dynamics of 2 DOF Robot Manipulator

In this paper, for a two-link manipulator which is subject to uncertainties and external disturbances in a finite time convergence guaranteed, robust trajectory tracking control problem is addressed. This section gives the mathematical background of the two link manipulator and proposed trajectory control methods.

In this work, proposed control methods are examined and validated on 2 link robot manipulator. For an n-link robot manipulator the equation of motion is given in equation (1) according to Euler-Lagrange theory (Lewis et.al. 2003).

$$M(q)\ddot{q} + V_m(q,\dot{q})\dot{q} + G(q) + F(\dot{q}) + \tau_d = \tau \qquad (1)$$

In this equation $q \epsilon R^n, \dot{q} \epsilon R^n$ and $\ddot{q} \epsilon R^n$ represents position, velocity and acceleration, respectively. Besides, $M(q)\epsilon R^{nxn}$ expresses positive-definite symmetric inertial matrix. $V_m(q,\dot{q})\epsilon R^{nxn}$ shows coriolis and centripetal force matrix. $G(q)\epsilon R^{nx1}$ represents the gravity component in the torque equation (1). As mentioned before, the overall system is open to external disturbances and uncertainties (friction) due to the model uncertainties. These terms are represented in the equation of motion with $F(\dot{q})$ and τ_d.

Equation of motion (1) can be expressed as follow:

$$\ddot{q} = M^{-1}(q)[\tau - (V_m(q,\dot{q})\dot{q} + G(q) + F(\dot{q}) + \tau_d)] \qquad (2)$$

The matrix and vector quantities and also the units in the above-mentioned manipulator equations which are used in this work are given as follows.

$M(q)=\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}$, $V_m(q,\dot{q})=\begin{bmatrix} V_{m11} & V_{m12} \\ V_{m21} & V_{m22} \end{bmatrix}$

$$M_{11} = (m_1 + m_2)a_1{}^2 + m_2 a_2{}^2 + 2m_2 a_1 a_2 \cos\theta_2 \tag{3}$$

$$M_{12} = M_{21} = m_2 a_2{}^2 + m_2 a_1 a_2 \cos\theta_2 \tag{4}$$

$$M_{22} = m_2 a_2{}^2 \tag{5}$$

$$V_{m11} = -m_2 a_1 a_2 \dot{\theta}_2 \sin\theta_2 \tag{6}$$

$$V_{m12} = -m_2 a_1 a_2 (\dot{\theta}_1 + \dot{\theta}_2) \sin\theta_2 \tag{7}$$

$$V_{m21} = m_2 a_1 a_2 \dot{\theta}_1 \sin\theta_2 \tag{8}$$

$$V_{m22} = 0 \tag{9}$$

$$G(q) = \begin{bmatrix} (m_1 + m_2) g a_1 \cos\theta_1 + m_2 g a_2 \cos(\theta_1 + \theta_2) \\ m_2 g a_2 \cos(\theta_1 + \theta_2) \end{bmatrix} \tag{10}$$

$$F(\dot{q}) = \begin{bmatrix} \dot{\theta}_1 + ksgn(\dot{\theta}_1) \\ \dot{\theta}_2 + ksgn(\dot{\theta}_2) \end{bmatrix} \tag{11}$$

The external disturbance is $|\tau_{di}| \leq 1, i = 1{,}2$ is a random noise with the magnitude bounded. The axis representation of the robot manipulator is given in Figure 1. The variable angles are demonstrated with $\theta_1,\ \theta_2$. While $a_1,\ a_2$ indicate the link lengths, $m_1,\ m_2$ show the link weights.

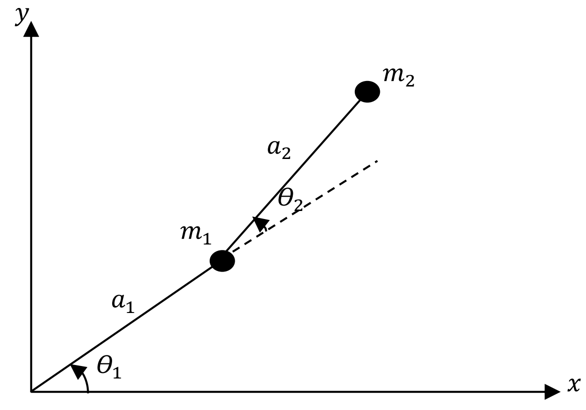

Figure 1. Two-link robot manipulator

2.2. Neural Network Based Adaptive Sliding Mode Control Design

In this chapter, a control system composed of the sliding mode controller, a Radial Based Function (RBF) based neural network descriptor and adaptive control is designed to make a robotic manipulator track the determined trajectory. As mentioned in the introduction part, neural network is used to represent the uncertainty of the robotic manipulator dynamics. In the adaptive control literature, neural networks are widely used to approximate unknown non-linear systems because of its abilities of approximation in the essence of neural networks (Lewis et.al. 2003). The function of the neural network is shown as below:

$$f_1(x) = W^T \phi(X) + \varepsilon(X) \tag{12}$$

In this equation, X represents the input vector, $W \epsilon R^{nx2}$ represents the weight matrices. Each element of W represents the coefficient of the ϕ function. $\varepsilon(X)$ represents the approximation error of the neural network. ϕ is constructed as a vector where, n>1 is the number of the neurons and it is shown as $\phi(X) = [\phi_1(X), \phi_2(X), \dots, \phi_n(X)]^T$. $\phi_i(X)$ shown in the equation (13) is the RBF.

$$\phi_i(X) = \exp\left(\frac{||x - c_i||^2}{\sigma_i{}^2}\right) \quad i = 1,2,\dots,n \tag{13}$$

In equation (13) c_i and σ_i are the center of the neuron and width, respectively.

RBF is expected to approach the equation below.

$$M(q)\dot{s} = M(\ddot{q}_d - \ddot{q} + \Lambda\dot{e}) = M(\ddot{q}_d + \Lambda\dot{e}) - M\ddot{q} \tag{14}$$

$$= M(\ddot{q}_d + \Lambda\dot{e}) - V_m s + V_m(\dot{q}_d + \Lambda e) + G + F + \tau_d - \tau \tag{15}$$

RBF will converge to the $f(X)$ function that will be extracted out of the equation (15).

$$f(X) = M(q)(\ddot{q}_d + \Lambda\dot{e}) + V_m(\dot{q}_d + \Lambda e) + G + F \tag{16}$$

By assuming that the sliding variable will converge to zero after inserting the equation (17) into equation (16), the controller is obtained as below:

$$\tau = f(X) - V_m s + \tau_d \tag{17}$$

After the result of the equation (17) and the error ($\varepsilon(X)$) from the RBF are considered, the controller design takes the following form.

$$\tau = f_1(X) + K_v s + v \tag{18}$$

When selecting this type of a controller, variable v is added in order to eliminate τ_d coming from outside and error $\varepsilon(X)$ coming from the RBF. In this case, K_v adaptive controller coefficient is used to make the system robust. In Eq.19 and Eq.20 v will provide resistance against the noise and error because $f_1(X)$ will converge to $f(X)$. K_v will ensure the stability of the system. The inputs of the $f_1(X)$ are determined as X = [$\ddot{q}_d, \dot{e}, \dot{q}_d, e, q_d$] which can be predicted from the equation (16).

$$v = -bsgn(s) \tag{19}$$

$$sgn(s) = \begin{cases} +1, & s > 0 \\ 0, & s = 0 \\ -1, & s < 0 \end{cases} \tag{20}$$

The Lyapunov function selected for this controller is shown in the equation (21). K_v comes out to be positive from the Lyapunov function.

$$V = \frac{1}{2} s^T M s + \frac{1}{2} tr(W^T {F_w}^{-1} W) \tag{21}$$

where M and F_w are positive matrices. W comes out from the Lyapunov function which is given in equation (22).

$$\dot{W} = F_w \, \phi(X) s^T \tag{22}$$

The variable v shown in equation (19) varies depending on the sliding surface. Adaptive control is used in this system and this controller determines the stability of the system. In this way, the uncertainties coming from the artificial neural network have been stabilized. In addition, the sliding mode controller which is robust against all noisy terms in mathematical model and the error $\varepsilon(X)$ from the RBF is used. The block diagram that summarizes this Neural Network Based Adaptive Sliding Mode Control is given in Figure 2.

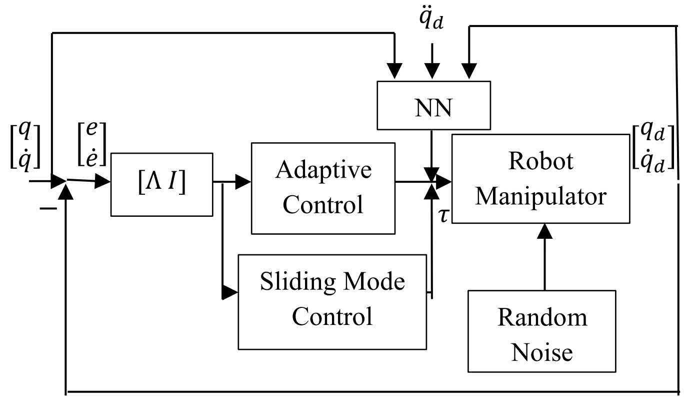

Figure 2. Neural Network Based Adaptive Sliding Mode Control

2.3. Neural Network Based Adaptive Sliding Mode Control Design with Genetic Algorithm

In this paper the genetic algorithm is used to find artificial neural network and sliding mode controller coefficients which are mentioned in previous subsection. The center of the neuron and width (c_i and σ_i respectively) of RBF and accordingly, the weight (W) coefficients of the neural network are approximated by the genetic algorithm. Additionally, the adaptive controller and the sliding mode controller coefficients K_v *and* Λ which affects the performance of overall system are also determined by the genetic algorithm. Genetic algorithm is designed for the purpose of minimizing the error between the reference trajectory and the tracked trajectory. The inputs of the genetic algorithm are trajectory-tracking errors whereas the outputs are generated controller coefficients and neural network parameters.

Genetic algorithms are formed by searching ideal results in a wide space to solve complex problems. For this reason, a population with all these coefficients mentioned above should be created. After the error is minimized, the most appropriate population of the system will be found coefficients.

2.3.1. Population Creation

A random solution group with possible solutions is created. Solution group is called population due to its similarity in biology. In Figure 3, the row formed solutions are called as chromosome. The sum of the chromosome numbers constitutes the population width. The experimenter determines the population size.

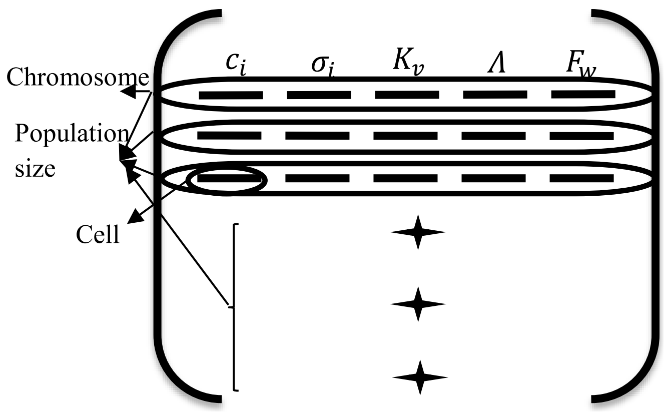

Figure 3. Population Representation

In the population, each coefficient is represented in a column. Coefficients in each row are used as the input for the controller and neural network. The cost function of the overall system is the trajectory tracking error which has to be minimized. The coefficients that are shown in figure 3 are selected randomly between 0 to 100.

After than population becomes exist, the experimenter follows the genetic algorithm scheme shown figure 4. Experimenter applies calculation of suitability value, selection, crossover, mutation respectively. When genetic algorithm reaches the maximum iteration, the optimal solution is approached.

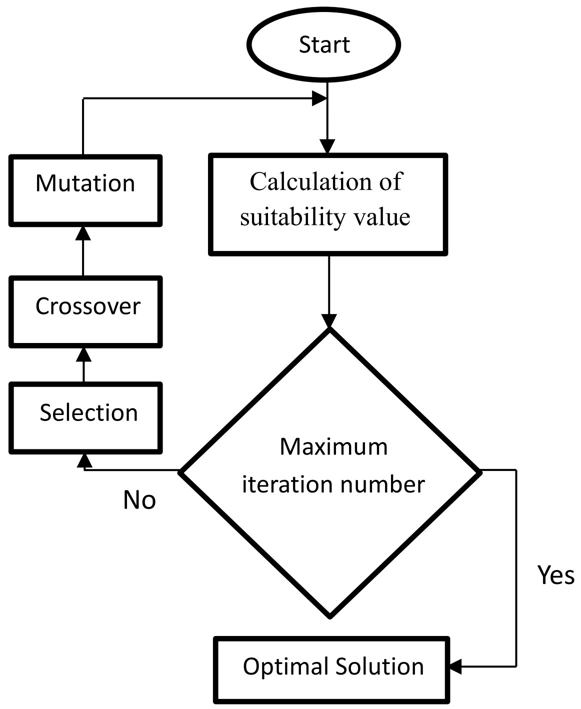

Figure 4. Genetic Algorithm Diagram

2.3.2. Roulette Wheel Selection

The suitability values of each chromosome is calculated, this suitability values that are error of robot manipulator system is vital for survival of the chromosome. According to these suitability values, the probability of survival is examined. In other words, the probability of survival varies according to the suitability value. After the probability of survival of each chromosome is calculated, from the normal probabilities cumulative probabilities are calculated. A random number is generated. If the random number falls within the cumulative probability values, this chromosome will be selected.

2.3.3. Mutation

Based on the initial population of chromosomes, which denote tentative solutions to the optimization problem, new generations are created by applying a randomly change at an arbitrary point in the individual. For each element vector, the mutant vector is produced according to equation (23).

$$p^{g}{}_{i,j} = p^{g-1}{}_{i,j} + rand^{g}{}_{i,j} * (100 - 0) \tag{23}$$

In this equation, the index g indicates the generation to which a vector belongs and p demonstrates the cell number.

2.3.4. Crossover

First, each chromosome in the population is paired with each other. Again, a random number is generated between 0 and 1. If this random number is smaller than the crossover rate, the crossover operation will be performed. While the crossover is performing, the chromosomes that are paired are divided from a random point. Divided pair chromosomes pieces are changed between each other from the point where they divide.

2.3.5. Genetic Algorithm-based Coefficient Calculation

The error term of the system is the input of the genetic algorithm and then the controller's coefficients are created by looking at those error terms. When the smallest error is reached, coefficients that are performed are optimum. The minimized formula is shown in equation (24)

$$s_v = \frac{mean(|e_1|)+mean(|e_2|))}{2} \tag{24}$$

where e_1 and e_2 are error of link 1 and link 2 respectively. The application of the genetic algorithm to the controllers is shown in figure 5.

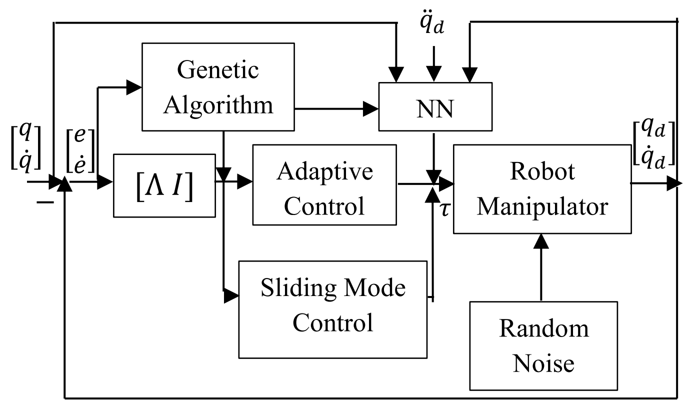

Figure 5. Neural Network Based Adaptive Sliding Mode Control with Genetic Algorithm

3. Results and Discussion

In this work, a novel controller approach was designed for trajectory-tracking control of a 2-link robot manipulator modelled using the Euler Lagrange approach. To verify the effectiveness of the proposed controller, numerical simulations are conducted in MATLAB/Simulink environment. In the first part, neural network-based design is used to model dynamic uncertainties of the robot manipulator. One of the most common smart approaches, the artificial neural network has a natural learning ability. The neural network can approach the nonlinear continuous state with random accuracy. The uncertainties are represented by error term in neural network's formula, so the neural network converges to the mathematical dynamics of the robot manipulator. Adaptive control is a parameter that needs for the system stability. Besides these controllers, the genetic algorithm has been developed to calculate the controller's coefficients that will make the system more stable. In this study, all systems were created on MATLAB / Simulink and simulations were done on this program. In the tables below, robot manipulator parameters and simulation inputs are given respectively.

Table 1. Robot Manipulators Parameters

$\mathbf{m_1}$	weight of link 1	1 kg
$\mathbf{m_2}$	weight of link 2	1 kg
$\mathbf{a_1}$	length of link 1	1 m
$\mathbf{a_2}$	length of link 2	1 m
$\mathbf{g}$	gravitational acceleration	9,8 m/s^2

The initial conditions of the system are considered to be $q(0) = [q_1(0) q_2(0)]^T = 0$, $\dot{q}(0) = [\dot{q}_1(0)\dot{q}_2(0)]^T = 0$. The noise is a random noise with the magnitude bounded.

Table 2. Simulation Parameters

$\mathbf{q_{d_1}}$	Desired trajectory for link 1	$\pi/2 - (\pi/4)\cos(0.5t)$
$\mathbf{q_{d_2}}$	Desired trajectory for link 2	$(\pi/3)sin(0.5t)$
$\boldsymbol{\tau_{d_1}}$	Noise for link 1	$\lvert\tau_{d_1}\rvert \leq 1$
$\boldsymbol{\tau_{d_2}}$	Noise for link 2	$\lvert\tau_{d_2}\rvert \leq 1$
$\mathbf{F(\dot{\theta}_1)}$	Friction term for link 1	$\dot{\theta}_1$+2sign $(\dot{\theta}_1)$
$\mathbf{F(\dot{\theta}_2)}$	Friction term for link 2	$\dot{\theta}_2$+2sign $(\dot{\theta}_1)$

Trajectory-tracking performance of Neural Network based Sliding Mode Adaptive Control (NNSMAC) is demonstrated with the figures given below (Fig. 6-8). For this controller tracking errors of each link and applied control inputs are given in Fig. 7-8 respectively.

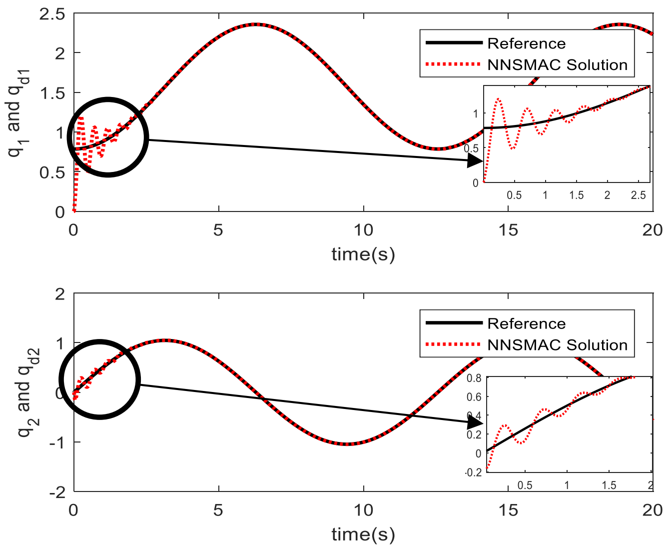

Figure 6. Position tracking link 1 and link 2 for NNSMAC

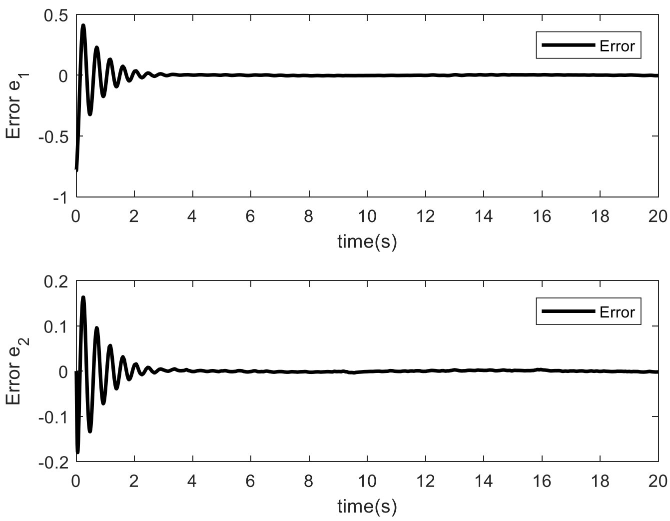

Figure 7. The Tracking Error Link 1 and Link 2 for NNSMAC

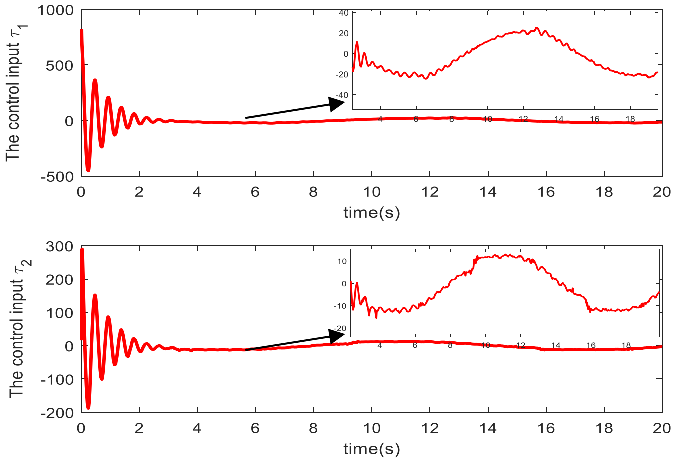

Figure 8. The Control Inputs for NNSMAC

As it can be seen from the figures 6 and 7, although trajectory-tracking performance of NNSMAC is satisfactory in the steady-state phase, the transient response of this controller does not have same satisfactory performance.

The performance of the Neural Network based Sliding Mode Adaptive Control with Genetic Algorithm (NNSMACWGA), which is proposed in this work, represented in figures below (Figs. 9-11). It is clearly seen that both transient and steady responses of the system is enhanced. The oscillations in the transient response was decreased obviously. Moreover, steady state performance was preserved.

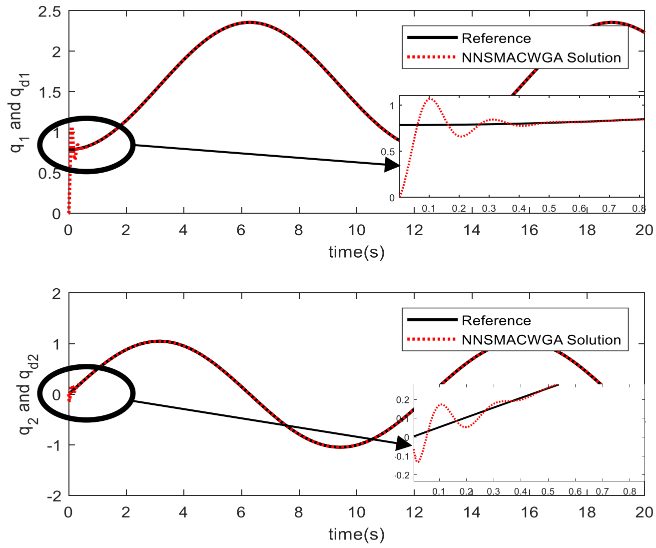

Figure 9. Position Tracking Link 1 and Link 2 for NNSMACWGA

Besides the trajectory-tracking performance improvement, reaching time to reference input was shortened with this novel controller. While reaching to reference input with NNSMAC took around 2 seconds, NNSMACWGA shortened this time to around 0.5 second.

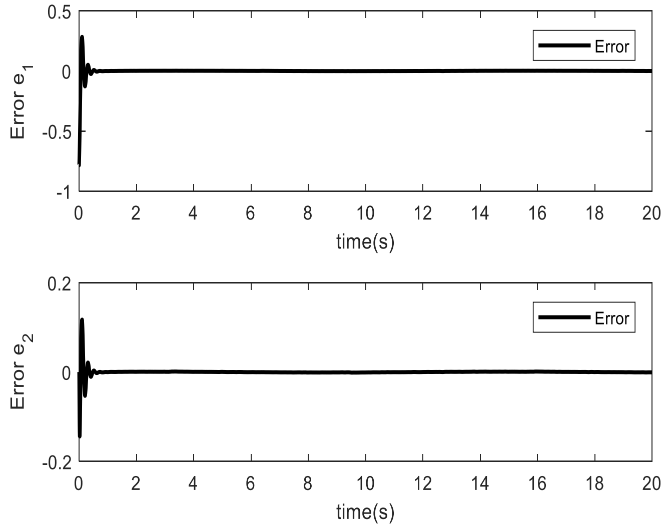

Figure 10. The Tracking Error Link 1 and Link 2 for NNSMACWGA

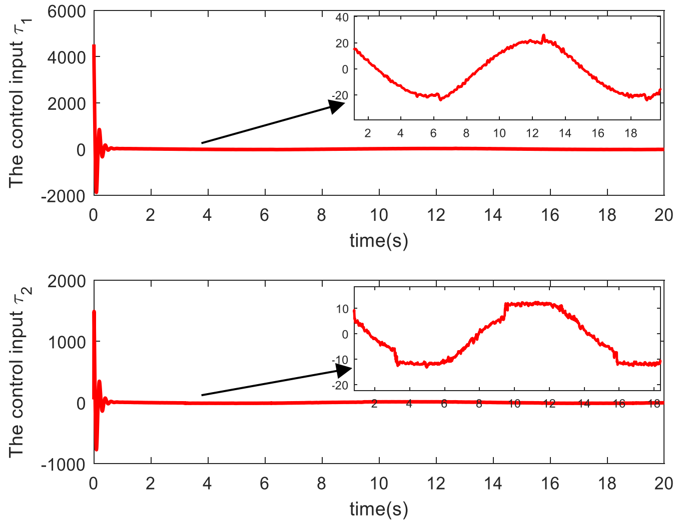

Figure 11. The Control Inputs for NNSMACWGA

The RMS errors of the trajectory tracking in the 1st and 2nd links were applied as the evaluation condition for two different methods that NNSMAC and NNSMACWGA. This RMS error table is shown below.

Table 3. Compare NNSMAC and NNSMACWGA

	Link 1 RMS error	**Link 2 RMS error**
NNSMAC	0,1228	0.0296
NNSMACWGA (population size 300)	0.0728	0,0112
NNSMACWGA (population size 200)	0.0739	0.0110
NNSMACWGA (population size 100)	0.0744	0.0122

4. Conclusion

In this paper, we presented a novel controller approach for robotic manipulator in order to achieve robust trajectory tracking against disturbances and uncertainties. The proposed controller contains neural network part for predict unknown model parameters, adaptive sliding mode controller part for robust trajectory-tracking and genetic algorithm part for calculating the adaptive controller parameters and neural network coefficients. The sliding controller laws are obtained by applying the Lyapunov stability theorem.

To verify the effectiveness of the proposed controller approach we conducted numerical simulations in MATLAB/Simulink environment. Moreover, the obtained results with NNSMACWGA are compared with NNSMAC. The proposed method gives robust trajectory-tracking results under disturbances with bounded error. Both in transient and steady state phase proposed NNSMACWGA controller reveal better performance. The obtained results showed that our controller responses converges to desired trajectory in less time than other method. Besides, chattering phenomena is reduced significantly with NNSMACWGA.

Reference

Amer, A. F., Sallam, E. A., & Elawady, W. M. (2011). Adaptive fuzzy sliding mode control using supervisory fuzzy control for 3 DOF planar robot manipulators. Applied Soft Computing, 11(8), 4943-4953.

Choi, Y., & Chung, W. K. (2004). PID trajectory tracking control for mechanical systems (Vol. 298). Springer Science & Business Media. Lewis, Frank L., Darren M. Dawson ve Chaouki T. Abdallah, Robot manipulator control: theory and practice, CRC Press, 2003.

He, J., Luo, M., Zhang, Q., Zhao, J., & Xu, L. (2016). Adaptive fuzzy sliding mode controller with nonlinear observer for redundant manipulators handling varying external force. Journal of Bionic Engineering, 13(4), 600-611.

Kumar, V., Nakra, B. C., & Mittal, A. P. (2011). A review on classical and fuzzy PID controllers. International Journal of Intelligent Control and Systems, 16(3), 170-181.

Lewis, F. L., Dawson, D. M., & Abdallah, C. T. (2003). Robot manipulator control: theory and practice. CRC Press.

Loucif, F., Kechida, S., & Sebbagh, A. (2020). Whale optimizer algorithm to tune PID controller for the trajectory tracking control of robot manipulator. Journal of the Brazilian Society of Mechanical Sciences and Engineering, 42(1), 1.

Nguyen, K., Nguyen, T., Bui, Q., & Pham, M. (2018). Adaptive antisingularity terminal sliding mode control for a robotic arm with model uncertainties and external disturbances. Turkish Journal of Electrical Engineering & Computer Sciences, 26(6), 3224-3238.

Sun, T., Pei, H., Pan, Y., Zhou, H., & Zhang, C. (2011). Neural network-based sliding mode adaptive control for robot manipulators. Neurocomputing, 74(14-15), 2377-2384.

Tilki, U., & Olgun, M. Robot Manipülatör Kontrolünde Doğrusal ve Doğrusal Olmayan Denetleyici Yapılarının Karşılaştırılması. 21. Otomatik Kontrol Ulusal Toplantısı TOK 2019.

Tran, M. D., & Kang, H. J. (2017). Adaptive terminal sliding mode control of uncertain robotic manipulators based on local approximation of a dynamic system. Neurocomputing, 228, 231-240.

Vijay, M., & Jena, D. (2018). Backstepping terminal sliding mode control of robot manipulator using radial basis functional neural networks. Computers & Electrical Engineering, 67, 690-707.