Robust Control With Fuzzy Based Neural Network For Robot Manipulators

Aşkın Mutlu
Faculty of Engineering, Department of Mechanical Engineering, Istanbul University - Cerrahpaşa, Türkiye
askin@iuc.edu.tr

Abstract: *The utilization of robotic systems is prevalent in various industries, such as defence and automotive, and is commonly utilized in industrial settings. The movements of these systems can be controlled through software programming, allowing for the manipulation of objects and modification of trajectory as desired. However, it is important to exercise caution during these operations as improper manipulation may result in undesired outcomes. As a result, the control of robotic systems has become a crucial aspect in modern industry. The parameters of robotic systems are subject to change based on the loads they carry. Robust control is a method that adapts the control system to accommodate these changes in parameters, thereby maintaining stability and performance. This control method allows for the desired level of control to be maintained even in the presence of changing system parameters. In contrast to traditional robust control methods, robust control utilizes variable parameters with a constant upper limit for parameter uncertainty. Control parameters are updated over time using cosine and sine functions, however, determining appropriate values for these parameters can be challenging. To address this issue, a neural network model utilizing fuzzy logic compensator is employed to continuously calculate the appropriate control parameter values. The effectiveness of this proposed control method is demonstrated through graphical representation.*

Keywords: Robot manipulators, neural network control, robust control, fuzzy logic, adaptive control.

1. Introduction

Previous research on the adaptive control for robot manipulators has resulted in the development of parameter estimation laws by Slotine et al. [1] and Sciliano et al. [2]. Building on these works, Spong [3] proposed a robust control law. However, this robust control law [3] is susceptible to chattering and large tracking error when there is uncertainty in the system's parameters. To overcome this, it is essential to determine an appropriate upper uncertainty bound limit.

Burkan and Askin [4] aimed to improve upon the previously proposed robust control method by designing an uncertainty estimation algorithm for the robust controller [3] algorithm based on the Lyapunov theory, thereby ensuring the stability of the uncertain system. The control parameters were then estimated using trigonometric functions such as Cosine and Sine. However, determining the appropriate values for these fixed control parameters is challenging. To overcome this, a fuzzy logic control based compensator was designed and the effects on the robot's tracking errors were investigated.

Fuzzy logic control has been studied extensively by various researchers. Its advantages include the ability to handle uncertainty, time-varying and complex systems, and the ability to incorporate the expertise of system control experts. Furthermore, it can be applied to non-linear and unknown mathematical model systems. Following the work of Zadeh [5] and Mamdani [6], many research groups have undertaken studies in this area. Fuzzy sets have found application examples in a wide range of areas, from control systems to various estimation methods [7]. A fuzzy logic controller can be implemented directly on a system [8] as well as to control its various parameters, with the goal of enhancing the overall performance of the controller [9].

Neural networks have been widely studied and applied in various fields, including control systems. They are particularly useful in applications where the dynamics of the system are complex or unknown, and traditional control methods may not be effective. Neural network controllers (NNC) are designed to approximate the unknown dynamics of the system and provide accurate control actions [10]. In the present study, the coefficients of the robust control system were treated as variables and their optimal values were determined through the utilization of a neural network trained using a fuzzy logic-based algorithm.

2. Control Strategy

The dynamic model of an n-joint manipulator, in the absence of friction or other disturbance effects, can be represented mathematically as [3].

$$\mathrm{M}(\theta)\ddot{\theta}+\mathrm{C}(\theta,\dot{\theta})\dot{\theta}+\mathrm{G}(\theta)=\tau \qquad \mathrm{Y}(\theta,\dot{\theta},\ddot{\theta})\pi=\tau \tag{2.1}$$

The dynamic model of the n-joint manipulator in the absence of friction or other disturbance effects can be represented by a constant p-dimensional vector of robot parameters (π), generalized coordinates (θ), an n-dimensional vector of applied torques or forces (τ), an nxn symmetric positive definite inertia matrix (M), an n-dimensional vector of centripetal and coriolis terms $\mathrm{C}(\theta,\dot{\theta})\dot{\theta}$, and an n-dimensional vector of gravitational terms (G(θ)). The control parameters are defined as

$$\tilde{\theta}=\theta-\theta_d\,;\,\dot{\theta}_{\mathrm{r}}=\dot{\theta}_{\mathrm{d}}-\Lambda\tilde{\theta}\,;\,\sigma=\dot{\theta}-\dot{\theta}_{\mathrm{r}}=\dot{\tilde{\theta}}+\Lambda\tilde{\theta} \tag{2.2}$$

The control law, based on the control parameters defined in equation (2.2), is presented in reference [4].

$$\begin{aligned}\tau &=\tau_0+\mathrm{Y}(\theta,\dot{\theta},\dot{\theta}_{\mathrm{r}},\ddot{\theta}_{\mathrm{r}})(\mathrm{u}_1+\mathrm{u}_2)\\ &=\mathrm{Y}(\theta,\dot{\theta},\dot{\theta}_{\mathrm{r}},\ddot{\theta}_{\mathrm{r}})(\pi_0+\mathrm{u}_1+\mathrm{u}_2)-\mathrm{K}\sigma\end{aligned} \tag{2.3}$$

where u_2 and u_1 are supplementary control inputs that are implemented to enhance the robustness of the system against parametric uncertainty, allowing for improved stability and performance of the control system in the face of changes in the system's parameters.

$$\begin{aligned}&\mathrm{M}(\theta)\dot{\sigma}+\mathrm{C}(\theta,\dot{\theta})\sigma+\mathrm{K}\sigma\\ &=\mathrm{Y}(\theta,\dot{\theta},\dot{\theta}_{\mathrm{r}},\ddot{\theta}_{\mathrm{r}})(\pi_0-\pi)+\mathrm{u}_1+\mathrm{u}_2)+\mathrm{K}\sigma\\ &=\mathrm{Y}(\theta,\dot{\theta},\dot{\theta}_{\mathrm{r}},\ddot{\theta}_{\mathrm{r}})(\tilde{\pi}+\mathrm{u}_1+\mathrm{u}_2)+\mathrm{K}\sigma\end{aligned} \tag{2.4}$$

The function $(\beta^2/\alpha)\cos(\int\alpha Y^T\sigma dt)\sin(\int\alpha Y^T\sigma dt)$ is employed as a parameter estimation law. In order to derive the control law, a Lyapunov function candidate is defined as follows

 WEB ISSN 2534-997X; PRINT ISSN 2534-8582

$$V = \frac{1}{2}\sigma^T M(\theta)\sigma + \frac{1}{2}\tilde{\theta}^T B\tilde{\theta} + \frac{1}{2}\hat{\pi}^T \Phi^2 \hat{\pi}\ ;\ \ V \geq 0 \qquad (2.5)$$

where $B \in R^{nxn}$ is a positive diagonal matrix of size nxn, Φ is chosen as a pxp dimensional diagonal matrix that varies over time.

The time derivative of V along the system (2.5) is

$$\dot{V} = \sigma^T M(\theta)\dot{\sigma} + \sigma^T \frac{1}{2}\dot{M}(\theta)\sigma + \tilde{\theta}^T B\dot{\tilde{\theta}} \qquad (2.6)$$
$$+\hat{\pi}^T \Phi^2 \dot{\hat{\pi}} + \hat{\pi}^T \Phi\dot{\Phi}\hat{\pi}$$

As a result, the time derivative of the Lyapunov function is written as.

$$\dot{V} = -\dot{\tilde{\theta}}^T K\dot{\tilde{\theta}} - \tilde{\theta}^T \Lambda^T K\Lambda\tilde{\theta} + \sigma^T Y(u_1 + u_2) \qquad (2.7)$$
$$+\sigma^T Y\tilde{\pi} + \sigma^T Y\hat{\pi}$$

Control parameters are defined in Equation (2.6)

$$\dot{V} = -\dot{\tilde{\theta}}^T K\dot{\tilde{\theta}} - \tilde{\theta}^T \Lambda^T K\Lambda\tilde{\theta} + \sigma^T Yu(t)_2 + \sigma^T Y\tilde{\pi} \qquad (2.8)$$

The rest of the proof is given in [3].

The fuzzy logic controller implemented in this study has two inputs (e_1, e_2) and two outputs (α,β). The inputs to the controller are the orbital tracking error values for the first and second manipulator (e_1, e_2), whereas the outputs are the torque values to be applied to the joints. These torque values are used to calculate the values of the controller's parameters, α ([0.5;1.5]) and β ([1;15]) These parameters are then used to adjust the system's dynamics and achieve the desired control performance.

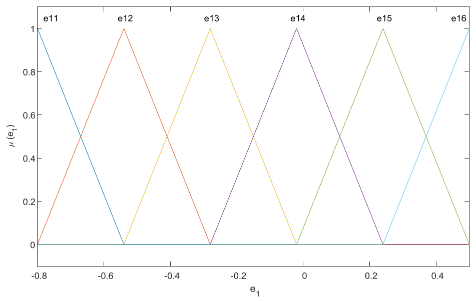

Figure 2.1 *Membership function for tracking error of first manipulator (Input value e_1 : [-0.8 ; 0.5])*

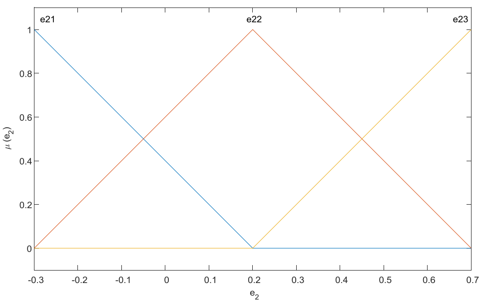

Figure 2.2 *Membership function for tracking error of second manipulator (Input values e_2 : [-0.3 ; 0.7])*

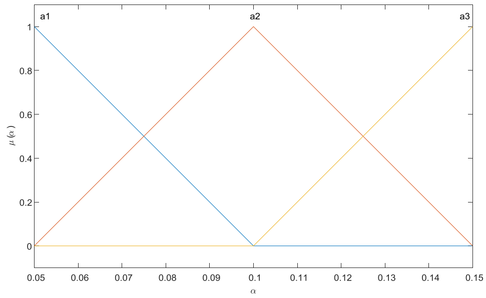

Figure 2.3 *Membership function of α:*

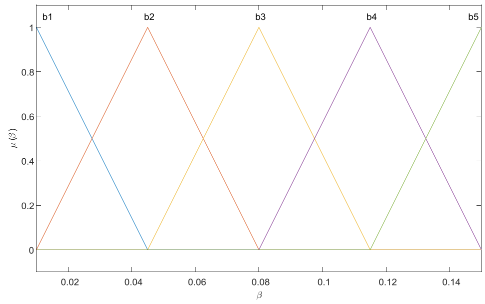

Figure 2.4 *Membership function of β*

Table 1*: Rule table for fuzzy logic*

e2 \ e1 (α / β)	e11	e11	e13	e14	e15	e16
e21	α1 β5	α3 β5	α2 β2	α2 β5	α2 β5	α2 β5
e22	α2 β4	α2 β4	α3 β3	α3 β5	α1 β4	α3 β4
e23	α3 β3	α1 β1	α1 β4	α1 β5	α3 β3	α3 β1

Using (If ... and ... Then ...) structure of the rule table is created.

In this study, a defuzzification method known as Centroid was employed to convert the fuzzy output of the fuzzy logic controller into a crisp value. The values of the controller's parameters, α and β, were then obtained and added to the system to improve its control performance and reduce the error found by the fuzzy logic controller.

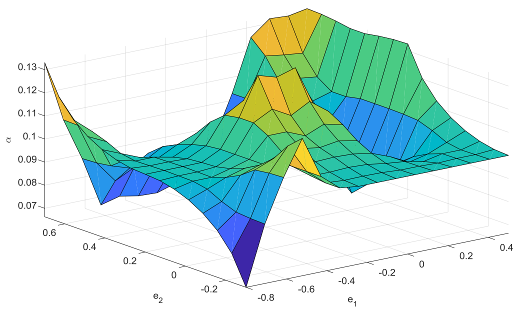

Figure 2.5 *Fuzzy Surface Model α(e_1,e_2)*

 YEAR VIII, ISSUE 2, P.P. 42-46 (2023)

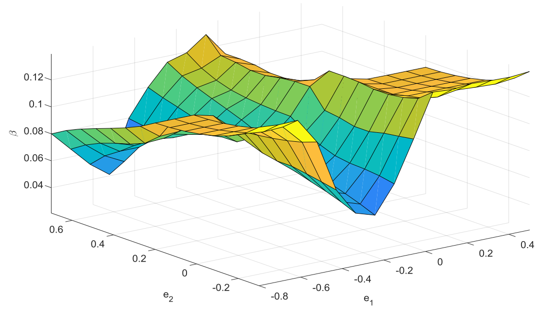

Figure 2.6 *Fuzzy Surface Model β(e₁,e₂)*

3. Neural Network Control

The efficacy of traditional control methodologies is dependent upon the veracity of the dynamic model of the system. However, the attainment of a precise mathematical model for complex systems with multiple variables, such as robotic manipulators, is often a challenging task. As a result, model-based control approaches may be inadequate in achieving the desired control performance and precision. To address this issue, a model-free intelligent controller based on fuzzy set theory, specifically fuzzy logic control, is proposed as a solution. The fuzzy logic controller employs appropriate linguistic fuzzy rules, derived from an operator's control experience and a database, for decision-making processes. These rules simulate human cognition through the application of fuzzy logic and fuzzy set operations.

Despite the advantages of fuzzy logic control, it presents inherent difficulties in implementation. Specifically, the definition of membership functions for linguistic variables exhibits a high degree of autonomy, and determining appropriate functions through trial and error can be laborious. Additionally, finding appropriate fuzzy logic control rules can also prove difficult. As an alternative, a neural network control strategy is proposed, as it possesses the ability to learn and adapt to the system's characteristics.

This study utilizes a multilayer feedforward neural network as the primary architecture due to its widespread use and versatility in various applications. The backpropagation learning algorithm is employed to adjust the weighting of the network, allowing it to learn and adapt in a manner similar to the neural networks of the human brain. The multilayer feedforward neural network consists of multiple processing elements connected through weighted data connections. The strength of these connections is determined by the weighting values. The activation value of each processing element is calculated by summing the input signals, each multiplied by their corresponding weighting values as described in reference [10].

$$net_j^k = \sum W_{ji}^k O_i^{k-1} \tag{3.1}$$

$$O_j^k = f\left(net_j^k\right) \tag{3.2}$$

where net_j^k is net input function, W_{ji}^k is the weighting, O_i^{k-1} is the output, and $f(*)$ activation function.

The training of the neural network utilizes the backpropagation algorithm, which adjusts the weight values in the network based on the discrepancy between the actual output and the desired output. The performance of the network is measured by an objective function, defined as the error between the actual output and the desired output. The objective of the training process is to minimize this error through adjustments of the weight values, thereby enabling the network to adapt to and learn the characteristics of the system. The object function defined as

$$E = \frac{1}{2}\sum_j \left(\theta_{dj} - O_j\right)^2 \tag{3.3}$$

The utilization of the steepest descent optimization method for the modification of the weighting values in the neural network, with the aim of minimizing the objective function, allows for the determination of the correction value for the weighting. It can be obtained

$$\Delta W_{ji}^k = \mu \delta_j^k O_i^{k-1} \tag{3.4}$$

$$\delta_j^k = -\frac{\partial E}{\partial net_j^k} \tag{3.5}$$

where μ is learning rate parameter, so

$$\delta_j^k = f\left(net_j^k\right) \sum_l \delta_l^{k+l} O_{lj}^{k+l}$$

The activation functions used for the neural network are a linear function, $f(x) = x$, for the output layer and a sigmoid function for the hidden layer. The linear function is chosen for the output layer as it allows for direct computation of the output, while the sigmoid function is used for the hidden layer as it allows for non-linearity and differentiability, which are important properties in neural networks.

$$f(x) = \frac{1}{(1+e^{-\mu x})} \tag{3.6}$$

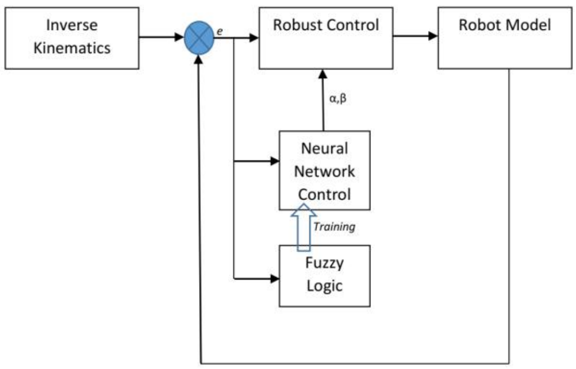

Figure 3.1 *Block diagram of the proposed neural network based on fuzzy-robust controller.*

In this study, as shown in Figure 3.1, alpha and beta parameters of the robust controller were determined by a neural network controller trained with fuzzy logic.

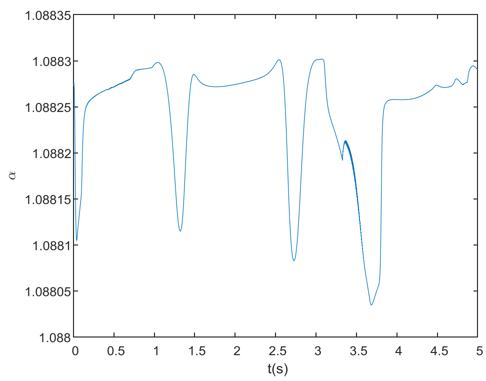

Figure 3.2 *Changing α parameter with time*

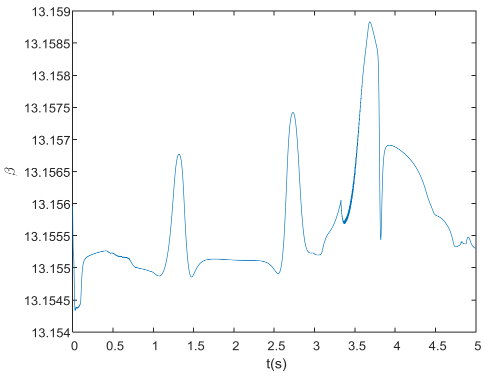

Figure 3.3 *Changing β parameter with time*

4. Simulation Results

To evaluate the effectiveness of the proposed control law, it was applied to a two-linked manipulator system. The performance of the controller was analyzed by monitoring the system's response to various inputs and evaluating the control performance metrics.

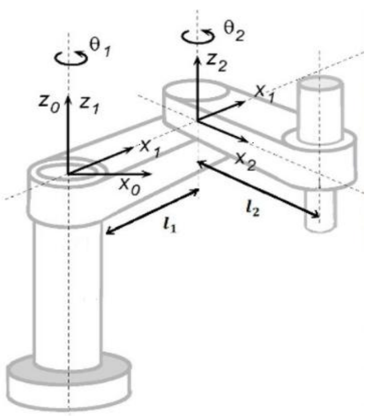

Figure 4.1 *Two-Link Robot Arm [3,4].*

Robot parameters are given as follows.

$$\begin{aligned} \pi_1 &= m_1 l_{c1}^{\ 2} + m_2 l_1^{\ 2} + I_1 \\ \pi_2 &= m_2 l_{c2}^{\ 2} + I_2 \\ \pi_3 &= m_2 l_1 l_{c2} \\ \pi_4 &= m_1 l_{c1} \\ \pi_5 &= m_2 l_1 \\ \pi_6 &= m_2 l_{c2} \end{aligned} \tag{4.1}$$

$$\mathrm{M}(\theta) = \begin{bmatrix} \pi_1 + \pi_2 + 2\pi_3 \cos(\theta_2) & \pi_2 + \pi_3 \cos(\theta_2) \\ \pi_2 + \pi_3 \cos(\theta_2) & \pi_2 \end{bmatrix} \tag{4.2}$$

$$\mathrm{C}(\theta, \dot{\theta}) = \begin{bmatrix} -\pi_3 \sin(\theta_2)\dot{\theta}_2 & -\pi_3 \sin(\theta_2)(\dot{\theta}_1 + \dot{\theta}_2) \\ \pi_3 \sin(\theta_2)\dot{\theta}_1 & 0 \end{bmatrix} \tag{4.3}$$

$$\mathrm{G} = \begin{bmatrix} \mathrm{g}(\pi_4 + \pi_5)\cos(\theta_1) + \mathrm{g}\pi_6 \cos(\theta_1 + \theta_2) \\ \mathrm{g}\pi_6 \cos(\theta_1 + \theta_2) \end{bmatrix} \tag{4.4}$$

To simulate the proposed control law, a specific trajectory was chosen for each joint of the two-linked manipulator system. The trajectory used in the simulation was 0.5cos (0.5πt)-0.5.

The simulation was performed under the same conditions and using the same trajectory for both the proposed control law and a comparison method. The results of the simulation are presented in Figures 4.2-4.3, which compare the performance of the proposed control law with the comparison method.

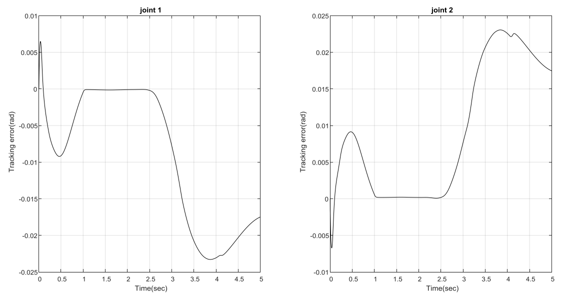

Figure 4.2 Tracking error for robust controller [3]

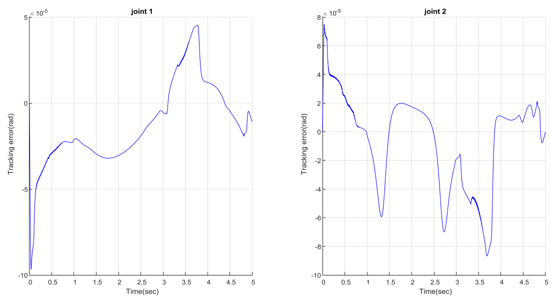

Figure 4.3 *Tracking errors for robust control with fuzzy based neural network*

As depicted in Figure 4.3, the proposed control law utilizing a neural network controller, was able to effectively reduce the tracking error of the two-linked manipulator system. The neural network controller, through the use of α and β, adapts the system's dynamics in real-time by updating the values of α and β as the simulation progresses. This results in the selection of the most appropriate values of α and β, leading to a reduction in the tracking error of the system.

As a result of the proposed control law, a very small tracking error was obtained for the two-linked manipulator system. The performance of the proposed controller was compared with a known robust controller, and the results are presented in Figure 4.4. As can be seen from the figure, the proposed controller provides superior performance when compared to the known robust controller. The results show that the proposed controller is able to achieve a much smaller tracking error compared to the known robust controller.

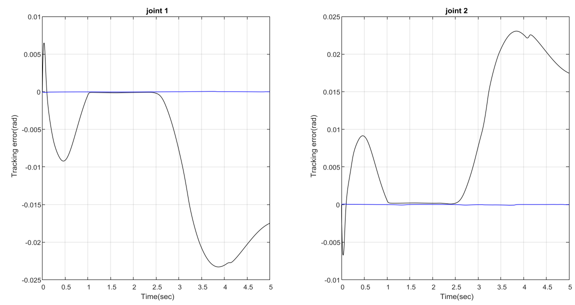

Figure 4.4 *Comparison of orbital tracking errors*

4. Conclusion

Computer simulations were conducted using the same model and trajectory. The effect of control parameters, α and β, on control performance were evaluated by keeping the values of K and constant while varying α and β. The results, depicted in figures 4.2 and 4.3, indicate that the tracking error of the robust control is substantial. In contrast, the proposed adaptive control algorithm effectively reduces the tracking error. It was observed that α and β values have a significant impact on tracking error, however, these values are fixed, making it difficult to select optimal values. Furthermore, the tracking error may not always be improved with these fixed values of α and β. Therefore, the proposed control algorithm was developed to enhance the control law. As depicted in the figures, the tracking error is significantly reduced through the implementation of the neural network controller.

5. References

1. SLOTINE, J.J., LI, W., 1987, On the adaptive control of robotic manipulator, The International Journal of Robotics Research, 6 (3), 49-59.

2. SCIAVICCO, L., SICILIANO, B., 1996, Modeling and Control of Robot Manipulators, The McGraw-Hill Companies, Great Britain,1-85233-221-2.

3. SPONG,M.W., 1992, On the robust control of robot manipulators, IEEE Transactions on Automatic Control, 37 (11), 1782-1786.

4. MUTLU A., BURKAN R., 2017, Fuzzy-Robust Control Of a Two-Link Robot Arm, International Journal for Science, Technics and Innovations for the Industry. 5, 214-217.

5. ZADEH, L.A., 1965, Fuzzy sets, Information and control, 8, 338-353.

6. MAMDANI, E.H., 1974, Application of fuzzy algorithms for control of simple dynamic plant, IEEE Proc., 121 (12), 1585-1588.

7. ZOR, C., ÇEBİ, F., 2018, Demand prediction in health sector using fuzzy grey forecasting, Journal of Enterprise in Information Management, 31 (6), 937-949.

8. ARIK, A.E., BİLGİÇ, B., 2023, Fuzzy control of a landing gear system with oleo-pneumatic shock absorber to reduce aircraft vibrations by landing impacts, Aircraft Engineering and Aerospace Technology, 95 (2), 284-291.

9. BURKAN, R., MUTLU, A., 2021, Robust control of robot manipulators with an adaptive fuzzy unmodelled parameter estimation law, Robotica, 40 (7), 2365-2380.

10. HUANG, S.J., HU, C.F., 1996, Neural Network Controller for Robotic Motion Control, Int J Adv Manuf Technol, 12, 450-454.