Kırklareli Üniversitesi
Mühendislik ve Fen Bilimleri Dergisi
Cilt 9, Sayı 2, 265-280, 2023

Kırklareli University
Journal of Engineering and Science
Volume 9, Issue 2, 265-280, 2023

Research Paper DOI: 10.34186/klujes.1315919

Simulation of Kinematic and Dynamic Model of Two Degrees of Freedom Robot Arm Using Simulink

Aytuğ Becenen[1*], Gürkan Tuna[2]

[1] *Trakya University, Institute of Natural and Applied Sciences, Edirne, Turkiye*
[2] *Trakya University, Edirne Vocational College of Technical Sciences, Edirne, Turkiye*

Received: 17.06.2023, Accepted: 20.12.2023, Published: 31.12.2023

ABSTRACT
The aim of this study is to realize the physical model of a robot arm with two degrees of freedom in the MATLAB Simulink environment, to show that the physical model of the robot arm matches the mathematical model and that the results of these models match. For this purpose, kinematic and dynamic modeling of a two-degree-of-freedom robotic arm is performed in the MATLAB Simulink environment. To ensure the accuracy of the physical model created in the MATLAB Simulink environment, mathematical equations are defined to a function within the physical model, in other words, a mathematical model is created. The kinematic equations of the robotic arm are examined in two steps as forward kinematics and inverse kinematics equations, and the geometric approach is used when obtaining the kinematic equations. The Lagrange-Euler method is used to obtain dynamic equations. The results obtained show that the mathematical kinematic and dynamic equations match the physical model in the MATLAB Simulink environment and reveal that the system is reliable.

Keywords: Forward kinematics; Inverse kinematics; Lagrange-Euler method; Robot arm

İki Serbestlik Dereceli Robot Kolunun Kinematik ve Dinamik Modelinin Simulink Kullanılarak Benzetimi

ÖZ
Bu çalışmada amaç iki serbestlik derecesine sahip bir robot kolun fiziksel modelinin MATLAB Simulink ortamında gerçeklemesinin yapılması, robot kolun fiziksel modeli ile matematiksel modelinin uyuştuğunu ve bu modellerin sonuçlarının eşleştiğini ortaya koymaktır. Bu amaç doğrultusunda, MATLAB Simulink ortamında iki serbestlik dereceli bir robot kolun kinematik ve dinamik modellemesi yapılmaktadır. MATLAB Simulink ortamında oluşturulan fiziksel modelin doğruluğundan emin olmak için fiziksel modelin içinde bir fonksiyona matematiksel denklemler tanımlanmaktadır, diğer bir ifadeyle matematiksel bir model oluşturulmaktadır. Robot kolun kinematik denklemleri ileri kinematik ve ters kinematik denklemleri olarak iki adımda incelenmekte olup, kinematik denklemler elde edilirken geometrik yaklaşımdan faydalanılmaktadır. Dinamik denklemler elde edilirken ise Lagrange-Euler metodu kullanılmaktadır. Elde edilen sonuçlar, matematiksel kinematik ve dinamik denklemlerin MATLAB Simulink ortamındaki fiziksel model ile eşleştiğini göstermekte ve sistemin güvenilir olduğunu ortaya koymaktadır

Anahtar Kelimeler: İleri kinematik; Ters kinematik; Lagrange-Euler metodu; Robot kol

**Corresponding Author: Aytuğ BECENEN
aytugbecenen@gmail.com*

1. INTRODUCTION

Modelling with simulation enables problems to be prevented before they occur in real life to be solved more easily and costless. To put it more clearly, system is necessary to ensure that it works correctly and reliably before a it is implemented, in this way saving time and costs. In this study, it is aimed to demonstrate that the physical model of the robot arm system matches the mathematical model and the accuracy of the physical model to be used before proceeding with the controller design. Therefore, in this study, the compatibility of the physical model and the mathematical model of the two degrees of freedom robot arm is compared in MATLAB Simulink environment and it is ensured that the mathematical and physical model outputs of the designed robot arm system match. Firstly, a robot arm with two degrees of freedom was designed in Solidworks solid drawing program. This robot arm designed in Solidworks solid drawing program was transferred to MATLAB Simulink environment. In this way, the basic block diagram schematic required for the creation of the physical model of the system was obtained. Then, mathematical calculations of the forward kinematic equations and inverse kinematic equations of the system were performed. After kinematic calculations, dynamic equation mathematical calculations of the robot arm system were performed. Mathematical calculations of kinematic and dynamic models are explicated under the heading of materials and methods. Kinematic and dynamic calculations were coded into function blocks in MATLAB Simulink environment and thus mathematical models of the system were created. The creation of the physical model was obtained by creating the kinematic and dynamic models with block diagrams on the basic block diagram scheme obtained by transferring the robot arm system from Solidworks environment to MATLAB Simulink environment. Physical and mathematical model outputs are described in the findings and discussion section. The results of the study are evaluated in the last section.

2. LITERATURE REVIEW

Robot arm systems are systems that operate in many fields today. There are many types of robotic arms for different areas and different needs. Robot arms can be generally analyzed under three headings: industrial robots, mobile robots and robots with haptic systems (Yıldırım, 2021). However, robot arms can be classified in many ways according to their joint structures, usage areas, features, controller type and methods used in their systems or working principles (Gürgüze & Türkoğlu, 2019). Therefore, robotic arms have been widely investigated in the literature. There are many academic studies on topics such as robot arm kinematics, robot arm dynamics, controller design for robot arm systems, and the use of artificial intelligence algorithms with controllers. And what these studies have in common is the frequent use of modeling and simulation techniques.

In (Sahay, Chattopadhyay, & Chowdhury, 2020), a solution for the dynamic equations of a robot manipulator with two degrees of freedom was presented. The solution gives graphs showing how different torque values applied to the joints of the robot arm affect the angular position and angular velocity of each joint.

In (Kebria, Al-wais, Abdi, & Nahavandi, 2016), a different mathematical model for the kinematics and dynamics of one of the robot arm models, UR5, was presented. The study was developed in MATLAB Simulink environment and the SimMechanics model of the robot was created. According to the authors of this research, this model is the most accurate kinematic and dynamic model for this robot to this day.

In (Ghaleb & Aly, 2018), modeling, simulation and control of a robot arm with two degrees of freedom were studied. Forward kinematics, inverse kinematics and dynamic equations are investigated and then a control design using a PID controller is carried out. The modeling is based on MATLAB Simulink.

In (Okubanjo, Oyetola, Osifeko, Olaluwoye, & Alao, 2017), mathematical modeling, control and simulation of a two degrees of freedom robot arm were presented. The robot arm was simulated using MATLAB Simulink and then controlled with a PID controller to reach and maintain a desired joint angle position.

In (Mahto, Kaur, & Jain, 2022), MATLAB Simulink software was used to simulate a robotic arm created in Solidworks environment. Tests of this robotic arm were performed for different input variables and changes in outputs were observed.

In (Zhang, Ma, & Zhang, 2022), the XB7 robotic arm was modelled in MATLAB Simulink environment and dynamic tests were performed. In this study, Newton Euler equation was used for inverse dynamic modeling method and simulation analysis of robotic arm dynamics was performed in Simulink environment.

In (Vivekanandan, Vishnu, Narayanasamy, & Yasodharan, 2019), mathematical modeling and kinematic analysis were performed for the VI5SNS robot arm model. The mathematical modeling of the robot arm was done with the help of the Denavit Hartenberg (D-H) approach. The kinematic equations were generated with the help of Jacobian matrix. Direction planning responses are generated and tested by means of a simulation program.

In (Aydın, 2022), a solid model of the slider-crank mechanism was made in Solidworks and then simulated with MATLAB/Simmechanics. The slider-crank mechanism has a single degree of freedom and its position control was realized with the help of a PID controller and the results were presented.

In (Aydın, 2022), real-time control of a three-degree-of-freedom RRR-type robot arm was realized. The robot arm was designed using Solidworks and modelled in the MATLAB Simulink environment. As input to the robot arm, the coordinates that the end point of the robot arm desired to reach were given and the angle values required for these coordinate values were calculated.

In (Aydın, 2022), the speed control of a slider crank mechanism with one degree of freedom was performed. The system using PID controller was simulated in the MATLAB Simulink environment and the results were presented.

3. MATERIAL AND METHOD

In this section, it is explained how the kinematic and dynamic equations of the robot arm with two degrees of freedom are constructed. The schematic representation of the robot arm is given in Figure 1 and the physical parameters of the robot arm system are given in Table 1.

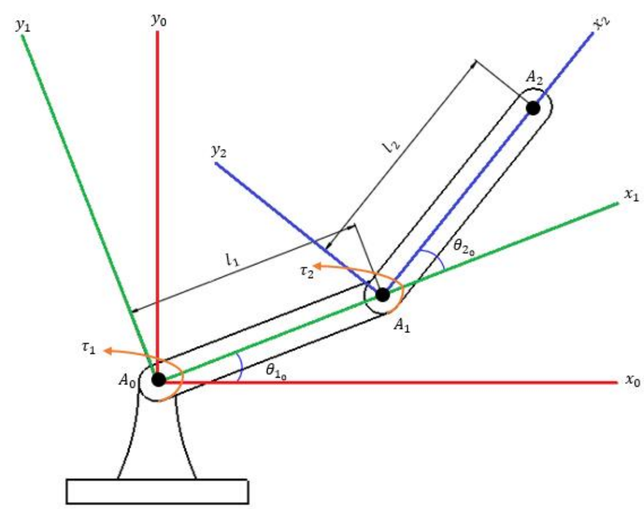

Figure 1: Two degrees of freedom robot arm.

Table 1: Physical Characteristics of the System

Parameters	Symbols	Magnitude	Unit
Length of the link 1	l_1	0.2	Meter
Length of the link 2	l_2	0.2	Meter
Weight of link 1	m_1	0.10060	Kg
Weight of link 2	m_2	0.10139	Kg
Moment of inertia of link 1	I_1	427.226*10^{-6}	Kgm^2
Moment of inertia of link 2	I_2	435.274*10^{-6}	Kgm^2

3.1. Robot Arm Kinematics

Robot arm kinematics refers to the relationship between the angles of the robot arm joints and the end point of the robot arm (Barakat, Gouda, & Bozed, 2016), (Küçük & Birgül, 2006). Calculating the endpoint of the robot arm by knowing the angle applied to the joints is called forward kinematics, and calculating the angle to be applied for this coordinate by knowing the position of the robot endpoint is called inverse kinematics. There is more than one way to calculate the kinematic equations (Küçük & Bingül, 2004). In this study, geometric method is preferred to calculate the kinematic equations. The forward kinematics equations that allow finding the coordinates of the robot's endpoint in a system with two degrees of freedom are given in equations (1) and (2).

$$x = l_1 * \cos\theta_1 + l_2 * \cos(\theta_1 + \theta_2) \tag{1}$$

$$y = l_1 * \sin\theta_1 + l_2 * \sin(\theta_1 + \theta_2) \tag{2}$$

The inverse kinematic equations derived from the forward kinematic equations are given in equations (3) and (5).

$$\theta_1 = atan2\left(K/p, \sqrt{1 - \left(\frac{K}{p}\right)^2} \right) \pm \operatorname{atan2}(x, y) \tag{3}$$

$$\frac{x^2+y^2+(l_1)^2-(l_2)^2}{2l_1} = K, K = p * \sin\big((\theta_2 + \theta_1) + \theta_1\big) \tag{4}$$

$$\theta_2 = atan2(y - l_1 * sin\theta_1, x - l_1 * cos\theta_1) - \theta_1 \tag{5}$$

The derivatives of the forward kinematic and inverse kinematic position equations give the velocity equations. Forward kinematic velocity equations enable the linear velocity of the robot arm endpoint to be found, and inverse kinematic velocity equations allow the angular velocity applied to the joints to be found. In this study, Jacobian matrix was used to find the kinematic velocity equations. The forward kinematic velocity equations of the robot arm with two degrees of freedom are given in equation (6) and the inverse kinematic velocity equations are given in equation (7).

$$V = \begin{bmatrix} V_x \\ V_y \end{bmatrix} = \begin{bmatrix} -l_1 * \sin\theta_1 - l_2 \sin(\theta_2 + \theta_1) & -l_2 * \sin(\theta_2 + \theta_1) \\ l_1 * \cos\theta_1 + l_2 \cos(\theta_2 + \theta_1) & l_2 * \cos(\theta_2 + \theta_1) \end{bmatrix} * \begin{bmatrix} \dot\theta_1 \\ \dot\theta_2 \end{bmatrix} \tag{6}$$

$$\begin{bmatrix} \dot\theta_1 \\ \dot\theta_2 \end{bmatrix} = \begin{bmatrix} -l_1 * \sin\theta_1 - l_2 \sin(\theta_2 + \theta_1) & -l_2 * \sin(\theta_2 + \theta_1) \\ l_1 * \cos\theta_1 + l_2 \cos(\theta_2 + \theta_1) & l_2 * \cos(\theta_2 + \theta_1) \end{bmatrix} * \begin{bmatrix} V_x \\ V_y \end{bmatrix} \tag{7}$$

3.2. Robot Arm Dynamics

Robot arm dynamics refers to the equations of motion of the robot arm. Thanks to the equations of motion, it can be calculated how the position, velocity and acceleration of the robot arm change with the force or torque applied to the joints (Hüseyinoğlu & Abut, 2018).

In this study, the Lagrange Euler (L-E) equation was used to establish the equations of motion. The L-E equation is a method of generating the equations of motion from a scalar function obtained by subtracting the potential energies from the kinetic energy of a mechanical system (Craig, 2014). The L-E equation is obtained by taking the difference between the total kinetic energy and the total potential energy of the system. The total kinetic energy of the system is obtained by calculating and summing the individual kinetic energies of all links and the total potential energy of the system is obtained by calculating and summing the individual potential energies of all links. The L-E equation for the robot arm system with two degrees of freedom is given in equations (8-10).

$$L = \sum T - \sum U \tag{8}$$

$$\sum T = \frac{1}{2}\left[m_1 {l_{g_1}}^2 + I_1 + m_2 {l_1}^2\right]{\dot{\theta}_1}^2 + \frac{1}{2}\left[m_2 {l_{g_2}}^2 + I_2\right]\dot{\emptyset}^2 + m_2 l_1 l_{g_2}\dot{\theta}_1\dot{\emptyset}\cos(\emptyset - \theta_1) \tag{9}$$

$$\sum U = \left[m_1 g l_{g_1} + m_2 g l_1\right]\sin\theta_1 + m_2 g l_{g_2}\sin\emptyset \tag{10}$$

Based on the L-E equation, the equations of motion of the system are derived. The matrix form of these equations of motion is given in equation (11).

$$\begin{bmatrix}\tau_1\\ \tau_2\end{bmatrix} = \begin{bmatrix} m_1 {l_{g_1}}^2 + I_1 + m_2 {l_1}^2 & m_2 l_1 l_{g_2}\cos(\emptyset - \theta_1) \\ m_2 l_1 l_{g_2}\cos(\emptyset - \theta_1) & m_2 {l_{g_2}}^2 + I_2 \end{bmatrix}\begin{bmatrix}\ddot{\theta}_1\\ \ddot{\emptyset}\end{bmatrix} - \begin{bmatrix} 0 & -(m_2 l_1 l_{g_2})\dot{\emptyset}\sin(\emptyset - \theta_1) \\ m_2 l_1 l_{g_2}\dot{\theta}_1\sin(\emptyset - \theta_1) & 0 \end{bmatrix}\begin{bmatrix}\dot{\theta}_1\\ \dot{\emptyset}\end{bmatrix} + \begin{bmatrix} m_1 g l_{g_1} + m_2 g l_1\cos\theta_1 \\ m_2 g l_{g_2}\cos\emptyset \end{bmatrix} \tag{11}$$

Finally, the matrix-vector representation of the equation of motion is given in equation (12).

$$\tau = M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) \tag{12}$$

3.3. MATLAB/Simulink Modelling of the System

After the design of the robot arm system was completed in Solidworks program, the system was transferred to MATLAB Simulink environment (MathWorks, 2003-2020). The appearance of the design of the robot arm system created in Solidworks environment is given in Figure 2.

Figure 2: 3D view of the robot arm.

By transferring the designed robot arm to the Simulink environment, the basic block diagram schematic of the robot arm system was created by MATLAB Simulink. The basic block diagram of the robot arm provided by Simulink is given in Figure 3.

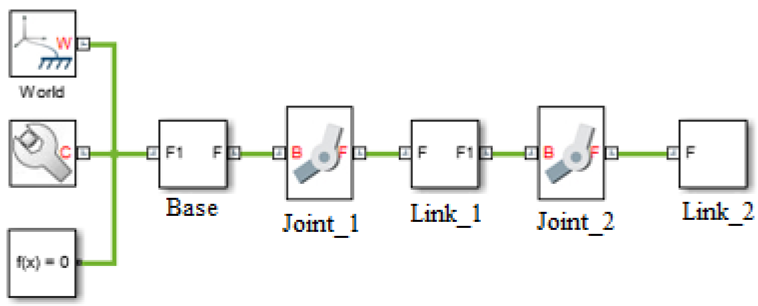

Figure 3: Basic block diagram of the robot arm.

In the basic block diagram of the robot arm given in Figure 3, the blocks representing the parts of the robot arm model can be expressed as follows; "Base" represents the base part of the robot arm system, "Joint_1" represents the joint where the angle θ_1 is applied, "Link_1" represents the first link of the robot arm system, "Joint_2" represents the joint where the angle θ_2 is applied and "Link_2" represents the second link of the robot arm system. The angles θ_1 and θ_2 are controlled by sensors. Counterclockwise is considered positive.

The forward kinematic physical model schematic of the robot arm with two degrees of freedom is given in Figure 4.

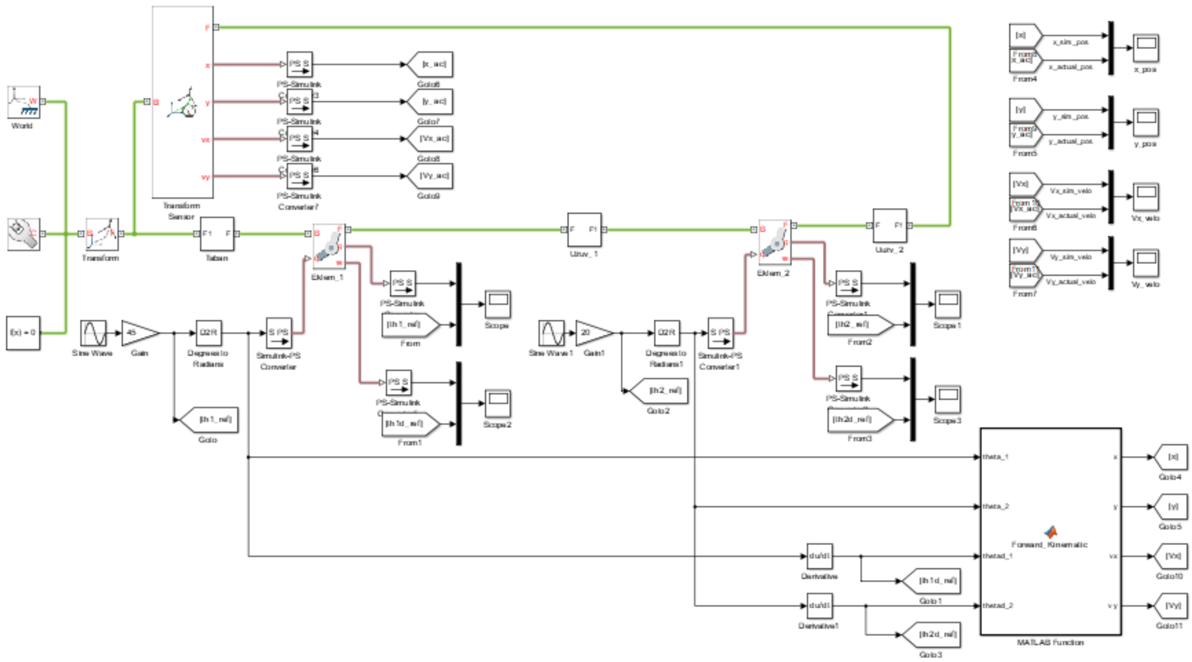

Figure 4: Two degrees of freedom robot arm forward kinematic modelling.

The forward kinematic mathematical modeling of the system was created by coding the forward kinematic equations of the robot arm system into the "Matlab Function" block in Figure 4. Physical model outputs (x_ac, y-ac, vx_ac, vy_ac, vy_ac) and mathematical model outputs (x, y, vx, vy) were compared with the help of scope blocks.

The inverse kinematic modeling of the robot arm system is given in Figure 5. For inverse kinematic modelling, two different trajectories are designed in which the end point of the robot arm is desired to follow. One of these trajectories allows the end point of the robot arm to move in a straight line, while the other allows it to move in a circle. In this way, the operational limits of the robot arm are determined. Then, the inverse kinematic mathematical modelling of the system was created with the inverse kinematic equations encoded in the function block. The "Robot Arm" block in Figure 5 represents the physical modeling of the robot arm system.

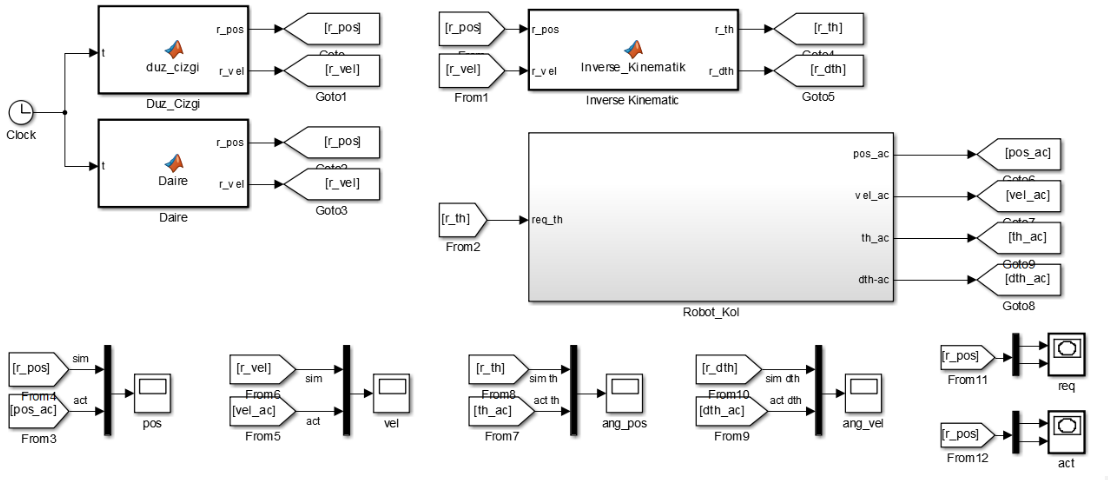

Figure 5: Two degrees of freedom robot arm inverse kinematic modelling.

The dynamic modeling of the robot arm system is given in Figure 6. During the dynamic modeling of the robot arm system, the mathematical modeling of the dynamic model was created by coding the equations of motion obtained with the L-E equation described in section 3.2 into the function block. The "Robot arm" block in Figure 6 represents the physical modeling of the robot arm system.

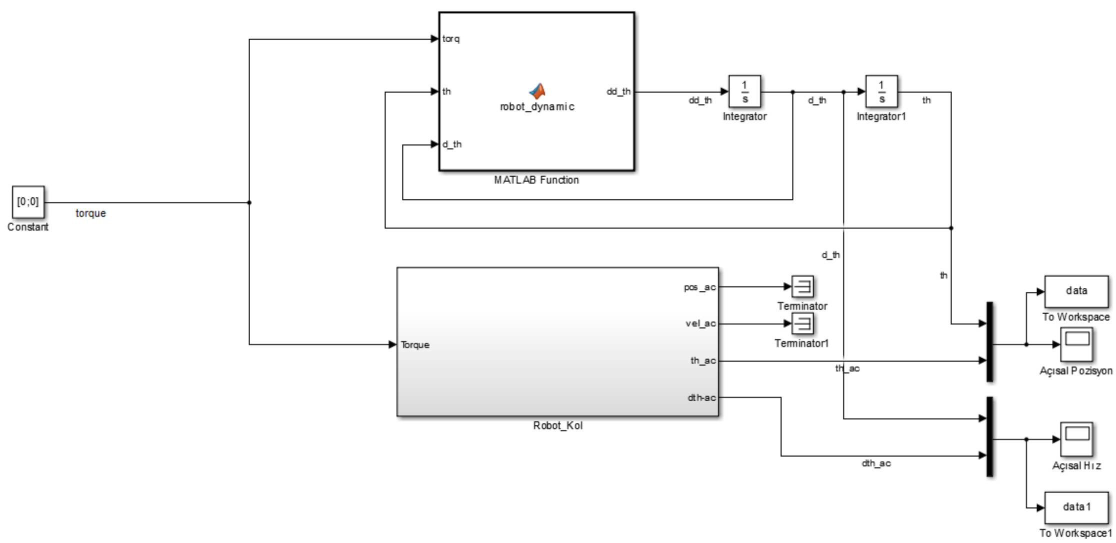

Figure 6: Two degrees of freedom robot arm dynamic modelling.

4. RESULTS AND DISCUSSION

The mathematical model outputs and physical model outputs of both the kinematic model and the dynamic model of the robot arm system with two degrees of freedom are compared. The position outputs of the mathematical models and physical models of the forward kinematic, inverse kinematic and dynamic systems were compared and checked for matching. The same process was repeated for velocity outputs.

The graph of the linear position outputs giving the position of the forward kinematic robot endpoint is given in Figure 7. The angle θ_1 applied to the mathematical model is the same as the angle θ_1 applied to the physical model. For the first joint of the robot arm to move 30 degrees, a $30 * \sin(5 * t)$ radian sine wave was applied to the first joint. Likewise, the value of the angle θ_2 applied to the mathematical model is the same as the value of the angle θ_2 applied to the physical model. For the second joint of the robot arm to move 20 degrees, a $20 * \sin(5 * t)$ radian sine wave was applied to the second joint.

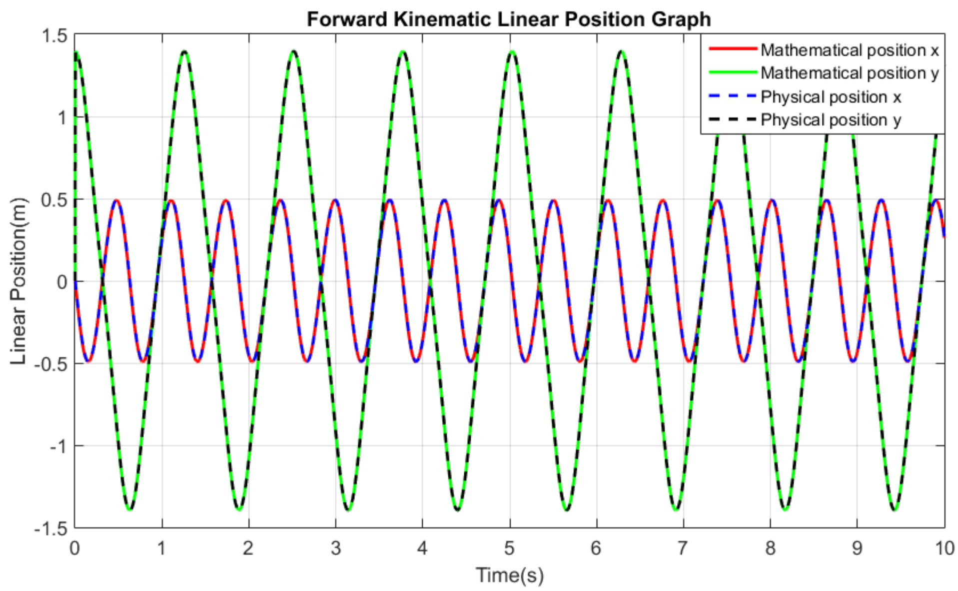

Figure 7: Forward kinematic linear position graph.

The graph of forward kinematic velocity outputs is given in Figure 8. The value of $\dot{\theta}_1$ applied to the mathematical model is the same as the value of $\dot{\theta}_1$ applied to the physical model. A $\frac{5*\pi}{6} * \cos(t)$ radian sine wave was applied to the first joint of the robot arm. Similarly, $\dot{\theta}_2$ applied to the mathematical model is the same as $\dot{\theta}_2$ applied to the physical model. A $\frac{5*\pi}{9} * \cos(t)$ radian sine wave was applied to the second joint of the robot arm. These applied velocities $\dot{\theta}_1$ and $\dot{\theta}_2$ are also the derivatives of the angles $\theta_1$and θ_2.

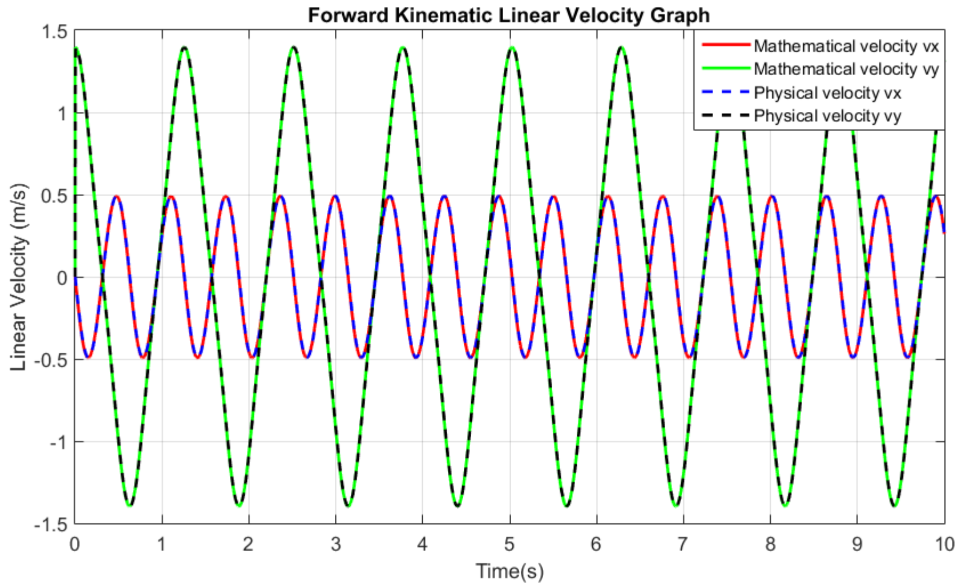

Figure 8: Forward kinematic linear velocity graph.

Inverse kinematics is a method to find the angle required to be in the desired position. In the two degrees of freedom robot arm system, x and y positions and linear velocities forming straight line and x and y positions and linear velocities forming circle were sent as input data to the mathematical and physical models. Thus, the angular positions and angular velocities required to obtain straight line and circle trajectories were achieved.

In the inverse kinematic model, the graph of the angular position outputs obtained by sending the input values required for the end point of the robot arm to follow the desired straight line trajectory to the mathematical and physical models is given in Figure 9.

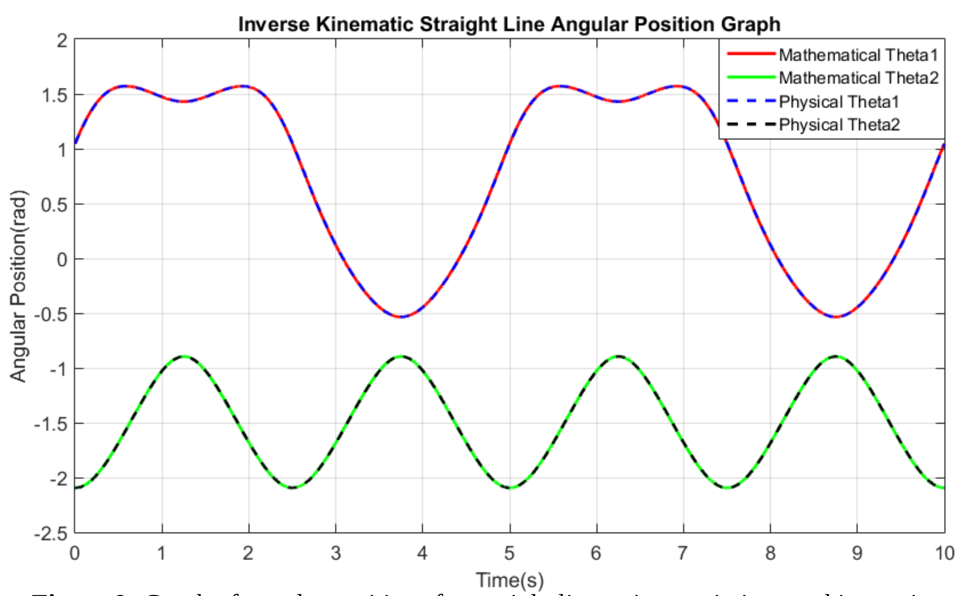

Figure 9: Graph of angular positions for straight line trajectory in inverse kinematics.

The graph of the angular velocity outputs obtained by sending the input values required for the robot arm to follow the straight line trajectory in the inverse kinematic model to the mathematical model and physical models is given in Figure 10.

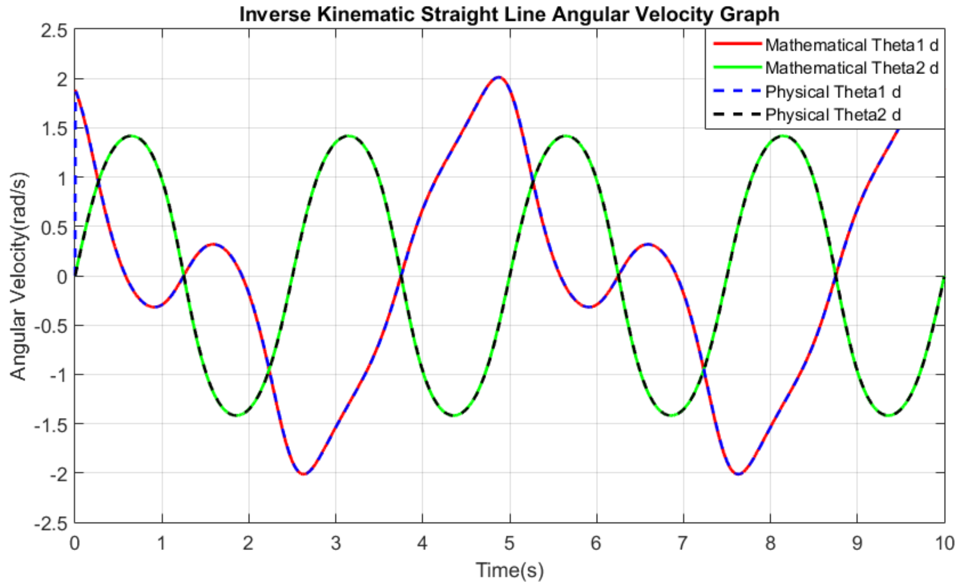

Figure 10: Graph of angular velocities for straight line trajectory in inverse kinematics.

In the inverse kinematic model, the graph of the angular position outputs obtained by sending the input values required for the end point of the robot arm to pursue the desired circle trajectory to the mathematical and physical models is given in Figure 11.

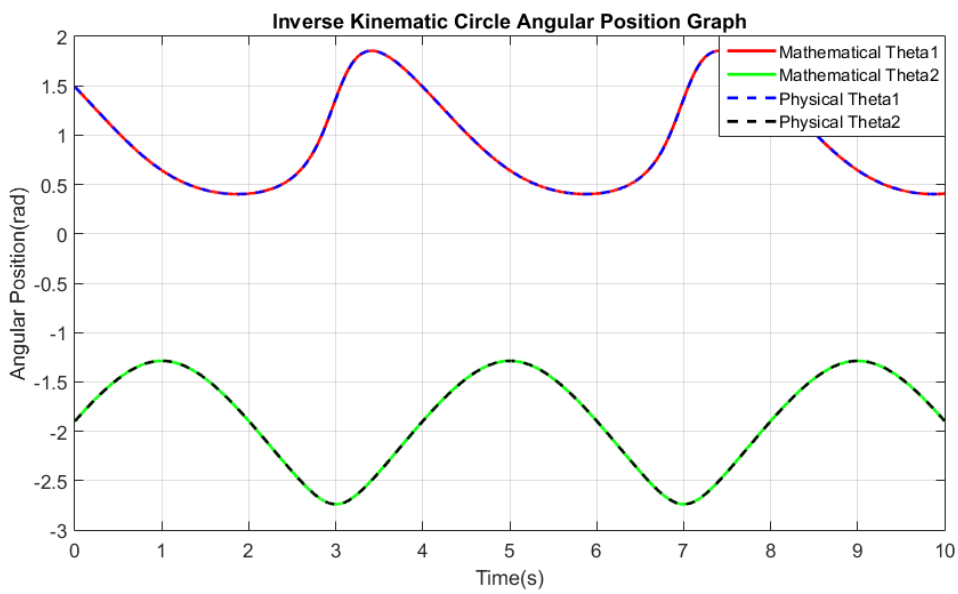

Figure 11: Graph of angular positions for circle trajectory in inverse kinematics.

The graph of the angular velocity outputs obtained by sending the input values required for the robot arm to follow the circle trajectory in the inverse kinematic model to the mathematical model and physical models is given in Figure 12.

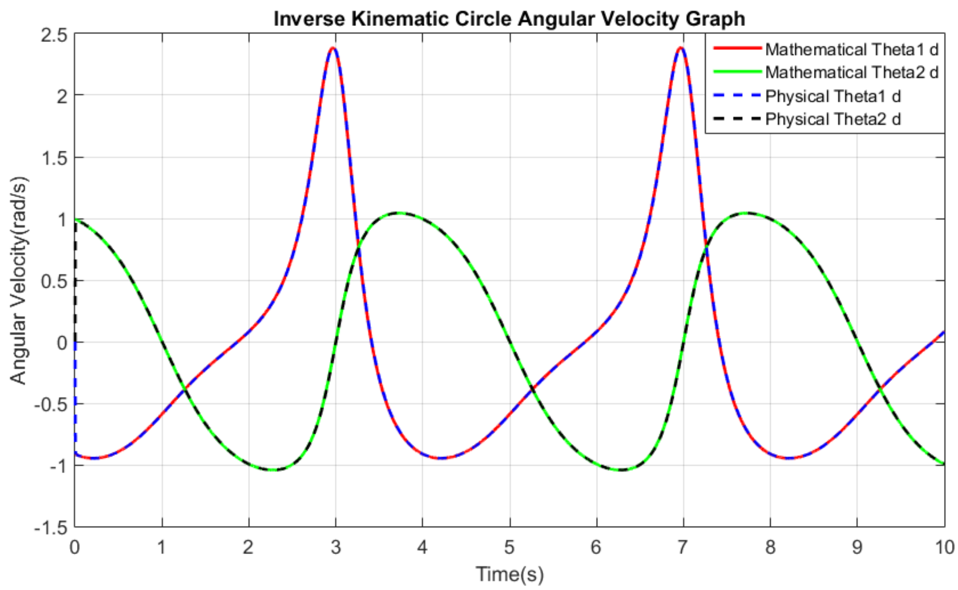

Figure 12: Graph of angular velocities for circle trajectory in inverse kinematics.

The input torque value applied to the mathematical model of the dynamic system is the same as the input torque value applied to the physical model. The angular position outputs of the dynamic system model of the robot arm are given in Figure 13.

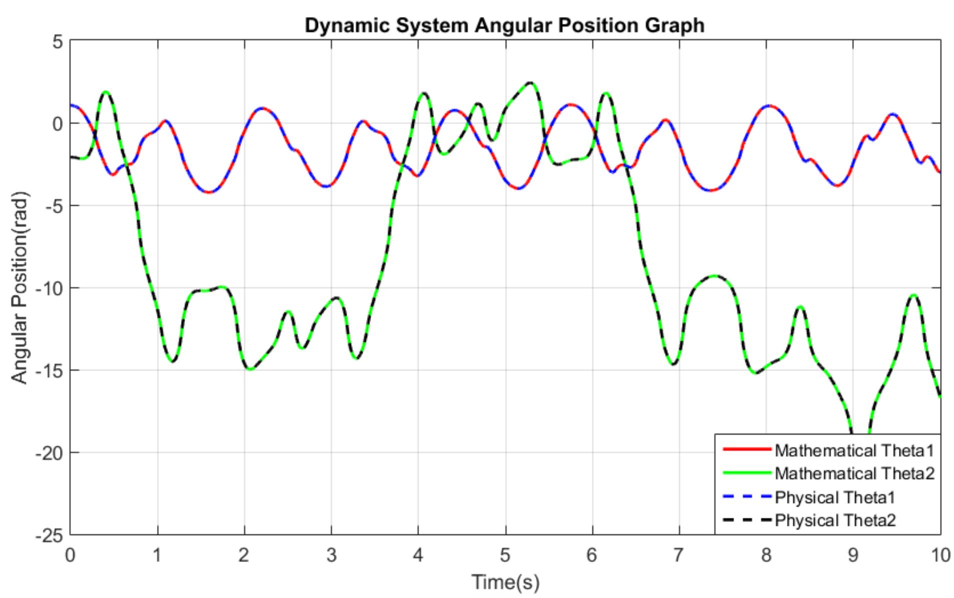

Figure 13: Dynamic system angular positions graph.

The velocity outputs of the dynamic system model of the robot arm system with two degrees of freedom are given in Figure 14.

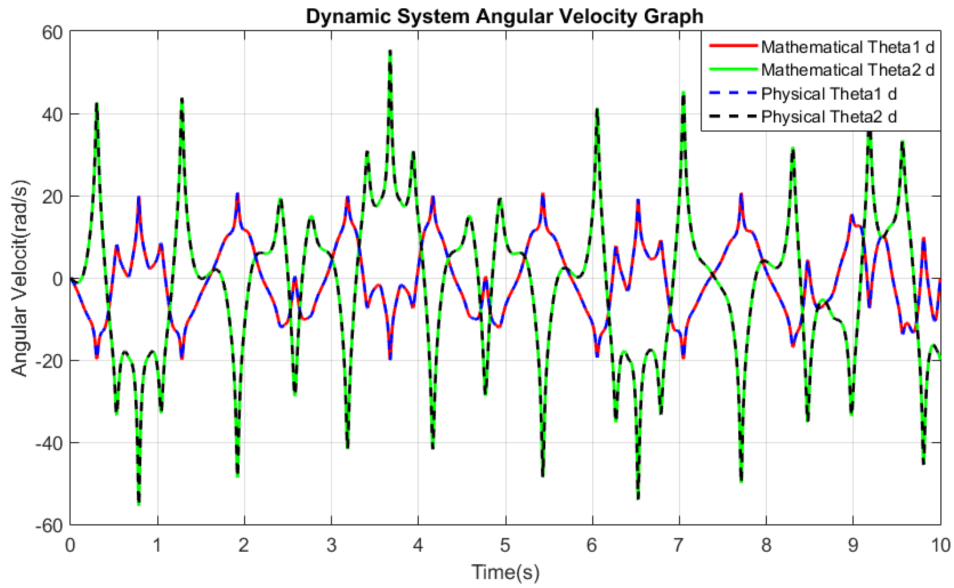

Figure 14: Dynamic system angular velocities graph.

5. CONCLUSION

In this study, firstly, a robot arm system with two degrees of freedom was designed in Solidworks and then this system was transferred to MATLAB Simulink and it was investigated whether the mathematical modeling and physical modeling matched. Different from other studies, dynamic modelling of the system was performed as well as kinematic modeling. The basic block diagram schematic required for the physical modeling of the robot arm system was obtained by transferring the robot arm system to MATLAB Simulink. Then the kinematic model and dynamic model systems were built with Simulink block diagrams on top of the basic block diagram scheme. The mathematical model of the robot arm system was created by writing the kinematic equations and dynamic equations of the two degrees of freedom system as code in function blocks in Simulink. Mathematical and physical models of the robot arm system were compared for forward kinematic, inverse kinematic and dynamic systems. While comparing the physical and mathematical models for the forward kinematic system, the same angle and velocity values were sent to the joints of the robot arm system and the trajectory followed by the endpoint of the robot arm was compared. When comparing the mathematical and physical model of the robot arm system in terms of the inverse kinematic system, two different trajectories that the robot arm endpoint expected to follow were created. When these trajectories were used as input values, it was checked whether the mathematical and physical models gave the same angular position and angular velocity values as output. For the dynamic system tests, the same torque value was sent as input to both the mathematical model and the physical model and the angular position and angular velocity outputs were compared. As a result of the comparisons, it was seen that the linear position, linear velocity, angular position and angular velocity outputs of the mathematical and physical models of the two degrees of freedom robot arm system matched. It is observed that the results obtained from the mathematical equations and the results obtained from the sensors of the physical model created in the simulation environment coincide. In other words, the mathematical models of the system and the physical models produced the same outputs. This proves the accuracy of the physical model of the robot arm system.

CONFLICT OF INTEREST STATEMENT

There is no conflict of interest among the authors.

CONTRIBUTIONS OF AUTHORS

A.B.: Conceptualization, methodology, investigation, simulation, writing-original draft preparation.

G.T.: Methodology, supervision, validation, review and editing.

REFERENCES

Aydın, M. (2022). Position Control of Slider Crank Mechanism. *Avrupa Bilim ve Teknoloji Dergisi*(42), 158-162.

Aydın, M. (2022). Real-Time PID Control of Three Degrees of Freedom RRR Type Robot Arm. In S.E. Ruşen & A. Ruşen (Eds.), *INSAC World Natural and Engineering Sciences* (pp. 243-256).

Aydın, M. (2022). *Speed Control of Slider-Crank Mechanism*. Paper presented at Anadolu 10th International Conference on Applied Sciences, Diyarbakır, Turkiye .

Barakat, A. N., Gouda, K. A., & Bozed, K. A. (2016). *Kinematics Analysis and Simulation of a Robotic Arm using MATLAB.* Paper presented at 4th International Connce on Control Engineering & Information Technology (CElT - 2016). Hammamet, Tunisia.

Craig, J. J. (2014). *Introduction to Robotics: Mechanics and Control.* Harlow: Pearson.

Ghaleb, N. M., & Aly, A. A. (2018). Modeling and Control of 2-DOF Robot Arm. *International Journal of Emerging Engineering Research and Technology, 6*(11), 24-31.

Gürgüze, G., & Türkoğlu, İ. (2019). Kullanım Alanlarına Göre Robot Sistemlerinin Sınıflandırılması. *Fırat Üniversitesi Mühendislik Bilimleri Dergisi, 1*(31), 53-66.

Hüseyinoğlu, M., & Abut, T. (2018). Dynamic Model and Control of 2 Dof Robotic Arm. *European Journal of Technic (EJT), 8*(2), 141-150.

Kebria, P. M., Al-wais, S., Abdi, H., & Nahavandi, S. (2016). *Kinematic and Dynamic Modelling of UR5 Manipulator.* Paper presented at 2016 IEEE International Conference on Systems, Man, and Cybernetics, Budapest, Hungary.

Küçük, S., & Bingül, Z. (2004). Robot Sistemlerinde Kinematik Yöntemlerin Karşılaştırılması. *Politeknik Dergisi, 7*(2), 107-117.

Küçük, S., & Birgül, Z. (2006). Robot Kinematics: Forward and Inverse Kinematics. In S. Cubero (Eds.), *Industrial Robotics Theory Modelling Control*. Pro Literatur Verlag, Germany / ARS, Austria.

Mahto, R. K., Kaur, J., & Jain, P. (2022). *Performance Analysis of Robotic Arm using Simulink.* Paper presented at 2022 IEEE World Conference on Applied Intelligence and Computing (pp. 508-512).

MathWorks. (2003-2020). *Simscape Multibody Link Reference.* The MathWorks Inc.

Okubanjo, A. A., Oyetola, O. K., Osifeko, M. O., Olaluwoye, O. O., & Alao, P. O. (2017). Modeling of 2 DOF Robot Arm and Control. *Futo Journal Series (FUTOJNLS), 3*(2), 80-92.

Sahay, N., Chattopadhyay, S., & Chowdhury, T. (2020). *Simulation of Robot Arm Dynamics Using N-E Method of 2-Link Manipulator.* Paper presented at Fifth International Conference on Inventive Computation Technologies, Coimbatore, India.

Vivekanandan, P., Vishnu, A., Narayanasamy, S. K., & Yasodharan, R. (2019). Modelling, Simulation and Control of 5 Axis Industrial Robot using MATLAB. *Indian Journal of Science and Technology, 12*(23), 1-7.

Yıldırım, G. (2021). *3 Eksenli Bir Robot Kolun Kontrol Metotlarının İncelenmesi Ve Verimlilik Analizi.* (Yayımlanmamış yüksek lisans tezi). Isparta Uygulamalı Bilimler Üniversitesi, Isparta.

Zhang, D., Ma, W., & Zhang, H. (2022). *Analysis and Simulation of Mechanical Arm Dynamics Model Based on Simulink.* Paper presented at 2nd International Conference on Algorithms, High Performance Computing and Artificial Intelligence (AHPCAI), Guangzhou, China.