

5-DOF serial robot manipulator design, application and inverse kinematic solution through analytical method and simple search technique

5 serbestlik dereceli (SD) seri robot manipulatörü tasarımı, uygulaması ve analitik yöntem ve basit arama tekniği yoluyla ters kinematik çözümü

Sabri UZUNER^{1*}, Nihat AKKUS², Metin TOZ³

¹Department of Electronics and Automation, Cumayeri Vocational School, Duzce University, Duzce, Turkey.

sabriuzuner@duzce.edu.tr

²Department of Mechatronics, Faculty of Technology, University of Marmara, Istanbul, Turkey.

nihat.akkus@marmara.edu.tr

³Department of Computer, Faculty of Technology, Duzce University, Duzce, Turkey.

metintoz@duzce.edu.tr

Received/Geliş Tarihi: 01.03.2019
Accepted/Kabul Tarihi: 22.07.2019

Revision/Düzeltilme Tarihi: 06.07.2019

doi: 10.5505/pajes.2019.95881
Research Article/Araştırma Makalesi

Abstract

In this study a five Degrees of Freedom (DOF) serial robot manipulator was designed and implemented. The inverse kinematics problem, which has not exact analytical solution (only one inverse kinematic solution for a predefined end effector position in three dimensional space), of the robot mechanism was solved by using the combination of the analytical method and a simple search method. In order to use the proposed method in real-time applications, the method is designed so that it can be used to solve the inverse kinematics problem for the next point of the end-effector while the mechanism is working. Moreover, so that to control the implemented mechanism, a user interface program was written by using Visual Basic programming language. Finally, the proposed inverse kinematic solution method was tested on two different trajectories, an arc shaped trajectory that composed of 58 points and a linear trajectory divided into 29 points. The obtained results revealed that the proposed method can be used successfully in solving the inverse kinematic problem of the designed mechanism.

Keywords: 5 DOF Serial robot, Inverse kinematic, Forward kinematic, Simple search algorithm, Robotic arm.

Öz

Bu çalışmada, beş serbestlik derecesine (SD) sahip bir seri robot manipulatörü tasarlanmış ve test edilmiştir. Robot mekanizmasının kesin analitik çözümü (üç boyutlu uzayda önceden tanımlanmış bir uç işlevcisi konumu için yalnızca bir ters kinematik çözümü) olmayan ters kinematik problemi, analitik yöntem ve bir basit arama metodu kombinasyonu kullanılarak çözülmüştür. Önerilen yöntem, gerçek zamanlı kullanılabilmesi için mekanizma çalışırken uç işlevcinin gideceği bir sonraki nokta için çözüm yapılabilecek şekilde tasarlanmıştır. Gerçekleştirilen mekanizmayı kontrol etmek için Visual Basic programlama dili kullanılarak bir kullanıcı arayüzü programı yazılmıştır. Son olarak, önerilen ters kinematik çözüm yöntemi 58 noktadan oluşan yay şeklinde bir yörünge ve 29 noktaya bölünmüş doğrusal bir yörünge olmak üzere iki farklı yörüngede test edilmiştir. Elde edilen sonuçlar, önerilen yöntemin tasarlanan mekanizmanın ters kinematik problemini çözmeye başarıyla kullanılabileceğini ortaya koymuştur.

Anahtar kelimeler: 5 SD seri robot, Ters kinematik, İleri kinematik, Basit arama algoritması, Robotik kol.

1 Introduction

In recent years, robot mechanisms have been used in almost every field in the industry, such as automotive technology, space exploration, rescue operations etc. Besides the industry, in the literature, it is seen that robotic mechanisms are also designed for using in special applications such as medical [1]-[2], agricultural [3] or mine rescue [4] applications. Serial robotic mechanisms consisting of rigid bodies that connected by revolute or prismatic joints are the most well-known robot mechanisms. The solution of the inverse kinematic problems of these mechanisms is vital in determining the appropriate joint parameters in real-time control applications. The robot controller employs these equations to calculate joint values by using the position of the end effector of the mechanism. Inverse kinematic problems are complex and highly nonlinear problems for analytical solutions and they closely related to the design of the relevant mechanism [5]. Regarding the analytical solution of these problems, more than one solution can be applicable for single position information in three dimensional

space depending on the structural parameters of the robot mechanism. Therefore, determination of which one of these solutions is the real solution is particularly problematic with respect to both the processing load and the duration in real-time control applications. For this reason, except for analytic solution, a number of different methods have been proposed for solving these problems in the literature. Some of these methods can be given as follows; Koker [6] employed an artificial neural network and genetic algorithm-based method in order to solve the inverse kinematic problem of a six-joint Stanford robot manipulator. Fu et al. [7] proposed a geometric algebraic method to solve the inverse kinematic problem of a 6R (R: Rotary Joint) robot mechanism with an offset wrist. Kucuk and Bingül [8] handled the inverse kinematic problem of sixteen basic six DOF serial robot manipulators with offset wrists and presented a numerical solution technique for analytically unresolved problems. In this technique, the authors randomly selected the first joint variable of the sample mechanism and then calculated the other joint variables iteratively, depending on this value. Liu et al. [9] solved the

*Corresponding author/Yazışılan Yazar

inverse kinematic problem of a five DOF serial robot manipulator through a method called virtual joint approach. Sarıyıldız and his colleagues [10] compared three different screw theory-based methods in order to solve the inverse kinematic problem of serial industrial robot manipulators. Park et al. [11] also proposed an iterative algorithm for the inverse kinematic solution of a five DOF serial robotic mechanism with offset wrist. Like the study of Kucuk and Bingül [8], this algorithm also starts with an estimated orientation value and iteratively reaches the solution. Sheng et al. [12] proposed a geometric inverse kinematic solution method for planar serial robot mechanisms that have redundant DOFs and tested their method on a sample of five DOF mechanism. El-Sherbiny et al. [13] tested the performance of artificial neural networks, fuzzy logic, and Genetic Algorithm-based methods which are used for inverse kinematic solution of serial robot mechanisms tested on a five DOF mechanism and then compared the results. Chaichawanit and Saiyod [14] proposed the A* algorithm for solving the inverse kinematic problem of serial robots. Feng et al. [15], used feed forward neural networks with a novel learning algorithm called extreme learning machine (ELM) for solving the inverse kinematics problem of serial robot manipulators. In another study, Kucuk and Bingül [16] proposed a new numerical algorithm, called NIKA for solving inverse kinematic problem of the serial robot manipulators with offset wrist. Aristidou and Lasenby [17] proposed a new heuristic iterative method called Forward and Backward Reaching Inverse Kinematics (FABRIK), for solving the inverse kinematic problem of the serial robot mechanisms. This approach uses a forward and backward iterative approach to find the solutions. Toz [18], proposed to use Ant Colony Optimization Algorithm for solving the inverse kinematic problem of a six DOF serial robot manipulator with offset wrist. Dereli and Köker [19] solved the inverse kinematics problem of a seven DOF redundant serial robot manipulator by using different forms of the PSO algorithm and concluded that PSO forms that use inertia weight parameters are more powerful than the standard one. In another study Dereli and Köker [20], solved the inverse kinematics problem of the same mechanism by a quantum behaved PSO algorithm. In [21], Li et al. designed a bio-inspired humanoid torso for the humanoid robots and solved the inverse kinematics problem of the mechanism by an algorithm that obtained as combining the chaos optimization and the quasi-Newton method. The authors found that the proposed hybrid method is more effective than the Newton-type methods in solving the inverse kinematics problem. Hrdina et al. [22] used geometric algebra for solving the inverse kinematics problem of the planar serial robot manipulators and also proposed different approaches for segmentation of the actuating rotations.

For the mechanisms of which inverse kinematic problem cannot be solved analytically, there also exist a number of different methods in the literature besides aforementioned methods. However, because of the different design strategies of the serial robot mechanisms and the different types of the joints and links used for these robots, there is no specific method that can be generalized to solve the inverse kinematic problem of all the serial robot mechanisms. As an example, the inverse kinematics solution of a six DOF serial robot mechanism that have Euler wrist is easier than the inverse kinematics solution of a robotic arm that uses an offset wrist instead of the Euler wrist. The reason behind this distinction is the differences between the structures of the Euler and the offset wrists. An Euler wrist is composed of three joints that have their axes are

intersecting on the same point while the axes of an offset wrist are not intersecting on the same point. Therefore, the proposed methods generally apply to a single mechanism or to a group of mechanisms. In the present study, a five DOF serial robot manipulator was designed and implemented using five revolute joints. The analytical method was not preferred because the solution of the inverse kinematic problem of this mechanism is not analytically applicable. In addition, optimization or artificial neural network-based methods were not adopted because of their long processing time and random values used in the initial stage. Instead, a method similar to the one proposed in [8] and [11] was employed in order to solve the inverse kinematic problem of the mechanism. In this proposed method, first, the angle of first joint is obtained analytically, then the angle of second joint is selected by a simple search technique and then other joint variables are obtained analytically. It should be noted that the proposed mechanism and the method were given in the MSc Thesis by Uzuner[27]. The difference between this method and other methods in the literature is that no estimated value is used at the beginning stage and only one angle value is searched in the algorithm and the others are analytically calculated. In this way, process time is reduced. The proposed mechanism was realized using RC servo motors to perform rotational movements. The proposed mechanism is designed to be controlled by an open loop control system. Thus, a POLOLU Micro Maestro Servo controller was used to send the control signals from the user interface program to the motors. Apart from these, other equipment (power supply, etc.) are chosen in relation to the needs of the mechanism. In addition, a graphical user interface has been designed through the Visual Basic programming language to help the user control the robot mechanism. In order to test the effectiveness of the proposed method, two different trajectories, linear and arc-shaped trajectories, were used and both the theoretical results and the experimental results were obtained by the experiments on the developed mechanism. The contributions of this work are as following:

1. A five (DOF) serial robot manipulator was designed and implemented,
2. The inverse kinematics problem of the designed mechanism was solved by using a novel method that consist of an analytical method and a simple search method,
3. The advantages of the proposed method are, there is no estimate value of any joints is required before inverse kinematics solution and only one angle value is searched in the algorithm. Thus, the solutions can be obtained in 18.6 ms average time that is very important in real-time control applications. And the proposed method with the help of the developed user interface program is successfully used to control the five DOF robotic manipulator in real time. As a final point, the proposed method solves the inverse kinematics problem while the mechanism is working. Thus, a nearly continuous moving is provided,
4. A user interface program was written for controlling the designed mechanism,
5. The proposed inverse kinematic solution method was tested on two different trajectories.

The organization of the other sections of this paper is as follows: In the second section, the design steps of the serial robot mechanism were briefly described. In the third section,

the forward and inverse kinematic equations of the designed mechanism were obtained using the DH method and the inverse kinematic problem of the proposed mechanism was solved. In the fourth section, experimental studies and their results were presented, and in the last section, the findings were summarized.

2 5 DOF serial robot mechanism and the design steps

The proposed serial robot mechanism consists of five revolute joints and a gripper as end-effector (Figure 1). The design steps of the mechanism are as follows;

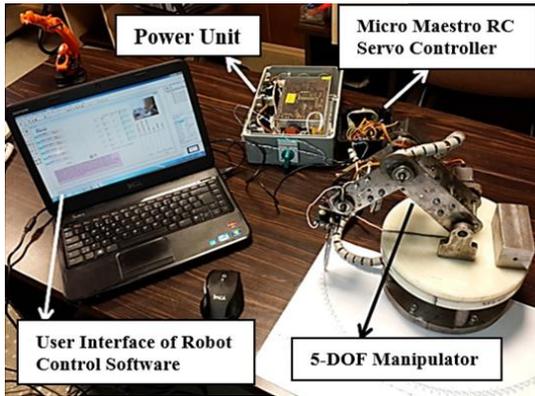


Figure 1. The designed robotic system [27].

2.1 Mechanical design

Since the mechanism was designed for testing purposes, the dimensions of the mechanism were determined accordingly within the limits that the mechanism can be used on an experiment table. It was also determined that the end effector has a capacity to carry up to 150 g. The CAD drawing of the developed mechanism was shown in Figure 2. Regarding the mechanical structure, there are some mechanical limitations in terms of its joints. These limitations are presented in Table 1.

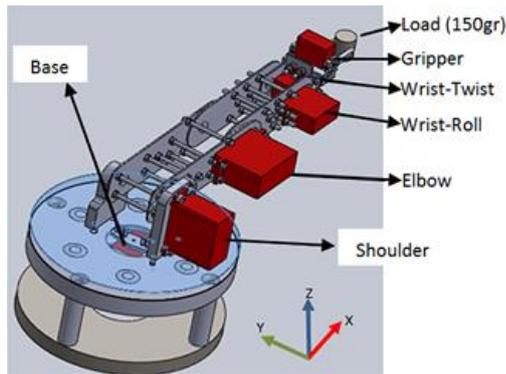


Figure 2. CAD drawing of 5-DOF manipulator [27].

Table 1. Mechanical limitations of the mechanism in terms of its joints.

Axis	Rotation angles
Gripper	0~330mm
Wrist-Roll	47.32° ~ (-47.64°)
Wrist-Twist	86.75° ~ (-83.34°)
Base	107.26° ~ (-102.8°)
Elbow	95.07° ~ (-84.36°)
Shoulder	146.831° ~ (-3.19°)

The most important step with respect to the mechanical design of serial robot mechanisms is the choice of motor for each joint [23]. In the present study, RC servo motors were used for the joints of the mechanism. The properties of the motors were given in Table 2.

Table 2. Parameters of the selected RC servomotors.

Axis	Torque (Nm)
Motor of Gripper	0.42168 Nm
Motor of Wrist-Roll	0.4903325 Nm
Motor of Wrist-Twist	1.2552512 Nm
Motor of Base	2.941995 Nm
Motor of Elbow	3.9 Nm
Motor of Shoulder	2.941995 Nm

2.2 Design of electronics

The electronic part of the designed serial robot mechanism involves the power supply and the control unit. There are six servo motors in the system. Three of them require 7.2 V input voltage and the other three require 6 V input voltage. For this reason, two different regulator circuits were designed for the 6 V motor and 7.2 V motor. Micro Maestro 6 Channel USB Servo Controller was used as the control unit of the developed system. This unit allows controlling the speed and acceleration of each motor separately. Figure 3 displays the power supply and control unit.

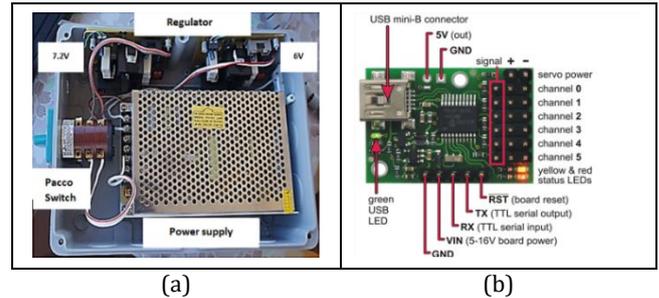


Figure 3. Electronic parts of the robotic arm. (a): Power supply and (b): Micro maestro servo controller.

2.3 User interface program

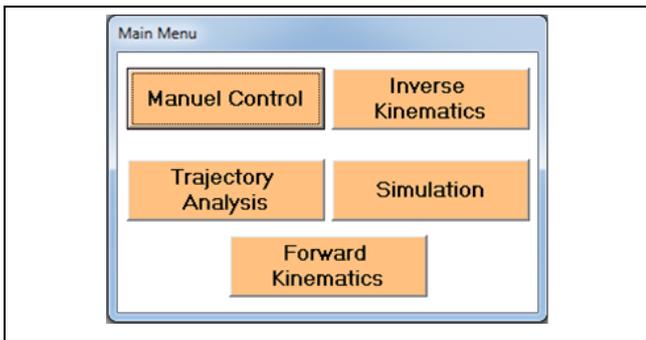
The required user interface program to control the developed serial robot mechanism was written through the Visual Basic programming language. This program is composed of five main parts: manual control, trajectory analysis, forward kinematics, inverse kinematics, and simulation. Figure 4a) displays the main screen of the program. This screen can be used for switching other screens according to the user's needs. In the manual control section (Figure 4b), each joint of the robot arm can be controlled independently of each other. In this section, the user was provided with the information of position, velocity, and acceleration for end effector and the opportunity to control the robot on a trajectory determined in advance and composed of several trajectory points. This part was named "serial motion" in the interface program. A series of movements can be repeated once or they can be performed continuously within a loop as occurring in pick-and-place motions. In the trajectory analysis section (Figure 4c), the third order polynomials were used to provide sensitive and vibration-free motion of the robot. This section provides position, velocity an acceleration graphics of the end-effector within the desired time. This program module has also been experimentally tested in a study by Uzuner et al. [24]. In the forward kinematics section (Figure 4d), the necessary controls for the forward kinematics analysis of the mechanism are involved. There are

two kinds of movements in the forward kinematic analysis section. One of them is the movement according to the reference (Go with respect to reference) and the other is according to the last position of the end effector (Go with respect to last position). The motion by the reference makes it possible to calculate the position of the end effector by the angle information inputted to the angle sections at every turn according to the reference coordinate system. On the other hand, in the movement according to the last position, the angular values of the last position of the end-effector are accepted as the reference and the movement is performed according to this last position. In addition, "Normal movement" and "Planned movement" options are available. Normal movement performs robot motion without trajectory planning between two points while planned movement accomplishes robot motion by trajectory planning between any two points. In the inverse kinematics section (Figure 4e), the inverse kinematics of the mechanism is solved through both analytical and the simple search techniques. In this section, the desired coordinates of the end effector are entered into the X-Y-Z sections and the inverse kinematic problem is solved by the proposed method with the help of the solver button. The results obtained are presented in the "result section". All of the

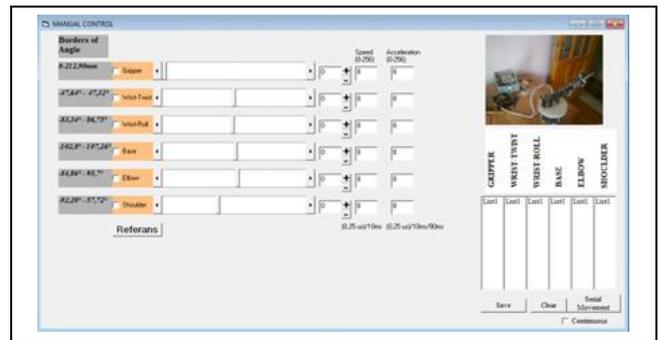
obtained solutions are applicable solutions for the robot; so, there are no solutions within these results which are not applicable because of the mechanical limitations of the robot. The reason for the large number of results is that all possible solutions in the loop for the second joint variant are obtained with a sensitivity of 0.01 degrees. By choosing any solution in these results (go by the selected one), the motion of the robot can be performed. By selecting the "Go by the least energy" option, the solution with the least sum of the angles among all the solutions will be selected as the solution and the robot arm will go to the specified position by using this solution, this option provides reaching the desired position with minimum energy. Finally, the Simulation section (Figure 4f) is the part where the movement of the robot arm is simulated in real time. In this section, when the robot arm is performing the desired motion, the simulation of the motion can be seen at the same time.

3 Forward and inverse kinematics

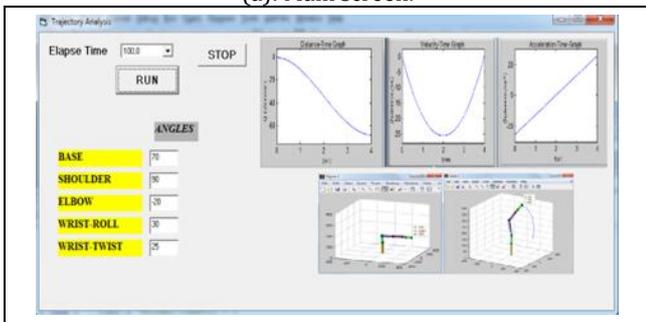
In this section, the forward and inverse kinematic equations of the developed serial robot mechanism are obtained by using the DH method.



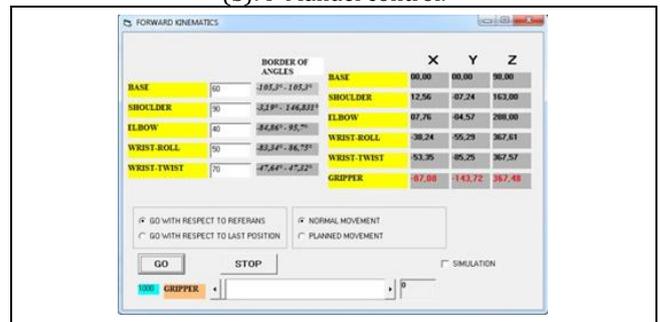
(a): Main screen.



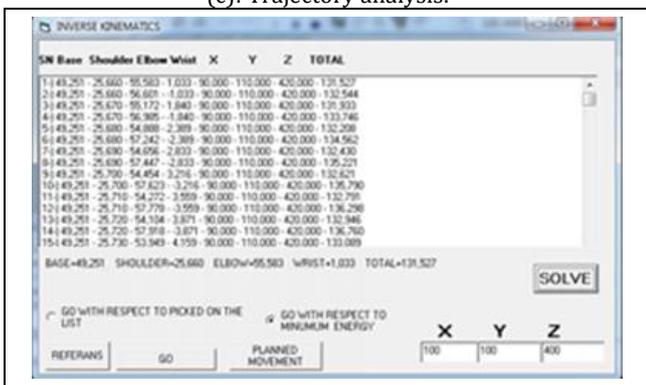
(b): F Manual control.



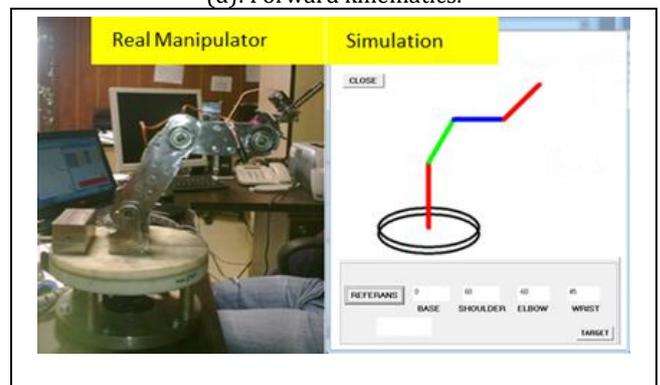
(c): Trajectory analysis.



(d): Forward kinematics.



(e): Inverse kinematics.



(f): Simulation.

Figure 4. Parts of the user interface program.

3.1 Forward kinematics

The forward kinematic equations of a serial robot mechanism describe the position and orientation of the end effector of the mechanism by the joint parameters according to the base coordinate system [8]. In this study, DH method was used to obtain the forward kinematic equations of the implemented mechanism. The placement of the coordinate system on each joint of the mechanism required for this method was shown in Figure 5.

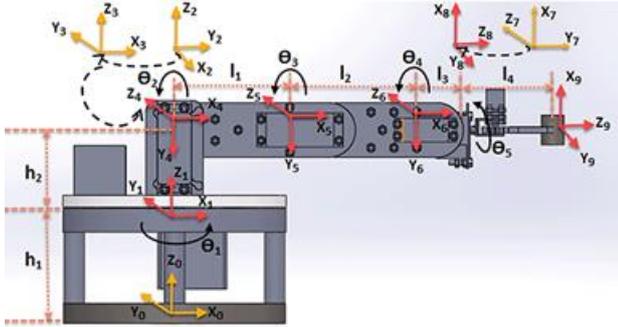


Figure 5. The replacement of coordinate system for DH method.

According to the Figure 5, the DH parameters of the mechanism are obtained as shown in Table 3, [27]. Even though the mechanism has five degrees of freedom, nine coordinate systems were defined by adding "0", "2", "3" and "7" coordinate systems to the joint coordinate systems for accurately determination of the mechanical structure of the mechanism. The main reason behind employing the nine coordinate systems instead of five coordinate systems is the e value given in Table 3. These values are the offsets between joints and are shown in Figure 6. These offsets make the structure of the mechanism more stable [27].

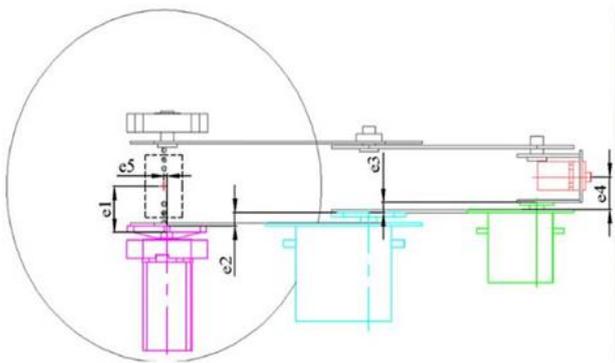


Figure 6. The "e" values of the manipulator.

Table 1. DH parameters [27].

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	$h_1=98$ mm	θ_1
2	0	0	$h_2=65$ mm	-90
3	0	$e_1=22$ mm	0	+90
4	90	0	$e_2=-7.5$ mm	θ_2
5	0	$l_1=125$ mm	$e_3=-5.5$ mm	θ_3
6	0	$l_2=104$ mm	$e_4=-14.5$ mm	θ_4
7	0	$l_3=33.5$ mm	$e_5=1.88$ mm	90
8	90	0	0	θ_5
9	0	0	$l_4=67.5$ mm	0

According to Table 3, transformation matrices for each of the coordinate systems can be written according to Eq. (1) [8].

$${}^{i-1}T_i = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1} d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

Where, ${}^{i-1}T_i$, s and c are transformation matrix between two consecutive coordinate systems, sine and cosine, respectively ($i=1,2,\dots,9$). And, the obtained transformation matrices are as follows;

$${}^0T_1 = \begin{bmatrix} c(\theta_1) & -s(\theta_1) & 0 & 0 \\ s(\theta_1) & c(\theta_1) & 0 & 0 \\ 0 & 0 & 1 & h_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}; {}^1T_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & h_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}; {}^2T_3 = \begin{bmatrix} 0 & -1 & 0 & e_1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^3T_4 = \begin{bmatrix} c(\theta_2) & -s(\theta_2) & 0 & 0 \\ 0 & 0 & -1 & -e_2 \\ s(\theta_2) & c(\theta_2) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; {}^4T_5 = \begin{bmatrix} c(\theta_3) & -s(\theta_3) & 0 & l_1 \\ s(\theta_3) & c(\theta_3) & 0 & 0 \\ 0 & 0 & 1 & e_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}; {}^5T_6 = \begin{bmatrix} c(\theta_4) & -s(\theta_4) & 0 & l_2 \\ s(\theta_4) & c(\theta_4) & 0 & 0 \\ 0 & 0 & 1 & e_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^6T_7 = \begin{bmatrix} 0 & -1 & 0 & l_3 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & e_5 \\ 0 & 0 & 0 & 1 \end{bmatrix}; {}^7T_8 = \begin{bmatrix} c(\theta_5) & -s(\theta_5) & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s(\theta_5) & c(\theta_5) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; {}^8T_9 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & l_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The forward kinematic equations of the mechanism can be obtained by multiplying these transformation matrices and consequently, the transformation matrix which defines the position and orientation of the end effector according to the base coordinate system can be written as in Eq. (3) [4].

$${}^{base}_{end_effector}T = {}^0T_1 {}^1T_2 {}^2T_3 \dots {}^{n-1}T_n \quad (3)$$

Where ${}^{base}_{end_effector}T$ is the transformation matrix for the end-effector. According to the Eq. (3), the position of the end-effector in the Cartesian coordinates was obtained as follows; where Px, Py and Pz are the Cartesian coordinates of the end-effector in x, y and z axes, respectively.

$$\begin{aligned} P_x = & \frac{l_3c(\theta_1 + \theta_2 + \theta_3 + \theta_4)}{2} + \frac{l_4c(\theta_1 + \theta_2 + \theta_3 + \theta_4)}{2} \\ & + \frac{l_1c(\theta_1 + \theta_2) + l_1c(\theta_1 - \theta_2)}{2} + \frac{l_3c(\theta_1 - \theta_2 - \theta_3 - \theta_4)}{2} \\ & + \frac{l_4c(\theta_1 - \theta_2 - \theta_3 - \theta_4)}{2} + \frac{l_2c(\theta_1 - \theta_2 - \theta_3)}{2} \\ & + \frac{l_2c(\theta_1 + \theta_2 + \theta_3)}{2} \\ & + s(\theta_1)(e_1 + e_2 + e_3 + e_4 + e_5) \end{aligned} \quad (4)$$

$$\begin{aligned} P_y = & \frac{l_3s(\theta_1 + \theta_2 + \theta_3 + \theta_4)}{2} + \frac{l_4s(\theta_1 + \theta_2 + \theta_3 + \theta_4)}{2} \\ & + \frac{l_1s(\theta_1 + \theta_2) + l_1s(\theta_1 - \theta_2)}{2} + \frac{l_3s(\theta_1 - \theta_2 - \theta_3 - \theta_4)}{2} \\ & + \frac{l_4s(\theta_1 - \theta_2 - \theta_3 - \theta_4)}{2} \\ & + \frac{l_2s(\theta_1 - \theta_2 - \theta_3) + l_2s(\theta_1 + \theta_2 + \theta_3)}{2} \\ & - c(\theta_1)(e_1 + e_2 + e_3 + e_4 + e_5) \end{aligned} \quad (5)$$

$$P_z = h_1 + h_2 + l_2s(\theta_2 + \theta_3) + l_1s(\theta_2) + l_3s(\theta_2 + \theta_3 + \theta_4) + l_4s(\theta_2 + \theta_3 + \theta_4) \quad (6)$$

3.2 Inverse kinematics

The inverse kinematics problem involves the finding of the joint parameters according to the position and orientation values of the end effector. The solutions of this problem are closely related to the design of the manipulator. For this reason, the solution methods can differ from one mechanism to another one. If the structure of the manipulator does not allow an analytical solution to this problem, this problem can be solved by iterative methods such as Newton-Raphson [25] and Sequential Monte Carlo [26] methods. The serial robot manipulator designed in this study has some offsets such as e values in its joints. In addition, there are four unknown variables ($\theta_1, \theta_2, \theta_3, \theta_4$) for the three kinematic equations and only P_x, P_y and P_z are given for the inverse kinematic solution as seen in Eqs. (4), (5) and (6). For this reason, the inverse kinematic problem of this manipulator cannot be completely solved analytically. In order to handle this problem, a combination of analytical method and a basic search technique is used. In this method, first, the θ_1 angle is analytically calculated. Then, a simple search cycle is initiated in which all values of θ_2 angle are tested with a sensitivity of 0.01 degrees. Then, the values of θ_3 and θ_4 are analytically calculated in each iteration of this loop, and in this way, the inverse kinematic problem of the robot is solved. The value of θ_5 can be any value that can be determined by the user. This angle just changes the orientation of the end effector. With the help of this joint, the robot end effector can pick or place the material to be moved in the desired orientation. As mentioned above; the analytical solution for θ_1 can be obtained as follows; firstly, ${}^0T = {}_{end_effector}^{base}T$ matrix can be written as in Eq. (7).

$${}^0T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & x \\ r_{21} & r_{22} & r_{23} & y \\ r_{31} & r_{32} & r_{33} & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

Where r_{ij} $i, j = 0,1,2$ represents the rotational elements of the transformation matrix, while x, y and z are the elements of the position vector of the end effector. The inverse kinematic equations of the manipulator can be obtained by multiplying

the inverse of the transformation matrices respectively given in Eq. (2) [8];

$$[{}^0T]^{-1} {}^0T = [{}^1T]^{-1} {}^1T_2^{-1} {}^2T_3^{-1} {}^3T_4^{-1} {}^4T_5^{-1} {}^5T_6^{-1} {}^6T_7^{-1} {}^7T_8^{-1} {}^8T_9 \quad (8)$$

here, $[{}^0T]^{-1}$ can be written as follows;

$$[{}^0T]^{-1} = \begin{bmatrix} c(\theta_1) & s(\theta_1) & 0 & 0 \\ -s(\theta_1) & c(\theta_1) & 0 & 0 \\ 0 & 0 & 1 & -98 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9)$$

If we substitute and rearrange Eq. (9) in Eq. (8) and then compare the corresponding matrix elements on both sides of the equation reciprocally, the following expression can be written for the angle θ_1 [27].

$$yc(\theta_1) - xs(\theta_1) = 181/50 \quad (10)$$

The θ_1 angle can be obtained from this equation as follows;

$$\theta_1 = \arctan2(-x, y) + \arctan2\left(\mp\sqrt{x^2 + y^2 - (181/50)^2}, (181/50)\right) \quad (11)$$

It should be noted that there are two solutions for the Equation 11 and the decision about selection of one of these two solutions is given by the developed user interface program according to the current end-effector position. As a second step, the simple search method is used for the angle θ_2 to find other angle values (Figure 7). The limits of the angle θ_2 are given in Table 1 as -3.19° and maximum 146.831° . The minimum amount of motion of the servomotor used for θ_2 is 0.01° . Using these parameters, θ_4 and θ_3 can be found for each value of the θ_2 angle as follows [27].

$$\theta_4 = \arctan2\left(\left(\sqrt{1 - \left(\frac{(xc(\theta_1) + ys(\theta_1) - 125c(\theta_2))^2}{2.101.104}\right)^2}\right), \left(\frac{(xc(\theta_1) + ys(\theta_1) - 125c(\theta_2))^2}{2.101.104}\right)\right) \quad (12)$$

$$\theta_3 = \arctan2\left(\begin{aligned} & ((101c(\theta_4) + 104)101s(\theta_1 + \theta_4) + 104s(\theta_1) \\ & - 101s(\theta_4)101c(\theta_1 + \theta_4) \\ & + 104c(\theta_1), (101c(\theta_4) + 104)101c(\theta_1 + \theta_4) \\ & + 104c(\theta_1) + 101s(\theta_4)101s(\theta_1 + \theta_4) \\ & + 104s(\theta_1) - (\theta_2)) \end{aligned}\right) \quad (13)$$

In the equation, θ_1 can be calculated as follows;

$$\theta_1 = \arctan2\left(\begin{aligned} & ((z - 163)c\theta_4 + (xc(\theta_1) + ys(\theta_1))) \\ & ((z - 163) - 125s(\theta_2)) \\ & - 101s(\theta_4)((xc(\theta_1) + ys(\theta_1)) - 125c(\theta_2)), \\ & ((z - 163)c\theta_4 + (xc(\theta_1) + ys(\theta_1))) \\ & ((xc(\theta_1) + ys(\theta_1)) - 125c(\theta_2)) \\ & + 101s\theta_4((z - 163) - 125s(\theta_2)) \end{aligned}\right) \quad (14)$$

Finally, since the angle θ_5 is only used for the direction of the end effector and it is independent of the position of the end effector, it can take any value. The flowchart of the proposed method was given in Figure 7, [27].

3.3 Applications of inverse kinematics

In this section, the designed serial robot mechanism was tested by using two different trajectories. The first one is an arc-

shaped with 58 points, and the other is a linear trajectory that is divided into 29 points. These trajectories were shown in Figure 8. In order to see the trajectories drawn by the robot, the center of the robot base was located to the origin (0, 0) of a Cartesian coordinate system on a paper that was used for the trajectory drawn performed by the robot (Figure 9).

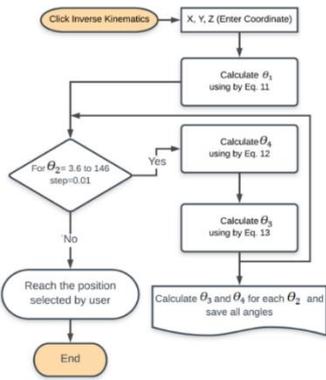


Figure 7. Flowchart of the proposed method.

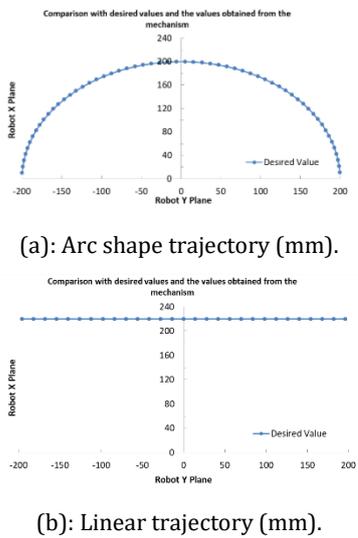


Figure 8. Desired trajectory points.

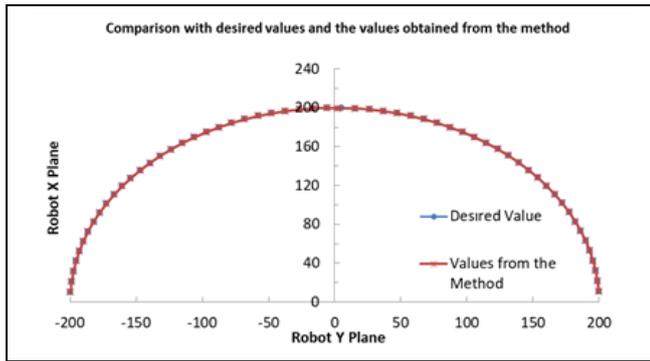


Figure 9. Experiments set-up.

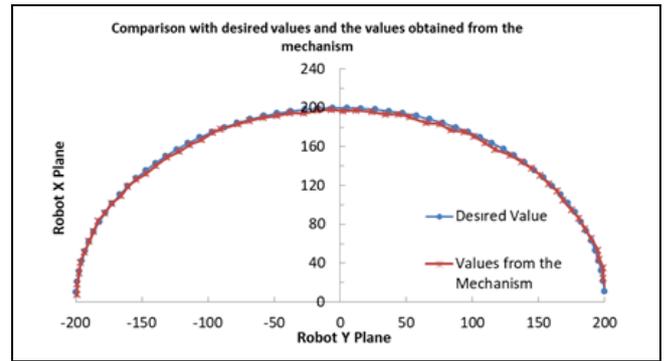
The points of each trajectory were defined as the position of the end effector of the manipulator and the inverse kinematic problem was solved through the method presented in the previous section. The joint angle parameters obtained by this method were applied to the serial robot mechanism and the obtained results were drawn by a pencil used as the end effector of the mechanism. It should be noted that the inverse kinematics solution of the trajectory points is obtained while the mechanism is following that trajectory. As the mechanism is moving between two points, the inverse kinematics solution for the next point is being calculated. Therefore, a nearly continuous moving is provided. The test setup used was shown in Figure 9. The inverse kinematic results obtained by the proposed method are presented in Table 4 for the arc-shaped trajectory and in Table 5 for the linear trajectory. In addition, in order to evaluate the application results of these solutions, both of the trajectories obtained by the proposed method and the trajectories drawn by the test system of the robot mechanism are drawn on the same graphics together. These were shown in Figure 10.

According to the Figure 10a and Figure 10b it can be seen that the proposed method ensures the complete solution of the inverse kinematic problem of the mechanism. And according to the Figure 10c and Figure 10d, it is seen that the serial robot mechanism tracks the arc-shaped trajectory with a maximum error of 4.35 mm and linear trajectory with a maximum error of 2.55 mm. In light of the obtained results, although the position error of the end effector is at millimetre level, it can be discussed that these errors are related to the design of the mechanism rather than the method of inverse kinematic solution. In this context, the design sensitivity is vital especially in serial robots because the errors appearing at every joint affect the end effector collectively. The execution time of the trajectory of the mechanism is very important regarding the control process of a robot mechanism. For this reason, a second experiment was carried out to determine the execution time values of each trajectory. Ten examples of time values obtained for arc-shaped and linear trajectories were given in Tables 6 and 7, respectively. All experiments were conducted using a computer with an Intel® Core™ i5-2450M 2.50 GHz CPU and 4 GB of RAM. According to the obtained results, it is seen that an inverse kinematic problem that cannot be solved analytically can be solved by the proposed method approximately in a 1 second for an arc-shaped trajectory and 0.618 seconds for a linear trajectory. It should be noted that these time values are the values obtained for solving the inverse kinematic problem of the entire trajectory. Based on these results, the average inverse kinematic solution time for a trajectory point is 18.6 ms.

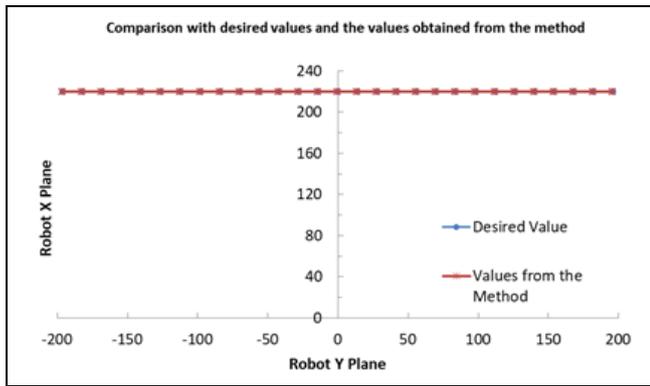
As a last validation of the proposed method, two error analysis methods, Mean Squared Error (MSE) and Root Mean Square Error (RMSE) were used. The error analyses were performed for both of the results of the proposed method and the results obtained by the mechanism. The MSE and RMSE were calculated for the X and Y coordinates of the trajectory points separately. The obtained results are given in Table 8. According to the results, it can be seen that the proposed method followed the coordinates of both trajectories in perfect. In contrast to the proposed method, the mechanism followed the coordinates of both trajectories with some error (Table 8). In light of these results, it can be said that the errors emerged during the robot's tracking the trajectories are because of the mechanism structure rather than the proposed method.



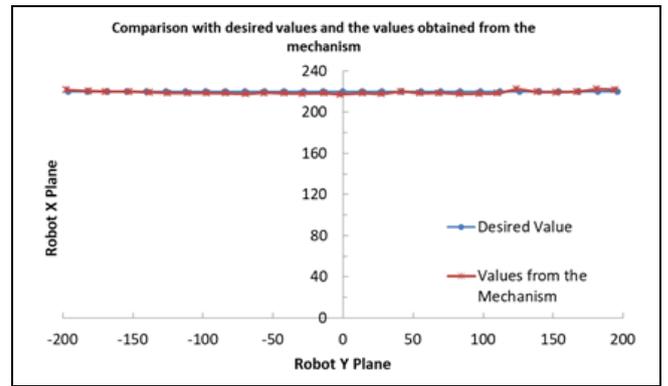
(a)



(c)



(b)



(d)

Figure 10(a); and (b): Trajectory points of the arc shaped and linear trajectories and the inverse kinematic solutions for these points obtained by the proposed method, respectively; (c); and (d): Trajectory points of the arc shaped and linear trajectories and the inverse kinematic solutions for these points drawn by the mechanism, respectively.

Table 4. Inverse kinematic solution for the arc shaped trajectory.

No	Arc Points					
	Trajectory points (mm)		Inverse kinematic solution (Degree)			
	X plane	Y plane	Base (θ_1)	Shoulder (θ_2)	Elbow (θ_3)	Wrist (θ_4)
1	10.46	-199.73	-88.039	43.57	-84.341	-31.568
2	21.08	-198.89	-84.987	43.57	-84.341	-31.568
3	31.63	-197.48	-81.937	43.57	-84.339	-31.576
4	42.1	-195.52	-78.886	43.57	-84.34	-31.571
5	52.54	-193	-75.808	43.57	-84.347	-31.545
6	62.64	-189.94	-72.785	43.57	-84.34	-31.57
7	72.66	-186.33	-69.734	43.57	-84.338	-31.577
8	82.84	-182.2	-66.587	43.56	-84.356	-31.447
9	92.06	-177.56	-63.632	43.57	-84.342	-31.565
10	101.38	-172.4	-60.579	43.57	-84.339	-31.573

Table 5. Inverse kinematic solution for the linear trajectory.

No	Linear Points					
	Trajectory points (mm)		Inverse kinematic solution (Degree)			
	X plane	Y plane	Base (θ_1)	Shoulder (θ_2)	Elbow (θ_3)	Wrist (θ_4)
1	220	-196	-42.402	14.4	-45.909	-1.305
2	220	-182	-40.326	18.92	-53.929	-0.64
3	220	-168	-38.116	22.72	-60.482	-0.533
4	220	-154	-35.764	26	-66.118	-0.522
5	220	-140	-33.267	28.86	-70.584	-1.458
6	220	-126	-30.619	31.38	-75.046	-1.225
7	220	-112	-27.82	33.59	-79.014	-0.922
8	220	-98	-24.872	35.51	-82.181	-1.24
9	220	-84	-21.779	37.14	-84.246	-2.812
10	220	-70	-18.549	38.37	-84.284	-7.301

Table 6. The results of inverse kinematics and time values for finding these results (arc shaped trajectory) (only 10 samples).

No	Arc Points				Time (millisecond)
	Desired values (mm)		Obtained values from the mechanism (mm)		
	X plane	Y plane	X plane	Y plane	
1	10.46	-199.73	7.45	-198.7	00:00.599
2	21.08	-198.89	18.39	-198.7	00:00.660
3	31.63	-197.48	29.38	-197.39	00:00.670
4	42.1	-195.52	40.09	-196.35	00:00.515
5	52.54	-193	51.17	-193.34	00:00.508
6	62.64	-189.94	62.55	-189.75	00:00.524
7	72.66	-186.33	72.32	-186.41	00:00.568
8	82.84	-182.2	83.23	-183.02	00:00.568
9	92.06	-177.56	90.93	-178.32	00:00.518
10	101.38	-172.4	101.52	-172.78	00:00.577

Table 7. The result of inverse kinematics and time values for finding these results (linear trajectory) (only 10 samples).

No	Linear Points				Time (millisecond)
	Desired values (mm)		Obtained values from the mechanism (mm)		
	X plane	Y plane	X plane	Y plane	
1	220	-196	221.56	-196.87	00:01.010
2	220	-182	220.59	-180.93	00:01.081
3	220	-168	220.06	-168.89	00:01.022
4	220	-154	219.91	-152.17	00:01.199
5	220	-140	219.27	-137.72	00:01.250
6	220	-126	218.65	-124.7	00:01.117
7	220	-112	218.53	-110.43	00:00.860
8	220	-98	218.5	-96.67	00:00.742
9	220	-84	218.2	-83.09	00:00.833
10	220	-70	217.77	-68.9	00:00.783

Table 8. The error analysis of the purposed method and the mechanism.

Trajectory	The method of error analysis	MSE (X axis coordinates of the robot)	MSE (Y axis coordinates of the robot)
Linear shaped (theoretical*)	MSE	0	0
	RMSE	0.001	0.001
Arc shaped (theoretical*)	MSE	0	0
	RMSE	0.005	0.001
Linear shaped (applied**)	MSE	1.777	1.920
	RMSE	1.333	1.386
Arc shaped (applied**)	MSE	2.885	0.355
	RMSE	1.698	0.578

*: Theoretical depict the results calculated for the inverse kinematic solutions obtained by the proposed method.

** : Applied is used to specify the results calculated for the drawn trajectories by the mechanism.

4 Results and discussion

In this study, a 5-DOF serial robot mechanism was designed and implemented. The inverse kinematic problem of the mechanism was solved by using the analytical method and simple search technique together. In addition, a user interface program was written through the Visual Basic programming language to control the developed mechanism. Finally, the proposed inverse kinematic solution method was tested through an arc-shaped trajectory composed of 58 points and a linear trajectory divided into 29 points. The validation of the proposed method is performed by using two error analysis methods, MSE and RMSE.

As an overall evaluation, it can be said that the proposed method is successfully used to solve the inverse kinematics problem of the designed mechanism in real-time applications. Moreover, the proposed method can also solve the inverse kinematics problem of the mechanism while it is working. Finally, it can be noted that the proposed method can be easily

adapted to the serial robots that have the same mechanical structures.

5 Abbreviations

DOF	:	Degrees of freedom,
R	:	Rotary joint,
ELM	:	Extreme learning machine,
DH	:	Denavit-hartenberg,
s	:	Sine,
c	:	Cosine,
Px, Py, Pz	:	Cartesian coordinates,
θ	:	Joint angle.

6 References

- [1] Yildiz I. "A low-cost and lightweight alternative to rehabilitation robots: omnidirectional interactive mobile robot for arm rehabilitation". *Arabian Journal for Science and Engineering*. 43(3), 1053-1059, 2018.

- [2] Srairi F, Saidi L, Hassam A. "Modelling control and optimization of a new swimming microrobot using flatness-fuzzy-based approach for medical applications". *Arabian Journal for Science and Engineering*, 43(6), 3249-3258, 2018.
- [3] Luo L, Tang Y, Lu Q, Chen X, Zhang P, Zou X. "A vision methodology for harvesting robot to detect cutting points on peduncles of double overlapping grape clusters in a vineyard". *Computers in Industry*, 99, 130-139, 2018.
- [4] Li Y, Zhu H. "A simple optimization method for the design of a lightweight, explosion-proof housing for a coal mine rescue robot". *Journal of the Brazilian Society of Mechanical Sciences and Engineering*. 40(7), 1-10, 2018.
- [5] Craig JJ. "Introduction to robotics: mechanics and control". Addison-Wesley Publishing Company, 113-144, Boston, USA, 1989.
- [6] Koker R. "A genetic algorithm approach to a neural-network-based inverse kinematics solution of robotic manipulators based on error minimization". *Information Sciences*, 222, 528-543, 2013.
- [7] Fu Z, Wenyu Y, Zhen Y. "Solution of inverse kinematics for 6r robot manipulators with offset wrist based on geometric algebra". *Journal of Mechanisms and Robotics*, 5(3), 310081-310087, 2013.
- [8] Kucuk S, Bingul Z. "The inverse kinematics solutions of fundamental robot manipulators with offset wrist". *IEEE International Conference on Mechatronics*, Taipei, Taiwan, 10-12 July 2005.
- [9] Liu ZZ, Liu HY, Luo Z. "Inverse kinematics analysis of 5 DOF robot manipulators based on virtual joint method". *Applied Mechanics and Materials*, 143, 265-268, 2012.
- [10] Sariyildiz E, Cakiray E, Temeltas H. "A comparative study of three inverse kinematic methods of serial industrial robot manipulators in the screw theory framework". *International Journal of Advanced Robotic Systems*, 8(5), 9-24, 2011.
- [11] Park J, Kim J M, Park HH, Kim J W, Kang G H, Kim S H. "An iterative algorithm for inverse kinematics of 5-DOF manipulator with offset wrist". *World Academy of Science, Engineering and Technology*, 72(6), 12-25, 2012.
- [12] Sheng L, Yiqing W, Qingwei C, Weili H. "A new geometrical method for the inverse kinematics of the hyper-redundant manipulators". *Robotics and Biomimetics, ROBIO'06*, Waikoloa Village, HI, USA, 16-18 September 2006
- [13] El-Sherbiny A, Elhosseini M A, Haikal AY. "A comparative study of soft computing methods to solve inverse kinematics problem". *Ain Shams Engineering Journal*, 9(4), 2535-2548, 2018.
- [14] Chaichawanant J, Saiyod S. "Solving inverse kinematics problem of robot arm based on a-star algorithm". *13th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, Khon Kaen, Thailand, 13-15 July 2016.
- [15] Feng Y, Yao-nan W, Yi-min Y. "Inverse kinematics solution for robot manipulator based on neural network under joint subspace". *International Journal of Computers Communications & Control*, 7(3), 459-472, 2012.
- [16] Kucuk S, Bingul Z. "Inverse kinematics solutions for industrial robot manipulators with offset wrists". *Applied Mathematical Modelling*, 38(7-8), 1983-1999, 2014.
- [17] Aristidou A, Lasenby J. "FABRIK: A fast, iterative solver for the inverse kinematics problem". *Graphical Models*, 73(5), 243-260, 2011.
- [18] Toz M. "Inverse kinematic solution of a 6 DOF serial robot manipulator with offset wrist by using alo algorithm". *Sigma Journal of Engineering and Natural Sciences*, Special Issue ASYU 2016, 8(2), 81-90, 2017.
- [19] Dereli S, Köker R. "IW-PSO approach to the inverse kinematics problem solution of a 7-DOF serial robot manipulator". *Sigma Journal of Engineering and Natural Sciences*, vol. 36(1), 77-85, 2018.
- [20] Dereli S, Köker R. "A meta-heuristic proposal for inverse kinematics solution of 7-DOF serial robotic manipulator: quantum behaved particle swarm algorithm". *Artificial Intelligence Review*, 53, 949-964, 2020.
- [21] Li T, Yao P, Luo M, Tan Z, Wang M, Guo Z. "Design and kinematics analysis of a novel six-degree-of-freedom serial humanoid torso". *International Journal of Advanced Robotic Systems*, 15(1), 1-10, 2018.
- [22] Hrdina J, Návrát A, Vašík P. "Notes on planar inverse kinematics based on geometric algebra". *Advances in Applied Clifford Algebras*, 28(3), 71-84, 2018.
- [23] Sandoval J, Nouaille L, Poisson G, Parmantier Y. "Kinematic design of a lighting robotic arm for operating room". *Computational Kinematics, Mechanisms and Machine Science*, 50, 44-52, 2018.
- [24] Uzuner S, Akkuş N, Toz M. "Trajectory planning of a 5-DOF serial robot manipulator in joint-space". *Journal of Polytechnic*, 20(1), 151-157, 2017.
- [25] Haug EJ. *Computer Aided Kinematics and Dynamics of Mechanical Systems*. 1st ed. Boston, USA, Allyn and Bacon Boston 1989.
- [26] Courty N, Arnaud E. "Sequential Monte Carlo Inverse kinematics". *Institute National Polytechnique de Grenoble, Avenue Félix Viallet, Grenoble, France Research Report*, 6426, 24, 2008.
- [27] Uzuner S. A New Approach to The Solution of Inverse Kinematics of Industrial Robots with Offset Wrist. MSc Thesis, Marmara University, İstanbul, Turkey, 2012.