



Inverse kinematic of European Robotic Arm based on a new geometrical approach

A. Zamanzadeh, H. Ahmadi*

Faculty of Mechanical Engineering, Shahrood University of Technology, Shahrood, Iran

ABSTRACT: In this study, inverse kinematics is solved for the European Robotic Arm. This robot is a type of space manipulator and has seven degrees of freedom. Forward and inverse kinematic is obtained by a new geometrical method. The limitations of robot workspace are calculated for robot joints and grippers. A geometrical inverse kinematic is presented for the first time, and the transferring process for various situations is developed to grasp the object at any position and orientation. Considering multiple missions, for robots in the transferring process, there isn't a unit method to support all the situations and tasks in the inverse kinematic problem. To determine the workspace of robot, its geometry and object situation (orientation and position) are considered. To this end, using the suggested inverse kinematic algorithm, the target coordinate and orientations are obtained. In the presented inverse kinematic algorithm, an analytical method is used to derive the joint space variables in terms of workspace variables. The European Robotic Arm can move step by step from one target point to the next one. So, a general transferring algorithm is presented to realize the robot's mission. In the presented algorithm, when the target is unreachable by the one-step operation, the transferring mission is utilized by the robot. Some simulation plans, to validate the proposed algorithms, indicate that the presented method works correctly.

Review History:

Received: 01/01/2020
Revised: 17/05/2020
Accepted: 21/06/2020
Available Online: 07/07/2020

Keywords:

The European Robotic Arm
Inverse Kinematics
Transferring Operation
7 Dof Robotic Arm
Space Manipulator

1. INTRODUCTION

Multi degrees of freedom space manipulators perform a significant role in the maintenance, repairing, and upgrading of space stations and satellite equipment. One of these manipulators, the European Robotic Arm (ERA), was presented by Crujjsen et al. [1]. It is an enormous movable robotic arm with 7 degrees of freedom. The ERA is utilized to install and repair equipment in the space stations. This robot is equipped with two end-effectors, two wrists, and two limbs. Both of the end-effectors can operate as a gripper to grasp the objects. This robotic arm can move to the mission's region for performing various activities.

The structure, locomotion mechanisms, and capabilities of the European Robotic Arm were described by Crujjsen et al. [1]. Lambooy et al. [2] presented some of the overall design aspects for various mechanisms of ERA. The efficiency of these robots in the space stations was analyzed by Boumans Heemskerk [3]. Feng [4] presented some studies relative to the design of the robot gripper and its applications. The inverse kinematic is an essential subject to control the robotic arm. Although some studies have been presented for ERA, they have not investigated the kinematic and dynamic analysis of this robot, whereas, the kinematic and dynamic analysis for different similar manipulator robots have been studied. The inverse kinematic problem of the robotic arm, industrial or non-industrial, fixed or mobile, was investigated by Romanelli

[5]. The limitations of the robot workspace were investigated by Cheein et al. [6]. Analytical inverse kinematics solver for anthropomorphic 7-DOF redundant manipulators ("Kuka-LWR" arm) with human-like configuration constraints was presented by Liu et al. [7]. The inverse kinematics of the "GMF a-510" robot was introduced by Kendrick [8] through the vectors Groebner basis theory and the "Denavit-Hartenberg" method. In another study in this field, Kendrick [9] addressed the advantage of the "Groebner Basis" theory. The kinematics and dynamics of the 6 degrees of freedom industrial "KUKA" robot were investigated by Díaz et al. [10]. In this research, for the kinematic and dynamics robot, the "Denavit-Hartenberg" method was used. In another study, Lee and Ziegler [11] presented various possible arm configurations for "PUMA" manipulators to solve the kinematic problem based on human arm geometry. A study on kinematics and workspace determination of a general 6-P US robot was investigated by Nabavi et al. [12]. In this work, a recently constructed 6-UP S mechanism for a metro station was selected, and an optimization method based on a Genetic Algorithm (GA) is used. Also, a new concept for 6-DOF workspace visualization representation named "workspace spheres" is presented. In the field of robotic arms, some researches have been focused on the gripper design; Xie [13] addressed the kinematics and dynamics of six degrees of freedom manipulators like industrial robots. The inverse kinematic problem of a 6-axis robot manipulator, consist of two

*Corresponding author's email: habibahmadif@shahroodut.ac.ir



different configurations grippers, was solved through proper geometrical precisions by Shahinpoor et al. [14]. Kinematic and inverse kinematics of a 6-DOF manipulator were realized through the D-H method by Sun et al. [15]. Concerning the theory of screws, another method was presented by Xie et al. [16] to solve the inverse kinematic problem for the 6-DOF space manipulator. Based on the main feature of the space robot, in comparison with other manipulators, these robots can perform the transferring operation. Sagara and Taira [17] studied a comprehensive study for kinematic, dynamic, and control of space floating robot, a manipulator with 6 degrees of freedom like the ERA. Duleba [18] addressed kinematic models of doubly generalized N-trailer systems. In this work, a Pfaff matrix for doubly generalized N-trailer systems was obtained based on the longitudinal limitations. For all active limitations, closed-form kinematics was extracted, while for other limitations, a recursive one was proposed. A modified procedure for solving the mission-oriented inverse kinematic of hyper-redundant space manipulators was proposed by Xu et al. [19]. For a spraying robot, Li et al. [20] synthesized a new mobile robot employing a telescopic arm. The workspace was enlarged and reduced because of the stretchable structure. An analytic inverse kinematic solution was proposed by Oh et al. [21] to consider joint limitations and self-collision avoidance for a redundant 7DOF manipulator with a spherical shoulder. Xu et al. [22] investigated the modeling and planning of a space robot for capturing the tumbling target by approaching the Dynamic Closest Point (DCP). Ghaedrahmati et al. [23] improved the efficiency of multibody systems by merged a unified constraint transfer matrix for all types of joints. The absolute interface coordinate was used by Ellenbroek and Schilder [24] as degrees of freedom. For this purpose, a coordinate transformation is presented from the absolute floating frame coordinate as the joints' local coordinate to the joints' absolute coordinate. Müller [25] addressed screw theory with the geometric setting and Lie group theory to substructure analysis of an intuitive and compact MBS modeling. Principal methods of forward and inverse kinematics for multibody dynamics analysis and kinematic synthesis were investigated by Komoda [26] to identify the arbitrary end-effector coordinate.

Having these facts in mind, there is no work on the inverse kinematic algorithm for the ERA robot. This is the main motivation of the current research work. Therefore, the contributions of this paper are as follows: (1) the inverse kinematic algorithm of ERA robot is presented for the first time, (2) this algorithm is performed by a geometrical method with a step by step approach to grasp an object with any desired positions and orientations, (3) an algorithm is developed for transferring operation for the first time, such that the robot gripper can grasp the object, which isn't placed in the robot workspace, and (4) two methods, through inverse-forward kinematic and a MATLAB Simscape model, are developed to validate the proposed algorithm. To do this, forward kinematic and all related problems are obtained utilizing the rotation matrices in the 3D coordinate. Two overall steps are considered for the inverse kinematic algorithm: the constant

and movable steps. In the constant mode, one of the end-effectors is fixed, and the other one can move. To walk using two end-effectors, in the moving phase, the ability of the robot is investigated. Such that, to the right grasping of the object, the robot can perform the transferring operation.

2. MODEL DESCRIPTION

The original ERA robot is shown in Fig. 1, and the schematic diagram of this robot in this study is illustrated in Fig. 2. Each robot's wrists has three rotational degrees of freedom, and 1 DOF is considered for the elbow joint. These end-effectors are designed for operations on the space stations. This robot can climb from a building, walks on the ground, and grasp the space structures to perform different tasks, such as handling the objects, welding, or monitoring.

3. FORWARD KINEMATIC

The geometrical characteristics of the robot are shown in Fig. 3. We consider a 3D coordinate in the center of each joint, and the distance between consecutive joints is considered as a link with the length L_i . Regarding Fig. 3, point "d" shows the end-effector (i.e., endpoint of the gripper) position, two principal links length is depicted by " L_3 ", the length of the wrist is indicated by " $2L_2$ " which is the distance between joints No.1 to No.3, and the length of each gripper is shown by " L_1 ".

The kinematic analysis is extracted by the rotation matrices approach, which is presented by Ginsberg [27]. These matrices are obtained relative to the reference coordinate system O-XYZ, which is defined in Fig. 3, and shown in Table 1.

In Table 1, "S" and "C" are sinus and cosine functions, respectively. The vectors illustrated in Fig. 3, are described in Table 2. These vectors are derived utilizing the rotation matrices relative to the reference coordinate system [27].

According to Fig. 3, the vector " r_{do} " shows the end-effector position, which is used as the forward kinematic, and is presented in Appendix.

4. INVERSE KINEMATIC

The presented algorithm for inverse kinematic has two main steps as follows. Step (I) the object is located in the robot workspace, and all the necessary conditions of the inverse kinematic algorithm are satisfied. In this step, if the object has been rotated only around the z-axis, or it has been rotated 180 Degrees around x or y-axis, there is no limitation to grasp the object, by the robot gripper. So, one step operation is sufficient to grasp the object. Thus, in this case, the object can be located in any direction around the z-axis, or 180 Degrees around x or y-axis. Step (II): there are two cases, which the robot must perform the transferring operation, as presented in the following: (a) the object is placed in the robot workspace, but it has been rotated around the x or y-axis (except 180 Degrees), and (b) the object isn't placed in the robot workspace. So, one-step operation isn't sufficient to grasp the object, and the robot must be moved to a position, that can properly grasp the object, in any position

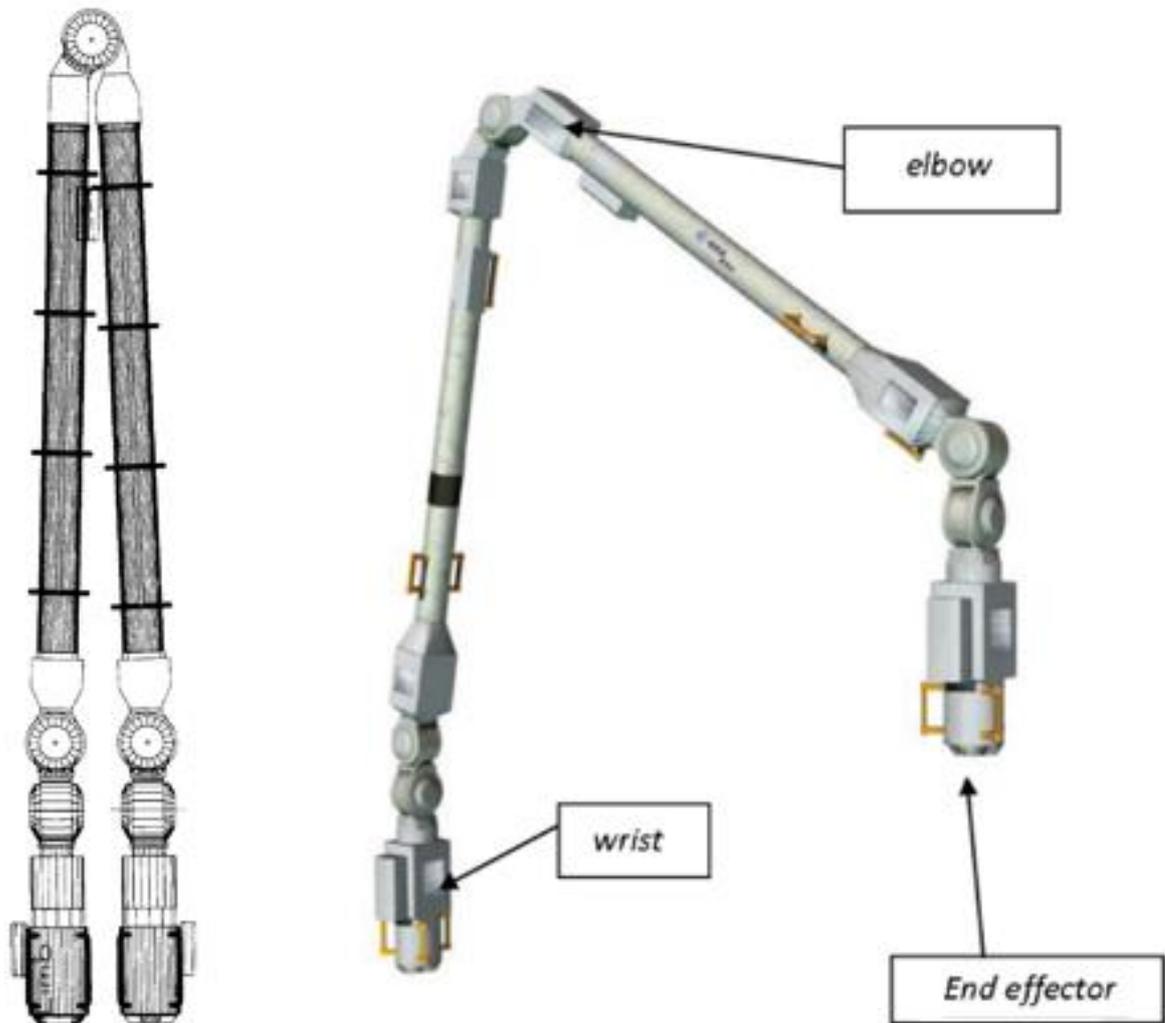


Fig. 1. European Robotic Arm [1,2]

and orientation. Although the robot has 7-DOF, in one-step operation, it needs only 6-DOF to grasp the object. In step (I), joint No. 2 is fixed, and the seventh-DOF is utilized for transferring operation, i.e., step (II). In other words, the ERA robot has two grippers with 3-DOF and 1-DOF elbow joint, i.e., in summation, the robot has 7-DOF. But, in each operation, one gripper is fixed to the platform and uses only 2-DOF, and the free gripper utilizes 3-DOF. These selections are depending on the type of mission, and are correctly set, in the inverse kinematic algorithm.

It should be noted that, although this method is specially presented for the ERA robot, this algorithm has been generally described. So, this step by step algorithm can be used for similar robots like ERA, which have specific property as follows: the robot is almost like ERA robot, but it is different in physical parameters, the robot is generally similar to ERA

robot, but has different limitations, which must be tailored for each case, and also this method can be used for a robot, that we can find its projection, in two separate planes. The last part of the algorithm, i.e., transferring operation, can be utilized for various robots.

Therefore, in this section, the inverse kinematics of the robot is investigated, and the joints angles will be obtained in terms of positions and orientations of the end-effector. It is assumed that the hook gripper is perpendicular to the object reference plane. To this end, first, the necessary conditions for the inverse kinematics algorithm are evaluated. Then, an algorithm is presented, which gives the exact solution, and according to the transferring operation this algorithm is developed. Transferring operation is the principal ability of this robot that gives a significant capability to perform many tasks by the robot. This capability allows the robot to extend

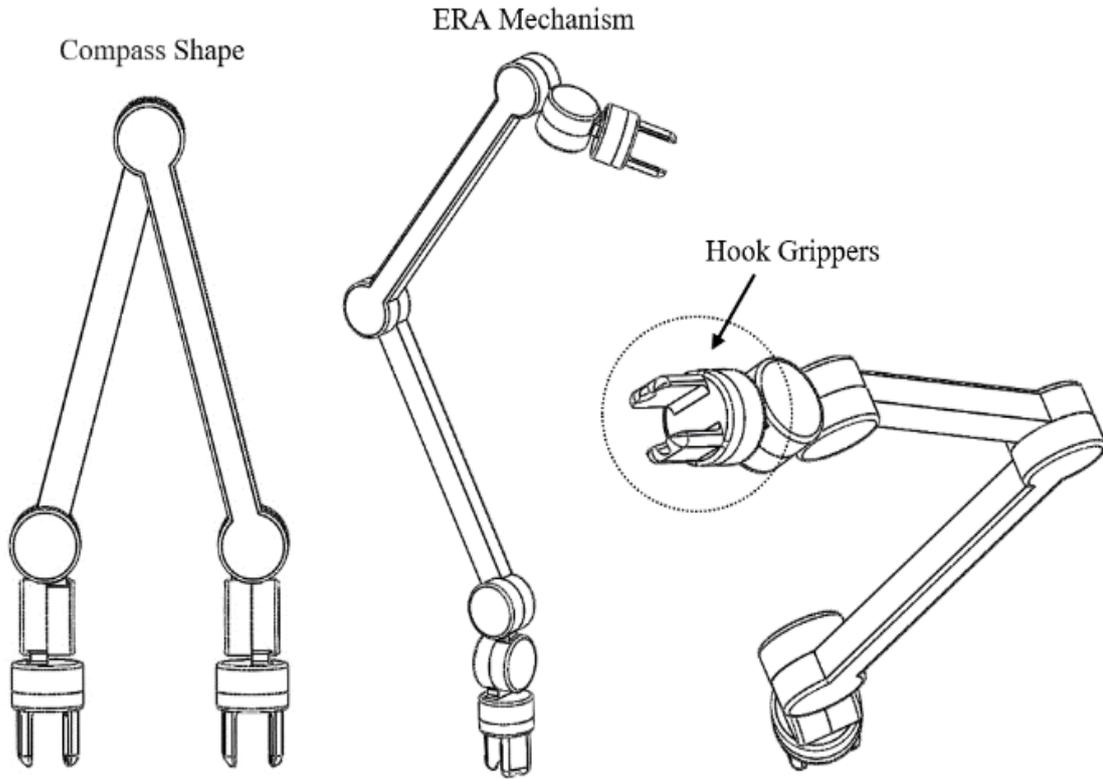


Fig. 2. Schematic diagram of a 7-DOF robotic manipulator in this study

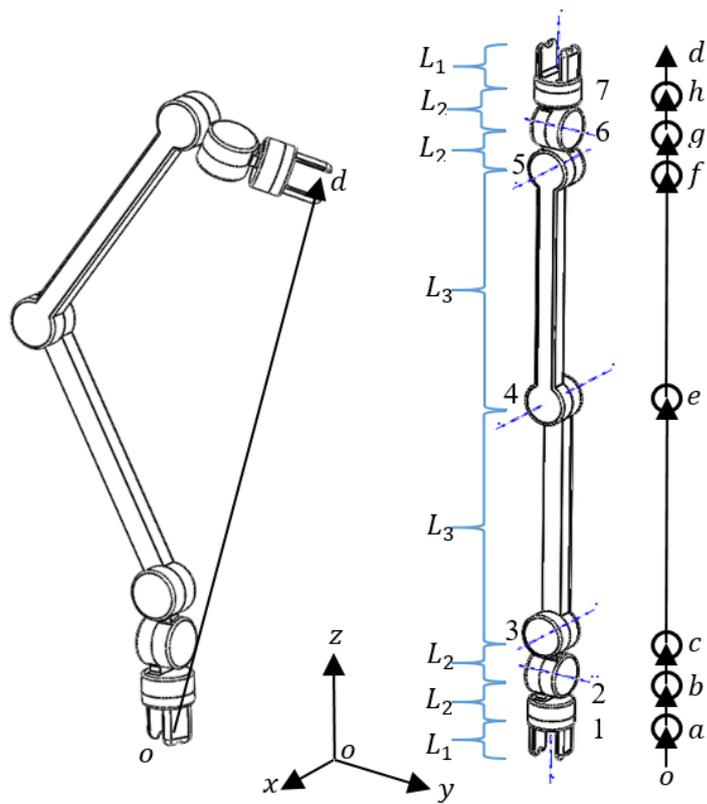


Fig. 3. Geometrical characteristics of the robot

Table 1. Rotational matrices.

| | |
|-------|--|
| R_1 | $\begin{bmatrix} C\theta_1 & S\theta_1 & 0 \\ -S\theta_1 & C\theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ |
| R_2 | $\begin{bmatrix} C\theta_2 & 0 & -S\theta_2 \\ 0 & 1 & 0 \\ S\theta_2 & 0 & C\theta_2 \end{bmatrix}$ |
| R_3 | $\begin{bmatrix} 1 & 0 & 0 \\ 0 & C\theta_3 & S\theta_3 \\ 0 & -S\theta_3 & C\theta_3 \end{bmatrix}$ |
| R_4 | $\begin{bmatrix} 1 & 0 & 0 \\ 0 & C\theta_4 & S\theta_4 \\ 0 & -S\theta_4 & C\theta_4 \end{bmatrix}$ |
| R_5 | $\begin{bmatrix} 1 & 0 & 0 \\ 0 & C\theta_5 & S\theta_5 \\ 0 & -S\theta_5 & C\theta_5 \end{bmatrix}$ |
| R_6 | $\begin{bmatrix} C\theta_6 & 0 & -S\theta_6 \\ 0 & 1 & 0 \\ S\theta_6 & 0 & C\theta_6 \end{bmatrix}$ |
| R_7 | $\begin{bmatrix} C\theta_7 & S\theta_7 & 0 \\ -S\theta_7 & C\theta_7 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ |

the workspace borders. Fig. 4 shows a flowchart, that presents all the necessary procedures of the suggested algorithm for the inverse kinematic algorithm. According to this figure, the necessary conditions of the workspace will be checked.

4.1. Workspace Limitations

Robot workspace limitations are divided into three parts, as follows: (1) joints rotation limitations, (2) the position vectors limitations, and (3) finding suitable orientations of the manipulator, regarding the limitations of links and object orientations. These limitations are presented in the following.

4.1.1. Limitations Of Rotating Joints

The rotation limitations of each joint are presented in Table 3. These limitations are considered based on the geometry

and movement ability of the robot. Joints No.1 and 7 rotate around the Z-axis without any limitations, so the mentioned joints can rotate 360 Degrees. These joints are connected to the grippers and should be had free rotations around their perpendicular's axis. Joints 3, 4, and 5 can rotate around the X-axis of the reference coordinate system. Also, joints No. 2 and 6 can rotate around the Y-axis of the reference coordinate system; however, only 6 degrees of freedom are sufficient for a one-step operation [2]. The yaw joint at the shoulder side (rotation around the Y-axis) is fixed at the initial position during the operation process, so joint No. 2 can be eliminated in "one step operation" [2]. Of course, this joint can be used in transferring operations because some missions need more than one step. The limitations of joint No.4 are illustrated in Fig. 5. We assume that the radius of all joints is the same;

Table 2 . Position vectors of joints in the reference coordinate system.

| Vectors | Specifications/value |
|----------|---|
| r_{ao} | $\begin{bmatrix} 0 \\ 0 \\ L_1 \end{bmatrix}$ |
| r_{ba} | $R_1^T \cdot \begin{bmatrix} 0 \\ 0 \\ L_2 \end{bmatrix}$ |
| r_{cb} | $R_1^T \cdot R_2^T \cdot \begin{bmatrix} 0 \\ 0 \\ L_2 \end{bmatrix}$ |
| r_{ec} | $R_1^T \cdot R_2^T \cdot R_3^T \cdot \begin{bmatrix} 0 \\ 0 \\ L_3 \end{bmatrix}$ |
| r_{fe} | $R_1^T \cdot R_2^T \cdot R_3^T \cdot R_4^T \cdot \begin{bmatrix} 0 \\ 0 \\ L_3 \end{bmatrix}$ |
| r_{gf} | $R_1^T \cdot R_2^T \cdot R_3^T \cdot R_4^T \cdot R_5^T \cdot \begin{bmatrix} 0 \\ 0 \\ L_2 \end{bmatrix}$ |
| r_{hg} | $R_1^T \cdot R_2^T \cdot R_3^T \cdot R_4^T \cdot R_5^T \cdot R_6^T \cdot \begin{bmatrix} 0 \\ 0 \\ L_2 \end{bmatrix}$ |
| r_{dh} | $R_1^T \cdot R_2^T \cdot R_3^T \cdot R_4^T \cdot R_5^T \cdot R_6^T \cdot R_7^T \cdot \begin{bmatrix} 0 \\ 0 \\ L_1 \end{bmatrix}$ |
| r_{do} | $r_{ao} + r_{ba} + r_{cb} + r_{ec} + r_{fe} + r_{gf} + r_{hg} + r_{dh}$ |

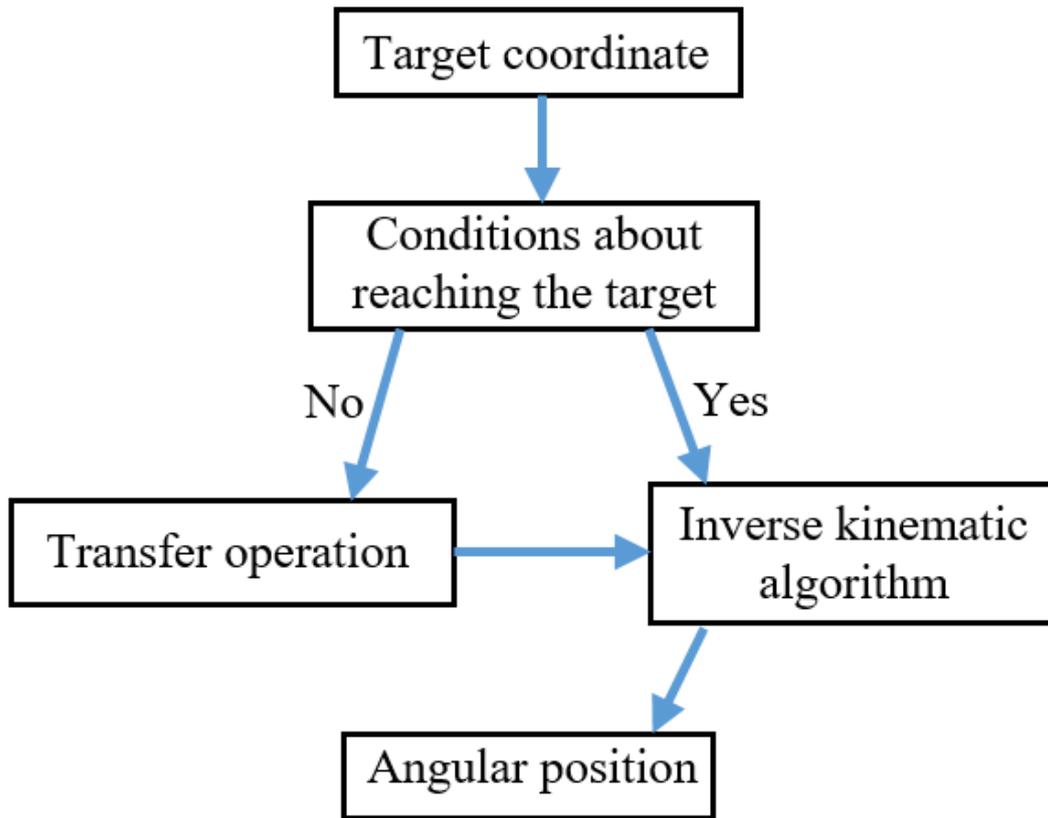


Fig. 4. Flowchart of the operation process to check the workspace conditions

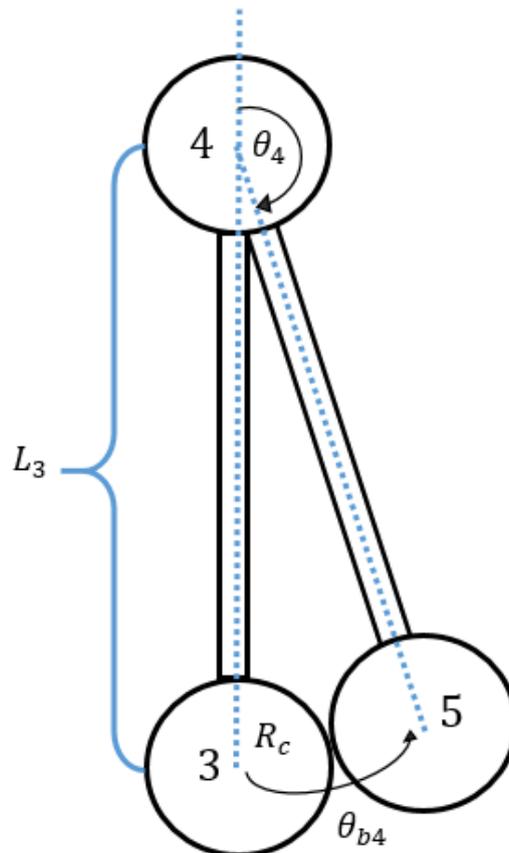


Fig. 5. Rotation limitations of joint No.4 according to joints No.3 and 5

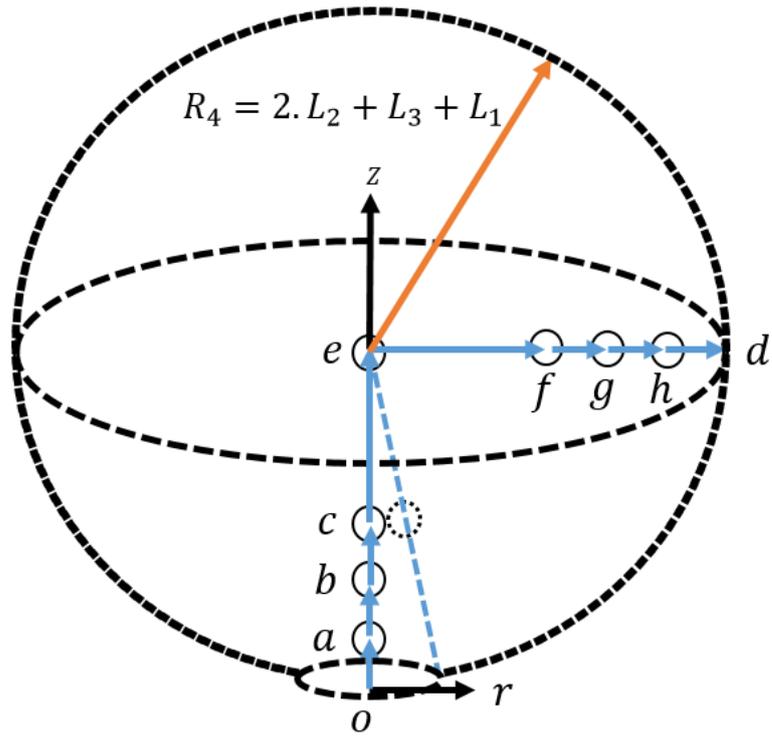


Fig. 6. Workspace boundaries for joint No.4

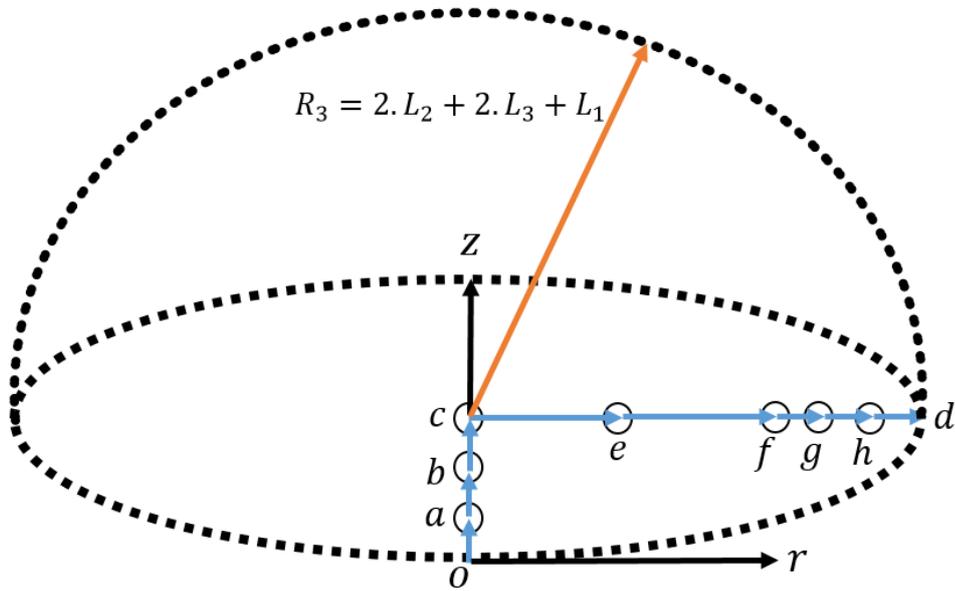


Fig. 7. Workspace boundaries for joint No.3

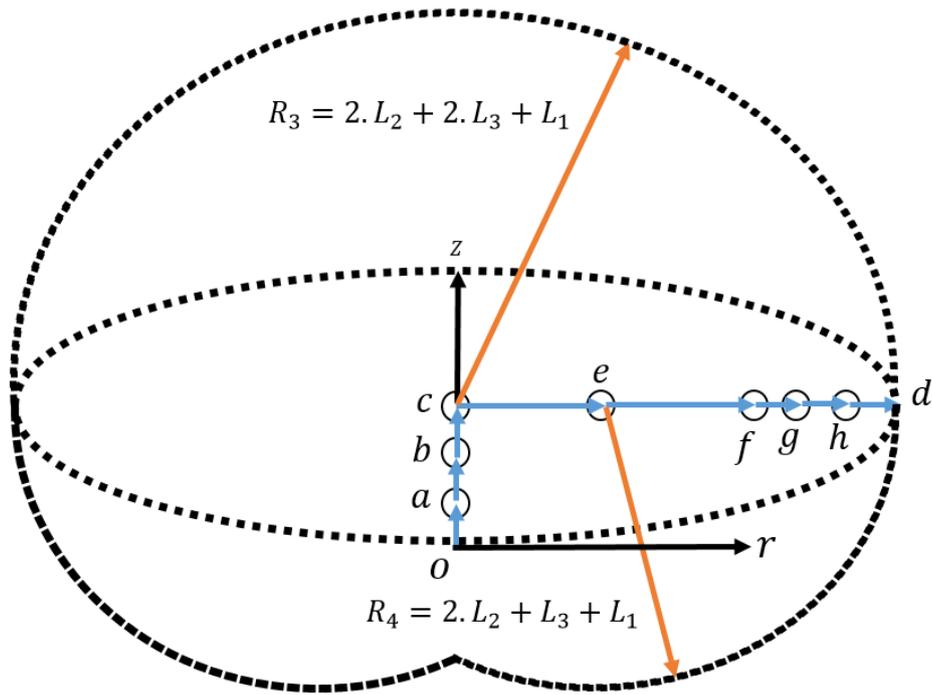


Fig. 8. The whole possible workspace of the robot

Table 3 .Joints motion ranges

| Joint | Ranges |
|------------|--|
| θ_1 | $0 \rightarrow \pm\pi$ |
| θ_2 | $0 \rightarrow \pm\frac{\pi}{2}$ |
| θ_3 | $0 \rightarrow \pm\frac{\pi}{2}$ |
| θ_4 | $0 \rightarrow \pm(\pi - \theta_{b4})$ |
| θ_5 | $0 \rightarrow \pm\frac{\pi}{2}$ |
| θ_6 | $0 \rightarrow \pm\frac{\pi}{2}$ |
| θ_7 | $0 \rightarrow \pm\pi$ |

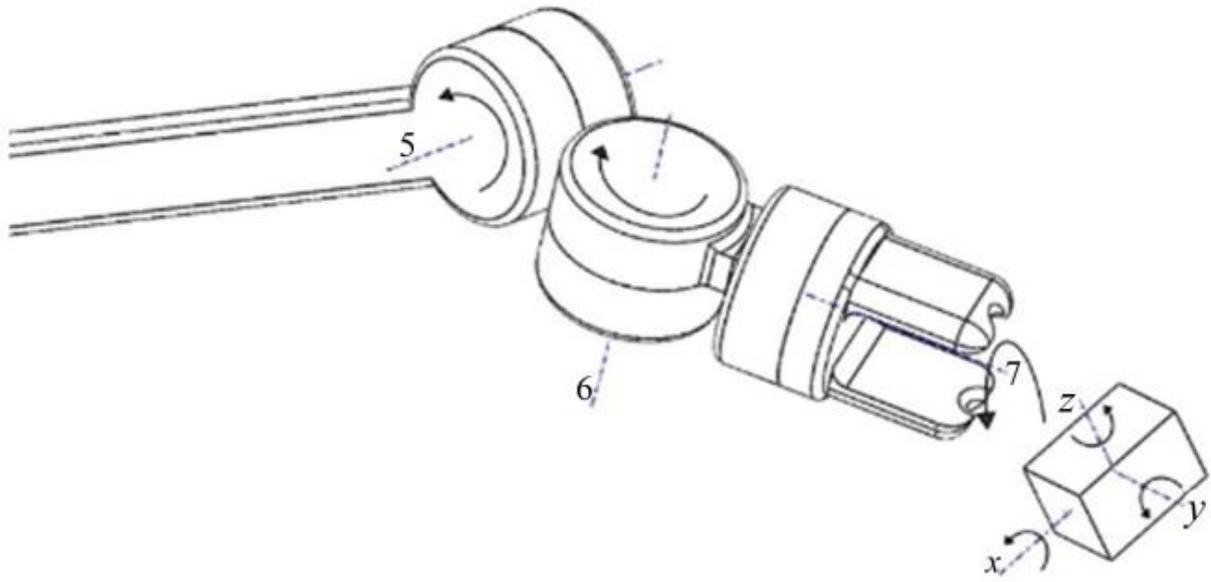


Fig. 9. Desired orientation of the object versus gripper orientation in the workspace

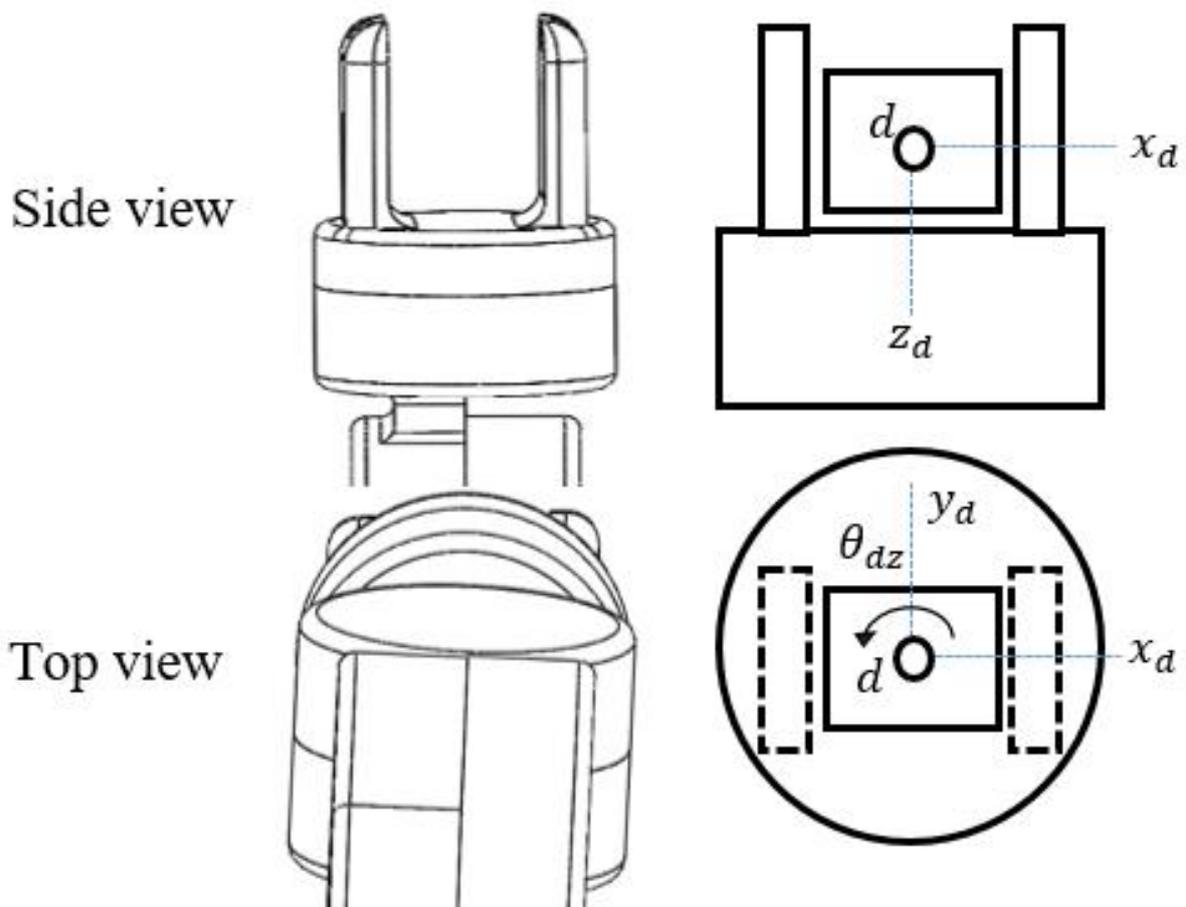


Fig. 10. Right set up to pick up an object by 2-finger gripper

therefore, regarding Fig. 5 the angle limitation for joint No.4 is given by Eq. (1).

$$\theta_{b4} = 2 \sin^{-1} \left(\frac{R_c}{L_3} \right) \quad (1)$$

It should be noted that the capital letters X, Y, and Z are for the reference coordinate system, and the smaller ones are for the object axes.

4.1.2. Position Vectors Limitations In 3d Space

Position vectors for each link are defined in Table 2, and the related rotation limitations are presented in Table 3. To determine the workspace limitations, we use Figs. (6-8). Regarding these figures, the maximum reachable space for joint No. 4, i.e. R_4 , and joint No. 3, i.e., R_3 , is shown in Fig. 6 and Fig. 7, respectively. Therefore, the external boundaries of the workspace are the summation of the reachable space by R_3 and R_4 , which is illustrated in Fig. 8.

Although all limitations are not yet considered in Fig. 8, this workspace has the least restrictions, and all the defined desired paths in this region are reachable for robot end-effectors. Regarding Fig.8, we define a new variable Z_n as

follows:

$$z_n = z - (L_1 + 2.L_2) \quad (2)$$

where Z_n is measured from point “c”.

Now, for positive Z_n , the length of position vector R_n relative to the point “c” is defined as follows:

$$R_n = \sqrt{x^2 + y^2 + Z_n^2} \quad (3)$$

Also, for the negative Z_n , the length of position vector R_m relative to the point “c” is defined as follows:

$$R_m = \sqrt{\left(\sqrt{x^2 + y^2} - L_3\right)^2 + z_n^2} \quad (4)$$

Regarding Eqs. (2-4), the entrance conditions to inverse kinematic are described in the following form:

$$\begin{cases} z_n \geq 0 \rightarrow R_n \leq R_3 \\ z_n < 0 \rightarrow R_m \leq R_4 \end{cases} \quad (5)$$

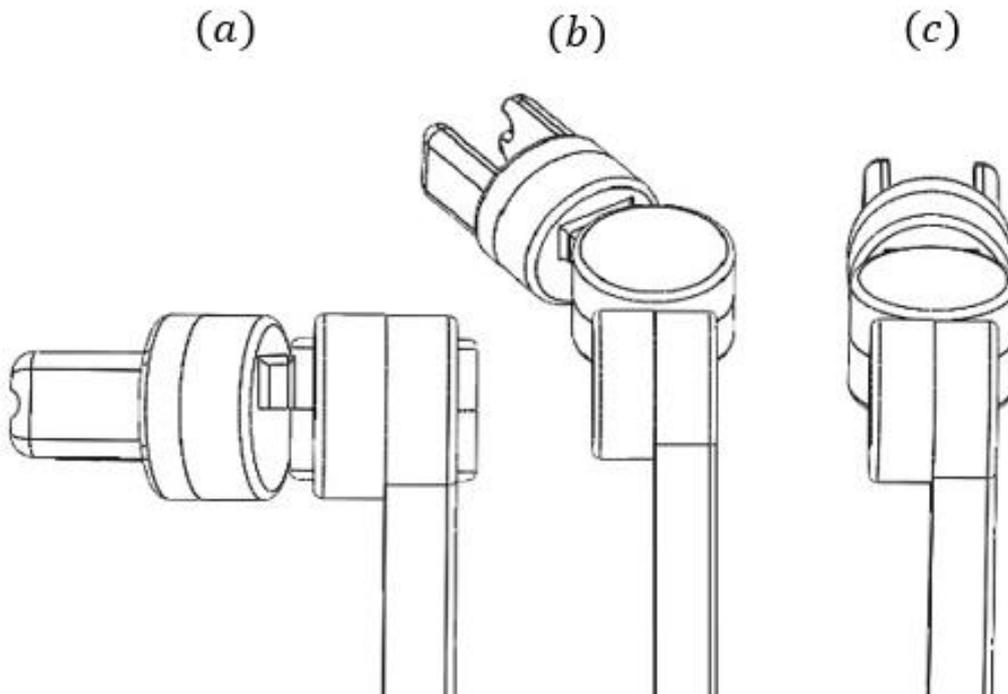


Fig. 11. The acceptable orientation of the end-effector (for more details see Fig. 28)

4.1.3. Gripper Orientation Limitations

Object (target) orientation relative to the gripper is shown in Fig. 9. Regarding this figure, the robot end-effector must be rotated such that the object can be grasped by the hooks gripper. The object orientations in the reference coordinate system are selected as the desired orientations for the gripper. Considering the limitations, mentioned in the two previous subsections, the gripper cannot grasp the object for any desired configuration in the one-step operation. Therefore, the necessary conditions for the object orientations must be defined to check whether the robot can grasp the object in the first effort or the transferring operation is essential.

To grasp the object by the gripper, the following approach is presented. Regarding the hook shape of the gripper, as shown in Fig. 10, we assume the object's shape is a cube form. Also, we suppose that the z-axis of the gripper is perpendicular to one of the object surfaces. Regarding Fig. 10, the z-axis of the object is selected in the opposite direction of the z-axis of the gripper. The motion of the gripper jaw is considered in the "x" direction of the object coordinate. Therefore, the right setup for object grasping is shown in Fig. 10.

The orientation limitations of the gripper have been considered in 2 situations. Situation (1): if the object has been rotated only around the z-axis or has been rotated 180 Degrees around x or y-axis. In this case, there is no limitation to grasp the object by the gripper. Situation (2): this case is more complicated than case 1, and the object has been rotated around the x or y-axis (except 180 Degrees). Fig. 11 indicates three cases, which can be occurred for this situation as follows. Case (a): the object has a rotation only around its y-axis, case (b): the object has rotation around both of y and x-axis, and case (c): the object has a rotation only around the x-axis. For each rotation of the object, around the x-axis and y-axis, joint No.5 and No.6 should be moved from the initial position, respectively. These cases are illustrated in Fig. 12, which shows the top view of the object in the X-Y plane. For right grasping, the object must be tangent to the dotted circle, as shown in Fig. 12. To do this, firstly, the object should be rotated around the z-axis such that it is become tangent to the dotted circle, according to Fig. 12.

Briefly, regarding Figs. (11-12), if the object isn't rotated around x or y or both of them and only is rotated around z-axis, the gripper can pick up or grasp the object without any problems. In this situation, if the object is rotated only around the z-axis, joint No.7 of gripper can make the necessary rotation to grasp the object. However, the solution exists if the dotted circle becomes tangent to the x-axis of the object. But, if the object is rotated around x or y or both of them, joints No.5 and 6 should be rotated appropriately, too. In this situation, the robot cannot reach the object in one step operation, and so the robot should be moved to another suitable position to grasp the object. In a separate section, this case is discussed in more details. According to Fig. 12, the necessary rotation relative to the X-axis around the z-axis is $\hat{\alpha}_3$. The angles θ_{dx} , θ_{dy} , and θ_{dz} are the object rotation angles around X, Y, and Z, regarding the reference coordinate system, respectively. The mentioned procedure

is summarized in Fig. 13. Regarding Fig. 13, the object orientations are considered as input for the inverse kinematics algorithm. These orientations of the object are three angular positions around z, y, and x-axis. All conditions to enter into the inverse kinematic algorithm are shown in Table 4. In this Table, the 'star' sign is an arbitrary angle except zero, and the number (6) is defined in Eq. (6). Regarding Table 4, if θ_{dz} isn't satisfied by Eq. (6), the robot must be transferred to a new coordinate, that has the appropriate conditions to enter into the inverse kinematic algorithm. But, if θ_{dz} is fulfilled by all conditions presented in Table 4, the requirements for entering into the inverse kinematic algorithm, are satisfied.

$$\theta_{dz} = \pm(\alpha - \frac{\delta}{2}) \quad (6)$$

In Eq. (6), the sign "positive" is used when the gripper is downward, and the "negative" sign is used for the upward gripper direction. When the object is rotated around x or y-axis more than 90 Degrees, for the grasping process, the gripper must be placed under the object. According to Table 5, if the z-axis direction of the object is upward, the sign (+) is used in Eq. (6), and when the object is downward, the sign (-) is used.

If all conditions presented in sections 4.1.1, 4.1.2, and 4.1.3, are satisfied, the robot can perform inverse kinematic algorithm, in a one-step operation, which is described in the next section.

4.2. Inverse Kinematic Algorithm Considering Object Orientation In The Workspace

After considering all limitations, we assume that the object orientations are satisfied with all necessary conditions. The inverse kinematic algorithm inputs are the object center of mass, as $\begin{bmatrix} x_d & y_d & z_d \end{bmatrix}$, and the object orientations, as $\begin{bmatrix} \theta_{dx} & \theta_{dy} & \theta_{dz} \end{bmatrix}$, regarding the reference coordinate system. As shown in Fig.14, the gripper wrist has three joints No.5, 6, and 7. The proposed algorithms should be reliable and straightforward to compute the output joints angles very fast and accurately

In the inverse kinematic process, first, as shown in Figs. (15) and (16), the top view of the system on the X-Y plane, regarding the reference coordinate system, is obtained. In this step, the angle between projections of the robot from joint No.1 to No.6, regarding the X-axis, is obtained. It should be noted that the angle related to joint No.2 will be zero based on the reason which is described in section 4.1.1. There are no other rotations around the y-axis. Second, links projections of joint No.1 until No.6 are considered in the r-Z plane (principal links plane). In the following, only X-Y, and r-Z planes are enough to analyze the presented method.

Fig. 17 shows that robot joints No.5, 6 and 7 should be rotated such that the robot gripper becomes perpendicular to the object (which is defined in section 4.1.3). Therefore, regarding this figure and Fig. 18 the necessary angles to the right grasping process will be obtained. It should be noted that regarding Fig. 18, the lengths g_1 , g_2 , and g_3 , are object positions relative to joint No. 6, regarding the reference

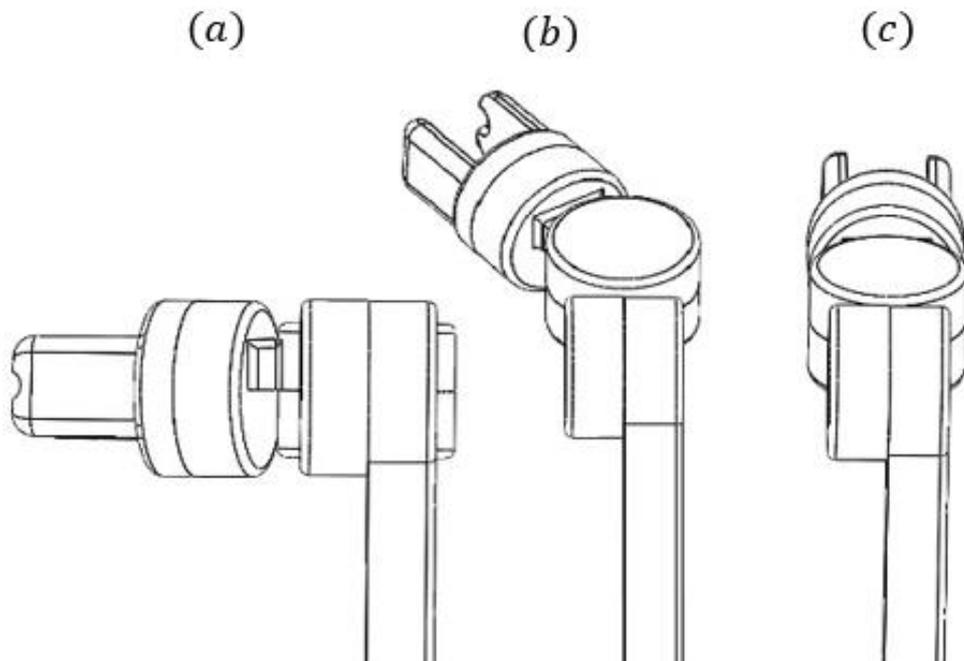


Fig. 12. Object frame projection and robot on the X-Y reference frame

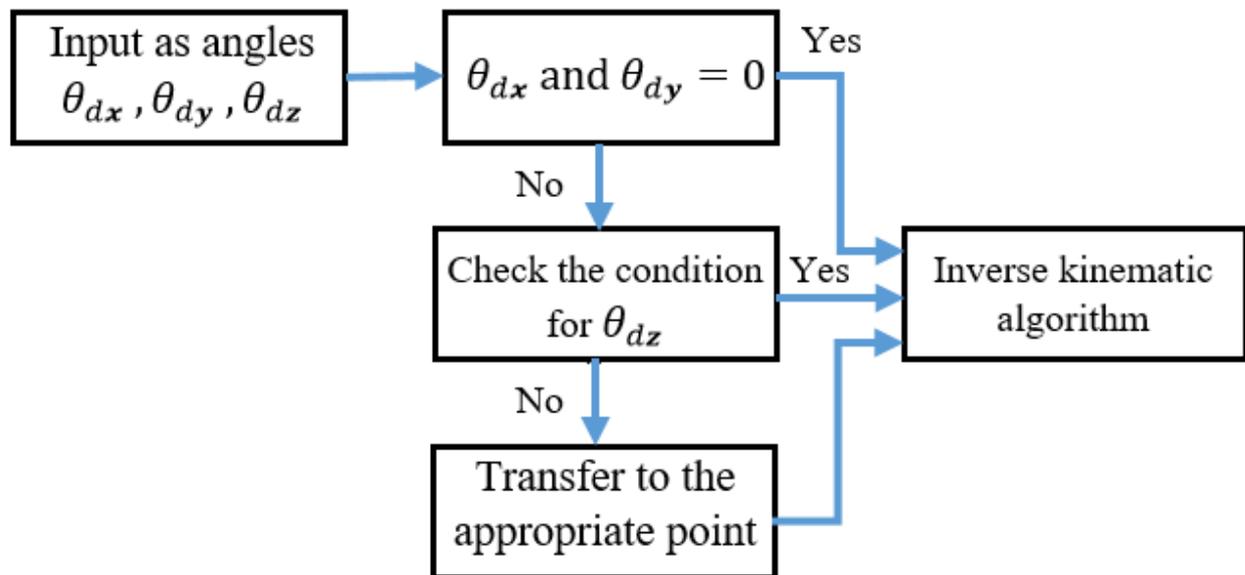


Fig. 13. The gripper orientation conditions to reach the desired target (object)

Table 4. The θ_{dz} angle orientation conditions to enter into the inverse kinematic algorithm

| θ_{dy} | θ_{dx} | θ_{dz} |
|---------------|---------------|---------------|
| 0 | 0 | * |
| 0 | * | (6) |
| * | 0 | (6) |
| * | * | (6) |
| π | π | * |

Table 5. The direction of the z-axis for both object and gripper considering the input angles.

| θ_{dx} | θ_{dy} | Gripper head direction | The object z axis direction |
|---------------|---------------|------------------------|-----------------------------|
| 0-90 | 0-90 | Downward | Upward |
| 0-90 | 90-180 | Upward | Downward |
| 90-180 | 0-90 | Upward | Downward |
| 90-180 | 90-180 | Downward | Upward |

coordinate system. Regarding Figs. 17-18 and considering object orientations as the desired directions of the gripper (the rotation of joints No.5, 6 and 7), g_1 , and g_2 are obtained using Eq. (7-8) in the following form.

$$g_1 = (L_1 + L_2) \cdot \sin(\theta_{dy}) \quad (7)$$

$$g_2 = (L_1 + L_2) \cdot \sin(\theta_{dx}) \cdot \cos(\theta_{dy}) \quad (8)$$

where L_1 and L_2 are the length of links 7 and 8, respectively.

Then, using Eqs. (7-8), and regarding Fig. 16, d_6 and l_2 can be obtained as follows:

$$d_6 = \tan^{-1}(g_1 / g_2) \quad (9)$$

$$l_2 = \sqrt{(g_1)^2 + (g_2)^2} \quad (10)$$

Now, using Fig. 15, other unknown geometrical characteristics are obtained by Eqs. (11-19) in the following form:

$$\alpha_0 = \pi - d_6 \quad (11)$$

$$\alpha_1 = \tan^{-1}(y_d / x_d) \quad (12)$$

$$l_1 = \sqrt{(x_d)^2 + (y_d)^2} \quad (13)$$

$$\alpha_2 = \sin^{-1}(\sin(\alpha_0) \cdot l_2 / l_1) \quad (14)$$

$$\alpha_3 = \alpha_1 - \alpha_2 \quad (15)$$

$$l_3 = (\sin(\alpha_4) \cdot l_1 / \sin(\alpha_0)) \quad (16)$$

$$l_3 = (\sin(\alpha_4) \cdot l_1 / \sin(\alpha_0)) \quad (17)$$

Considering Fig. 19, the joints angles No.1 and 6, are obtained regarding Eqs. (18) and (19), as follows:

$$\theta_1 = \alpha_3 - \pi / 2 \quad (18)$$

$$\theta_6 = -\theta_{dy} \quad (19)$$

Now the position of joint No.6 in the r-Z plane must be obtained. The position of this joint in the r-Z plane is presented in Eqs. (20) to (22). The projection vector of this joint, in the reference coordinate system, in r-direction, is denoted by r_6 , which can be obtained as:

$$\text{if: } \begin{cases} \theta_{dy} = 0 \rightarrow r_6 = l_1 - l_2 \\ \theta_{dy} \neq 0 \rightarrow r_6 = l_3 \end{cases} \quad (20)$$

Regarding Fig. 18, the lengths g_3 , i.e., the object position relative to joint No. 6 in the Z-direction, regarding the reference coordinate system, is obtained as:

$$g_3 = (L_1 + L_2) \cdot \cos(\theta_{dy}) \cdot \cos(\theta_{dx}) \quad (21)$$

So, considering Fig. 18, the height of joint No. 6 in the Z-direction, which shown by z_6 , is calculated in the following form.

$$z_6 = z_4 + g_3 \quad (22)$$

Now, by having joint No.6 coordinate in the r-Z plane, the coordinate of joint No.5 can be calculated. According to Fig. 20, the position of joint No.5 in the r-Z plane is obtained, as Eqs (23) to (24).

$$z_5 = z_6 + L_2 \cdot \cos(\theta_{dx}) \quad (23)$$

$$r_5 = r_6 - L_2 \cdot \sin(\theta_{dx}) \quad (24)$$

In this step, by having the position of joint No.5, we can obtain the position of joint No. 3 and No.4. It should be noted that joints No.3, 4, and 5, are located in r-Z plane, therefore, due to the position of joint No.5 and considering Figs. (21) to (22), via geometrical calculations, we can write Eqs. (25) to (33), for computing the parameters, which are illustrated in Fig. (22). To do this, first, regarding Fig. 21 the position of joint No.5, regarding joint No. 3, in the r-Z plane, are introduced in the following form.

$$z = z_5 - (L_1 + 2L_2), r = r_5 \quad (25)$$

Now, according to Fig. 22, by helping geometric calculation, we can write the following relation to obtain the unknown parameters in this figure.

$$R = \sqrt{r^2 + z^2} \quad (26)$$

$$\varnothing = \cos^{-1} \left(\left(R^2 - (L_3^2 + L_3^2) \right) / (2.L_3.L_3) \right) \quad (27)$$

$$\beta = \tan^{-1} (z / r) \quad (28)$$

$$\alpha = \sin^{-1} (\sin(\varnothing).L_3 / R) \quad (29)$$

Again, considering Fig. 22, the necessary relations to find the unknown variables in the inverse kinematic algorithm can be written as:

$$\theta_4 = \pi - \varnothing \quad (30)$$

$$\gamma = (\alpha + \beta) \quad (31)$$

$$|\theta_3| = (\pi / 2) - \gamma \quad (32)$$

$$\theta_5 = \theta_3 + \theta_4 + \theta_{dx} - \pi \quad (33)$$

In the process of inverse kinematic solving, for two planar link manipulators, two possible answers have been obtained, that Fig. 23 shows these two situations. In this condition, a solution is acceptable, which has less rotation relative to the previous operation. Obviously, according to Fig. 23, the situation “a” is selected for the inverse kinematic answer.

The final step of the inverse kinematic algorithm, regarding Fig. 24, which shows the projection of the robot in the X-Y reference coordinate system, is obtaining the necessary rotation of joint No.7. In the following subsection, because of previous steps of the inverse kinematic algorithm, the rotation of the gripper from its initial situation to the final position of the object must be obtained. This rotation will be achieved by joint No.7 and can be occurred in two separate states as follows:

State A: when the object has no rotation around its x or y-axis, the necessary rotation for joint 7 is obtained regarding the object rotation around the z-axis. According to Fig. 24, when the robot rotates by joint No. 1, the y-axis of the gripper isn't parallel to the object surfaces. Therefore, joint No. 7, in the gripper, must compensate for this rotation, too. So, regarding this rotation and the necessary desired rotation of the object around z-direction, the gripper should be rotated such that it can grasp the object in the right form. It should be noted that the object rotation up to 180 Degrees around the y or x-axis can be considered in this state because in these situations, the gripper is always perpendicular to the X-Y reference plane. Also, when the gripper direction is considered upward or downward, these conditions are different and are illustrated in Fig. 25 and Table 6. The gripper and object projections in the X-Y reference plane, are shown in Fig. 25. In this figure, Case “a” represents the upward gripper, (i.e., the object is rotated 180 Degrees around x as Fig. 25 “a₁” or y-axis as Fig. 25 “a₂”). Case “b” is shown for the downward gripper; in this case, “b₁” is for no rotation around x and y-axis. Whereas “b₂” illustrates the rotations around both axes, equal to 180 Degrees.

In this state, considering Fig. 24, Fig. 25 and Table 6, if the object rotation is considered, as θ_{dz} , the final value of rotation for joint No. 7, to the right grasping can be obtained by Eq. (34).

$$\theta_7 = \text{sign}(\theta_{dz}).\theta_1 - \theta_{dz} \quad (34)$$

State B: in this state, the object has the desired rotation around x or y-axis except for 0 or 180 Degrees. According to the definition in section 4.1.3, by substituting Eq. (6) into Eq. (34), considering Fig. 19, the value of θ_7 is obtained as

$$\theta_7 = 0 \quad (35)$$

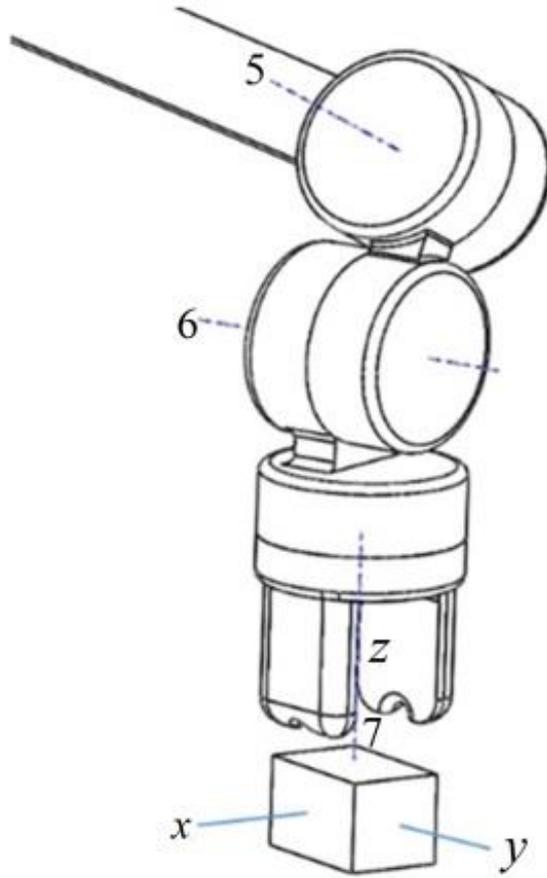


Fig. 14. Joints of gripper wrist to grasp the object

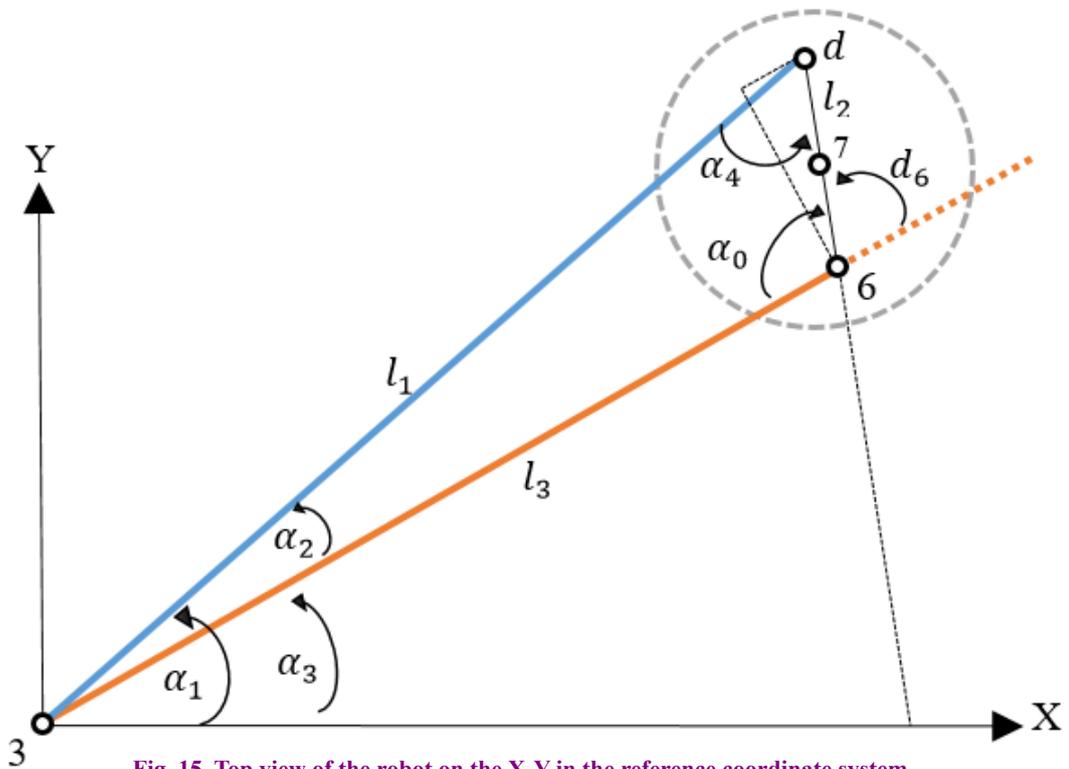


Fig. 15. Top view of the robot on the X-Y in the reference coordinate system

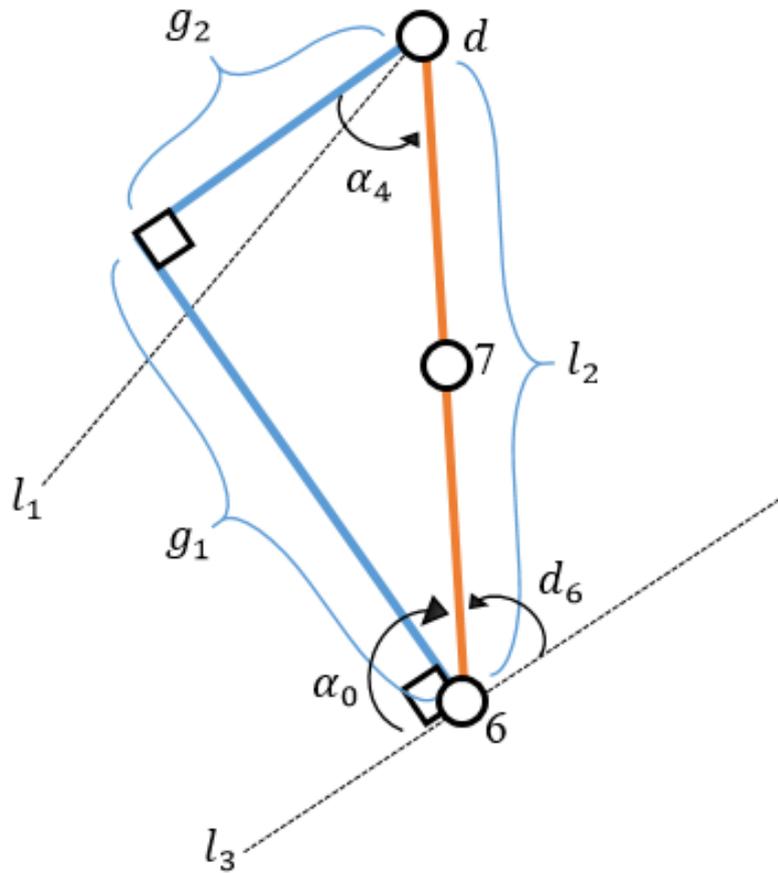


Fig. 16. A part of Fig.15 for more details

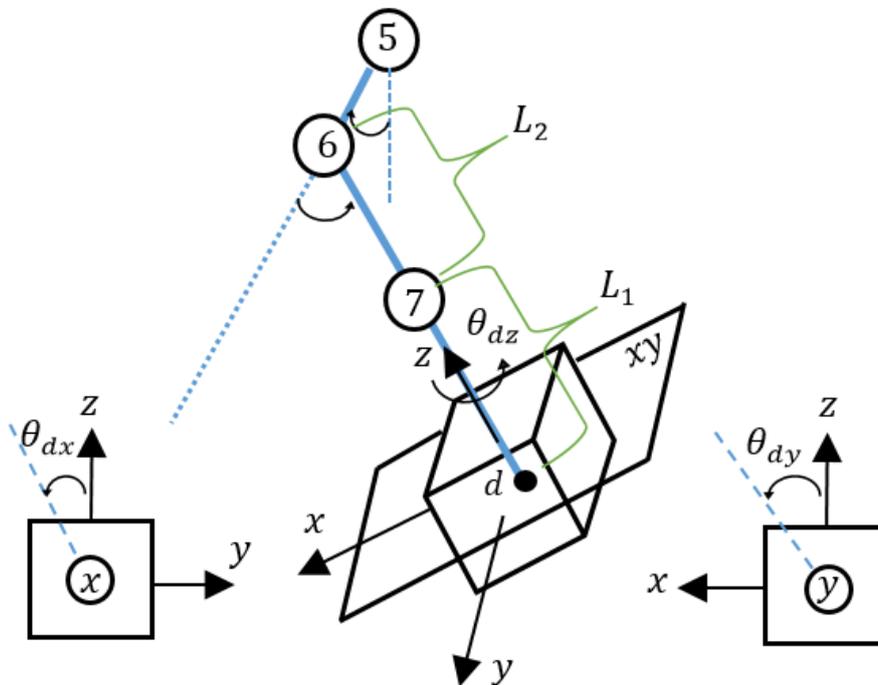


Fig. 17. Object orientations in different side views

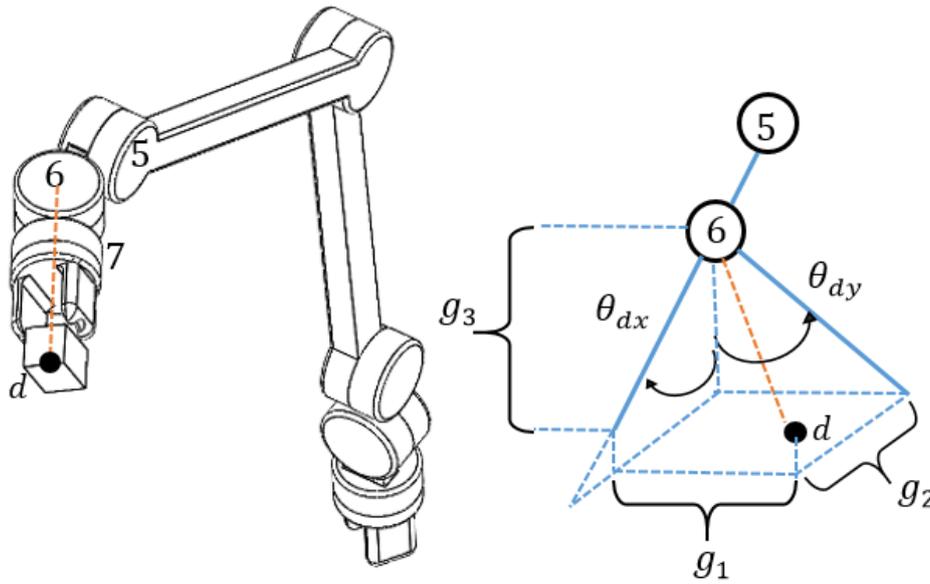


Fig. 18. Rotations of joints No.5 and 6 to grasp the object

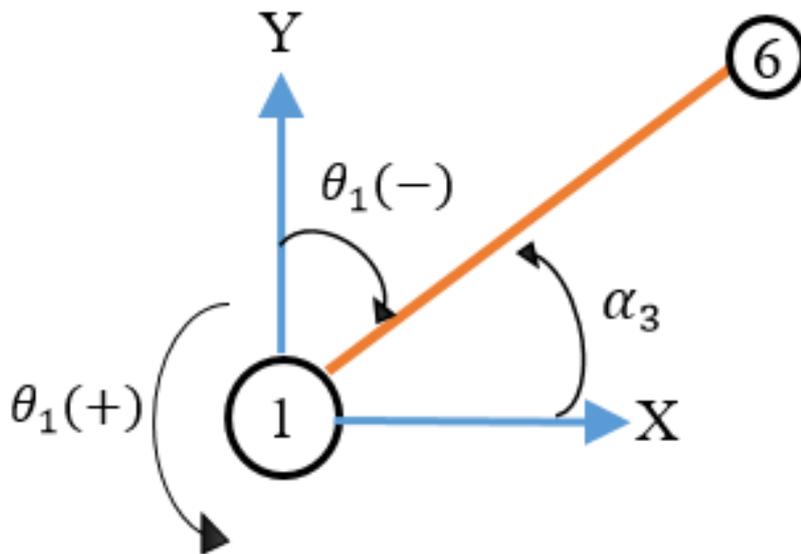


Fig. 19. The rotation of joint No.1 (Top view)

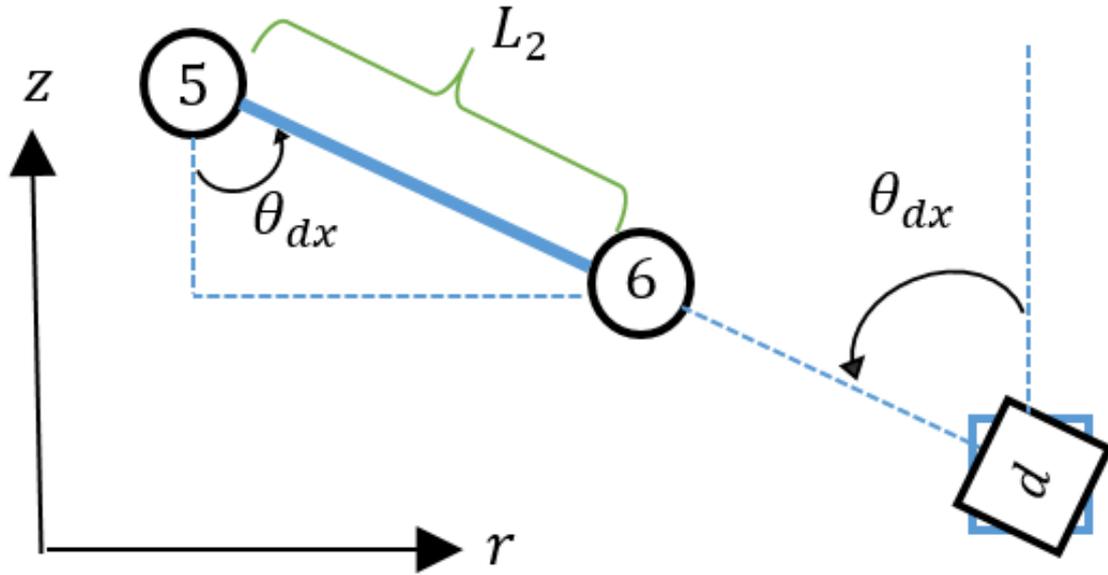


Fig. 20. Coordinate joint No.6 relative to joint No.5

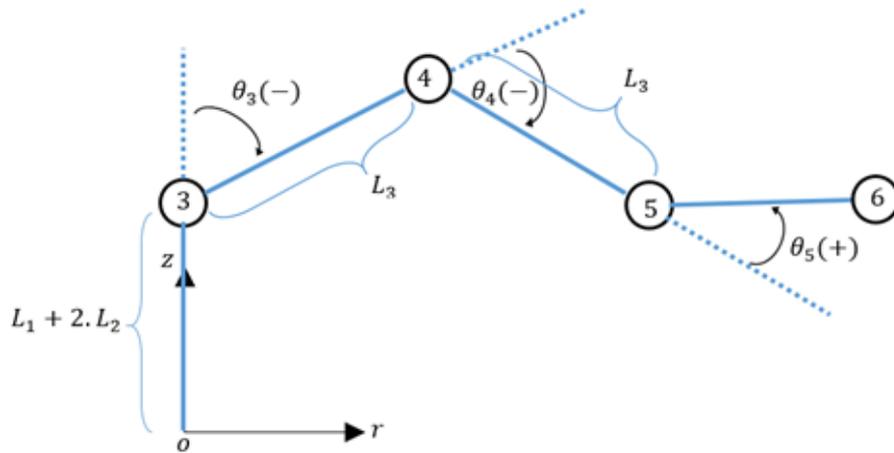


Fig. 21. Robot links in the r-Z plane. The positive sign for counterclockwise rotation is considered in the reference frame

The “one-step algorithm” has been finished here, and the summary results for the inverse kinematic of the ERA robot are listed in Table 7.

4.3. Transferring Operation

In the previous section, the one-step operation is presented so that one gripper of the robot is fixed, and another one is moved to grasp the target. But in this section, an algorithm is introduced which the robot performs a walking process to reach the final target point. In each step of transferring

operation, for the robot, from the current target point to the previous target point, there is exactly one step. Moreover, this point should be satisfied with all the necessary conditions in the inverse kinematic algorithm to grasp the object. Consequently, we must find a set of appropriate points to transfer the robot until the last step is realized correctly. The ERA robot has two grippers, which for each step of transferring operation, one gripper is fixed, and the other one is moved such that the gripper reaches a suitable position. It is hard to design a unit algorithm for various applications of

these types of robots. So, only two cases are considered in the two next subsections that will cover basic situations of transferring operations.

4.3.1. General Transferring In Space

Transferring operation of the robot is discussed in this section. Of course, this section is presented for the ERA robot, but the suggested algorithm can be utilized for similar robots like ERA. So, transferring operations can be extended to apply for many similar robots like ERA, and we can consider different limitations in the workspace and performing various operations. Generally, when a target isn't placed in the robot workspace, the robot must be moved to another place close to the target so that it can correctly grasp the object. Therefore, if the utilized robot and its limitations aren't satisfied in the one-step operation, the transferring operation is proposed. We can deploy a mission for a specific environment and operations by defining the necessary points and orientations. Afterward, the robot can be moved from the origin of the reference coordinate system, and it is transferred step by step to the next point considering new obtained points. Some application for this case is as follows: pick and place operation of objects in the space station (main application of ERA) or handling and inspection tasks through other equipment like cameras. In this case, an algorithm is needed, for transforming all points and orientations to the local coordinate, which is introduced in this section. Considering these three following subsections for transferring operation, we can cover many operations and tasks by robots like ERA robot.

First, it is assumed that all defined conditions for the inverse kinematic algorithm are fulfilled, and all positions and orientations of each target exist in the workspace. The robot can start from the reference coordinate system, and step by step, the robot is transferred to the next point considering the middle defined points until the robot reaches the final target positions, as shown in Fig. 26. To this end, because the inverse kinematic is described, for a one-step operation, regarding the reference coordinate, it is necessary to transform all the points and orientations to local coordinate using Eqs. (36-38) as follows:

$$r_{i-1}^i = r_0^i - (r_0^1 + r_1^2 + \dots + r_{i-2}^{i-1}) \quad (36)$$

$$R_{i-1}^i = R_0^i \cdot (R_0^{1T} \cdot R_1^{2T} \dots R_{i-2}^{i-1T}) \quad (37)$$

$$R_{i-1}^i = R_0^i \cdot (R_0^{1T} \cdot R_1^{2T} \dots R_{i-2}^{i-1T}) \quad (38)$$

where r_{i-1}^i is the vector of transferring coordinate from point " $i-1$ " to " i ", similarly R_{i-1}^i is a rotation matrix of coordinate " i " relative to " $i-1$ " until the zero coordinate is reached (the reference coordinate system).

4.3.2. Transferring To Achieve The Desired Orientation

Considering section 4.1.3, when the object orientation is not reachable in a one-step operation, we need to transfer the robot to a proper position. The problem is finding a new point such that the robot can grasp the main target regarding the

desired orientation. According to the limitations presented in section 4.1.3, the object projection on the X-Y reference coordinate plane must become tangent to the specific circle. According to Fig. 27, the center of this circle is joint No. 1, and the radius of this circle is the projection vector between joint No. 1 to joint No. 7 in the X-Y plane (hereafter, this circle is named "basis circle"). Therefore, the main topic, in this section, is finding a new center point, such that the new basis circle becomes tangent to object projection, in the X-Y plane.

Fig. 28 shows three different rotations of the object whose basis circle isn't tangent to the object surfaces. Regarding this figure, if the object turns around the x-axis, to perform the right grasping, the gripper must take a distance " g_2 " along the radius of basis circle by joint No. 5 (case c in Fig. 28). Furthermore, if the object turns around the y-axis, to perform the right grasping, the gripper must make distance " g_1 " along the tangent line on the basis circle by joint No. 6 (case a, in Fig. 28). If the object turns around both x and y axes, the gripper must take a distance " g_2 " and " g_1 ", along the radius of basis circle by joint No. 5 and the tangent line on the basis circle by joint No. 6, respectively (case b in Fig. 28).

Fig. 29 shows the necessary geometric to find the right center of the basis circle for the right grasping of the object. Based on Figs. 28-29, we can say that the " t_3 " is the appropriate target point to transfer the robot from " t_0 " in the origin of the reference coordinate system. In the point t_3 , the inverse kinematic algorithm can be done easily, and the necessary conditions to enter into the algorithm will be satisfied. To this end, considering Fig. 29, the equations of the lines, " L_x " and " L_y ", that are parallel to the x and y-axis of the object frame, respectively, can be obtained as follows.

$$L_y \rightarrow (y - y_d) = \tan\left(\frac{\pi}{2} + \theta_{dz}\right) \cdot (x - x_d) \quad (39)$$

$$L_x \rightarrow (y - y_d) = \tan(\theta_{dz}) \cdot (x - x_d) \quad (40)$$

Regarding Fig.29,

the line equation of " L_x^p ", which is parallel to " L_x " and passes from the point " t_0 ", is obtained in the following form:

$$L_x^p \rightarrow y = \tan(\theta_{dz}) \cdot x \quad (41)$$

The intersection of two lines, L_y and L_x^p , makes the point " t_1 ", therefore, solving two equations (39) and (41) in terms of " x " and " y " gives the following coordinate for the point " t_1 ".

$$x_{t_1} = \frac{y_d - \tan\left(\frac{\pi}{2} + \theta_{dz}\right) \cdot x_d}{\tan(\theta_{dz}) - \tan\left(\frac{\pi}{2} + \theta_{dz}\right)} \quad (42)$$

$$y_{t_1} = \tan(\theta_{dz}) \cdot x_{t_1}$$

Regarding Fig. 29, the coordinates for the point " t_2 " can be derived as follows:

$$\begin{aligned} y_{t_2} &= y_{t_1} + g_1 \cdot \sin(\theta_{dz}), x_{t_2} \\ &= x_{t_1} + g_1 \cdot \cos(\theta_{dz}) \end{aligned} \quad (43)$$

where “ g_1 ” has been obtained in section 4.2. Finally, according to Fig. 29, the coordinates for the point “ t_3 ” can be obtained in the following form.

$$\begin{aligned} y_{t_3} &= y_{t_2} - g_2 \cdot \cos(\theta_{dz}) \\ x_{t_3} &= x_{t_2} + g_2 \cdot \sin(\theta_{dz}) \end{aligned} \quad (44)$$

where “ g_2 ” has been obtained in section 4.2. By transferring the robot into the point t_3 , it can grasp the object without any problem. Briefly, if object orientation is not suitable for a one-step operation, the above procedure is utilized to find new point t_3 such that the robot can properly grasp the object.

4.3.3. Optimized Transmission

To develop the algorithm presented in the previous section, the inverse kinematic algorithm is extended for a situation, such that the object coordinate is not placed in the robot workspace, i.e., the conditions presented in section 4.1.2 and 4.1.3 are not satisfied. In this situation, the robot must be moved to the nearest point relative to the object. But, there is no unit algorithm, such that it can cover all problems, and calculate the nearest next point. It is assumed that the gripper can achieve all the space points, which will be calculated in the developed algorithm as the new points to perform transferring operation. Firstly, the calculation method for finding the point “ t_3 ”, defined in the previous section, will be obtained. It should be noted that, in this situation and a one-step operation, the point t_3 isn't in the robot workspace. It is necessary to determine some points step by step for reaching point t_3 , and according to section 4.3.2, for finding point t_3 , key points “ t_1 and t_2 ” are also required. After obtaining the coordinate t_1 by Eq. 42, we must move this point on the line “ L_y ”, regarding Fig. 30, such that it takes a suitable distance from the object to perform inverse kinematic properly. In this algorithm, after finding the point t_1 , the necessary conditions of inverse kinematic should be rechecked. Therefore, regarding Fig. 30, point t_1 is selected close to the object, in an appropriate position, which must be placed into the dotted circle. Regarding Fig. 30, the radius of this circle “ R_{circle} ” is defined based on the length of the robot links, and is suggested in the following form, which must become in the range of $[0, 2L_3]$, such that the robot can operate appropriately:

$$\begin{aligned} \Delta x &= x_d - x_{t_1}, \Delta y = y_d - y_{t_1} \\ R_{circle} &= \sqrt{\Delta x^2 + \Delta y^2} \end{aligned} \quad (45)$$

After finding the point “ t_1 ”, we can obtain the point t_3 from Eqs. (43) and (44). Now, according to Fig. 31, the first operation is the transmission in 3-dimension space between the point “ S_0 ” at the first position of the robot and the point t_3 , which represents an appropriate point to enter into the inverse kinematics algorithm. The robot should walk across the straight line (transferring vector), which connects S_0 to t_3 . The length of this line is obtained as follows:

$$r_{t_3} = \sqrt{x_{t_3}^2 + y_{t_3}^2 + z_d^2} \quad (46)$$

where, r_{t_3} is the length of vector S_0 to t_3 . Using the robot links limitations defined in section 4.1.2, the maximum transferring step on this straight line is assumed in the following form.

$$r_t = 2.L_3 \quad (47)$$

The minimum number of steps is calculated by dividing the total length of the transferring vector (Eq. 46) on the length of the two principal links (Eq. 47) as follows:

$$i = \frac{r_{t_3}}{r_t} \quad i \in N \quad (48)$$

where i indicates the maximum number of transferring steps, which is rounded to a smaller integer number, and regarding Fig.31, for reaching to the point “ S_i ”, the “ i ” steps are needed. The last step of transferring, to reach into the point t_3 , is defined as:

$$r_{t(i+1)} = r_{t_3} - i \cdot r_t \quad (49)$$

Regarding Fig. 31, we can define the angle between the z-axis and S_0t_3 , i.e., \varnothing , and the angle between the projection of the robot, in the X-Y plane and X axis, i.e., θ , which shows the directions of the transferring vector, as follows:

$$\begin{aligned} \theta &= \tan^{-1}(y_{t_3} / x_{t_3}) \\ \varnothing &= \tan^{-1}(r_{t_3} / z_d) \end{aligned} \quad (50)$$

Regarding Eq. (50), the position vector for each point S_i , is calculated in the following form.

$$s_i : \text{for } i = 1 \quad \begin{cases} x_i = r_t \cdot \cos(\theta) \cdot \sin(\varnothing) \\ y_i = r_t \cdot \sin(\theta) \cdot \sin(\varnothing) \\ z_i = r_t \cdot \cos(\varnothing) \end{cases} \quad (51)$$

5. SIMULATION RESULTS AND VALIDATION

At first, to validate the suggested inverse kinematic algorithm, two procedures are performed. In the first validation method, the suggested formulation, for inverse kinematic algorithm is validated by the forward kinematics relations, through a written code in the MATLAB Simulink. In the second validation method, we made a dynamic model, for the ERA, in the MATLAB Simscape, and check the errors between the suggested inverse kinematic algorithm and the data recorded by the sensors of the Simscape model.

5.1. First Validation Procedure

So far, the inverse kinematic of the ERA robot isn't

presented. Therefore, there isn't any similar sample, such that can compare the suggested algorithm with those presented in the literature. Thus, in this subsection, we present a procedure to check the validation of the proposed algorithm, which is performed by helping the MATLAB Simulink. In this procedure, at first, object positions and orientations, as the input data are given into the proposed inverse kinematic algorithm. Outputs of this step are the angular positions of the robot as the joints space variables. In the second step, we use the forward kinematic relations, which are presented in Appendix. To this end, the angular positions obtained in the first step, as the outputs of the inverse kinematic algorithm, are inserted into the forward kinematic relations. The result of this step is the positions and orientations of the end-effector as workspace variables. To investigate the validation, for the suggested approach, these workspace variables must be equal to the object positions and orientations in the input of the first step. Therefore, in this procedure, the inverse kinematic is validated by the forward kinematics. Briefly, the method of validation is illustrated in Fig. 32. The error between this process (inverse-forward kinematic) and the given desired trajectory is investigated through the block diagram, as shown in Fig. 32.

For the first validation, three planes are designed to validate and simulate the suggested inverse kinematics algorithm. To this end, the desired path (position and orientation of the object) is selected, as the input for the inverse kinematics algorithm. First, the object orientation is checked to satisfy the necessary conditions of the inverse kinematics algorithm based on the gripper orientation. Then, a transferring operation is programmed for an object that isn't placed in the robot workspace. Links length of the robot, in the simulations, is defined in Table 8.

For the first simulation, the desired path is selected by Eq. (52), as shown in Fig. 33. The error between the desired path and inverse-forward kinematic results is shown in Fig. 34.

$$\begin{cases} x = 100 \cos(t) \\ y = 100 \sin(t) \\ z = 10 * t \end{cases} \quad (52)$$

According to Fig. 34, the path error is less than 8e-14 (mm); therefore the inverse-forward kinematic test verifies that the presented inverse kinematics algorithm works correctly.

For the second simulation, we assume that the object orientation changes, and therefore, we check the gripper orientation performance for right grasping. To ensure that the gripper can grasp the object, the rotation matrix of the object and gripper must be the same. However, according to the definition presented in section 4.1.3, there are some necessary conditions to grasp the object, i.e., when the gripper grasps the object, the z-axis direction of the gripper and object must be opposite. According to Table 1, the rotation matrix of the end-effector is obtained as follows.

$$R_{Robot} = R_1^T \cdot R_2^T \cdot R_3^T \cdot R_4^T \cdot R_5^T \cdot R_6^T \cdot R_7^T \quad (53)$$

Also, the following equation is derived for the object orientation.

$$R_{aim} = R_{z,\theta_{dz}}^T \cdot R_{x,\theta_{dx}}^T \cdot R_{y,\theta_{dy}}^T \cdot R_{x,\pi}^T \quad (54)$$

Now, the error of the rotation matrix can be calculated in the following form:

$$\text{error} = R_{aim} - R_{Robot} \quad (55)$$

In this step of the simulation, the object positions and orientations are presented in Table. 9. First, the necessary conditions to enter into the inverse kinematic algorithm are checked. After passing the required conditions, these coordinates are inserted into the inverse kinematic algorithm, and the results are shown in Fig. 35.

Fig. 35 shows the rotation error for the input, according to Eq. (55). Regarding this figure, the results verify the presented inverse kinematics algorithm.

For the third simulation, the transferring operation is considered. Table. 10 presents the input coordinate for transferring operation. These coordinates are not in the workspace, because these points don't satisfy the entering conditions for the inverse kinematics algorithm. Therefore, the transferring operation is necessary to grasp the object.

For this case, based on the method presented in sections 4.3.1-4.3.3, and according to Eqs. (39-51), all the middle points from the basis center to point t_3 are obtained, as shown in Table 11. Then, by Eq. (36-38), all these coordinate changes to the local coordinate. In the final step, the inverse kinematic algorithm is performed to grasp the object at the point t_3 . In this procedure, first, the algorithm proposed in section 4.3.3 is used to find t_1 , close to the projection of the object in the X-Y reference plane. Then, utilizing Eqa. (43) and (44), t_2 and t_3 are found. From the origin to the point t_3 , the eight necessary steps are obtained, and the ninth step is from S_8 to t_3 . Finally, from point t_3 to object, just one step is needed to complete the operation of grasping. The orientation in the last step is [40 30 60] (Degree). So, these ten steps are needed to reach and grasp the object. The results of the last step are shown in Fig. 36.

$$r_{t_3}^{\text{target}} = r_0^{\text{target}} - r_0^{t_3} \quad (56)$$

$$R_{t_3}^{\text{target}} = R_0^{\text{target}} \cdot (R_0^{t_3T}) \quad (57)$$

According to Fig.36, the robot reaches the desired object with negligible error. Therefore, the transferring operation works correctly.

5.2. Second Validation Procedure

For the second validation, for the suggested kinematic algorithm, we made a dynamic model in the MATLAB Simscape. This model is entirely equal to the ERA, regarding the geometrical characteristics and structures. In this model, we create enough sensors for the joint-space variables (angles of links) and workspace variables (positions and orientations of the gripper). Now, we run the dynamic model and record the data of these sensors. The data recorded in

the gripper sensor is the direct kinematic data (workspace variables), and the data recorded in the joints angles are the inverse kinematic data (joint-space variables). These data are the right information between the workspace and joint-space variables of ERA, which can be used to validate the suggested inverse kinematic algorithm. To this end, the data recorded in the gripper sensor is inserted into the suggested formulations of the inverse kinematic algorithm. Then the

inverse kinematic gives the angles of the links. This process is shown as a flowchart in Fig. 37. To validate the suggested inverse kinematic algorithm, the output of this algorithm should be equal to the data recorded by joints angles sensors. The simulation results are illustrated in Figs. 38-39. As can be seen from these figures, there are minor errors between the inverse kinematic algorithm, and the data recorded by sensors. Therefore, the inverse kinematic algorithm works correctly.

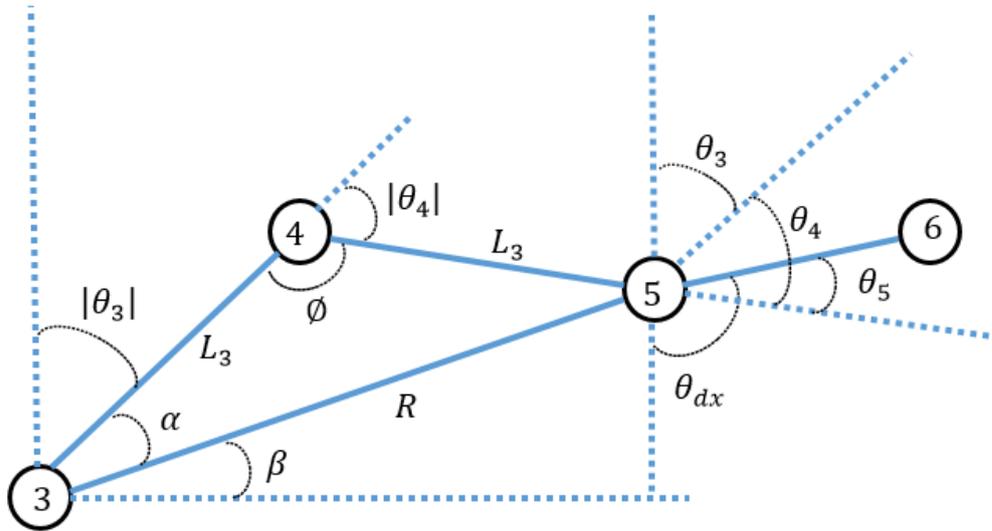


Fig. 22. Definition of the necessary parameters to find the angular position of joint No.5

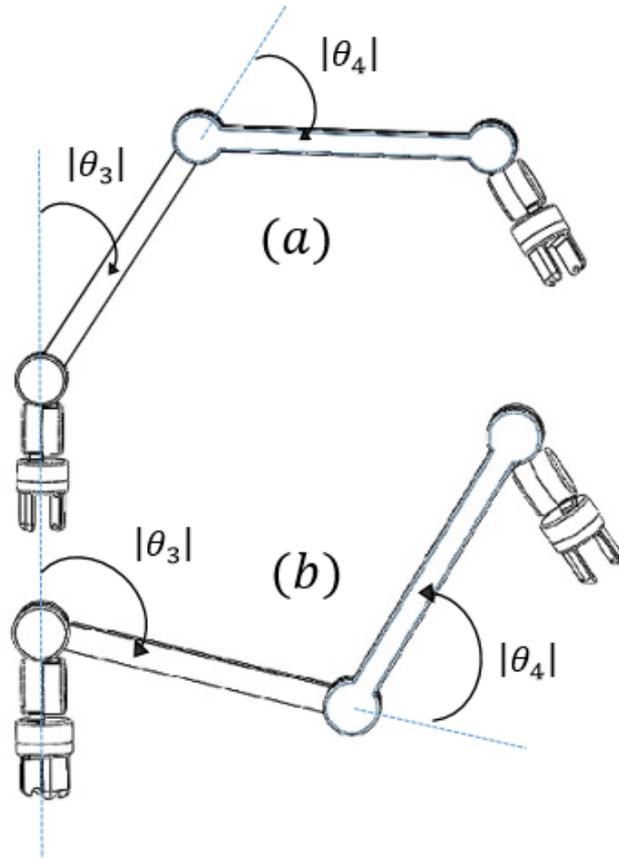


Fig. 23. Two possible solutions for the inverse kinematic of two-link planner robot

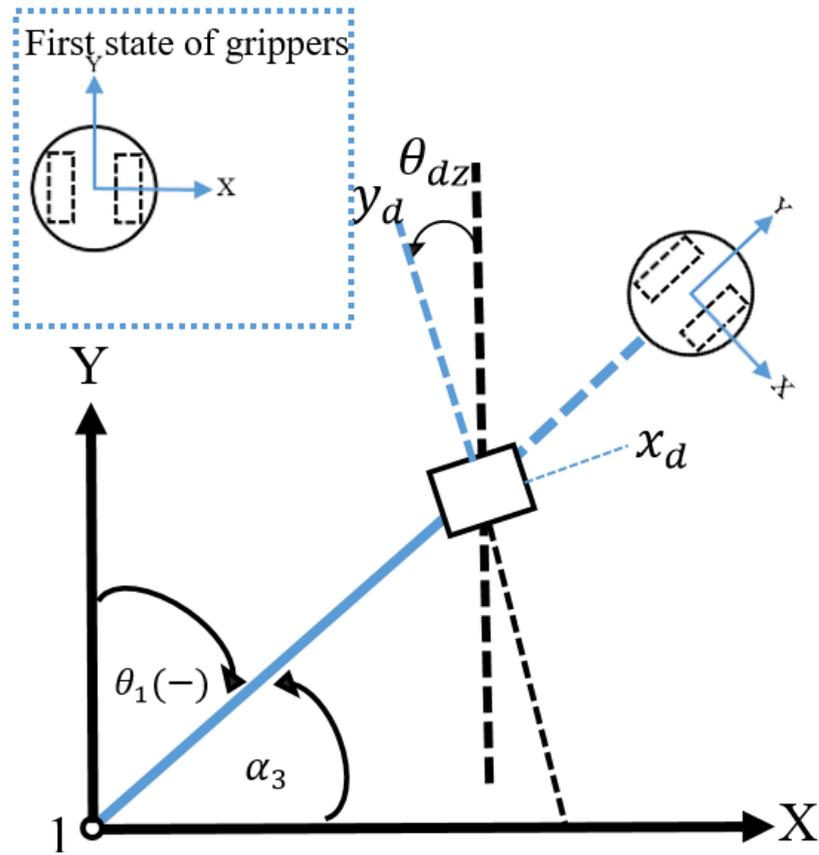


Fig. 24. The first and second situation of the gripper after rotation of robot by joint No.1

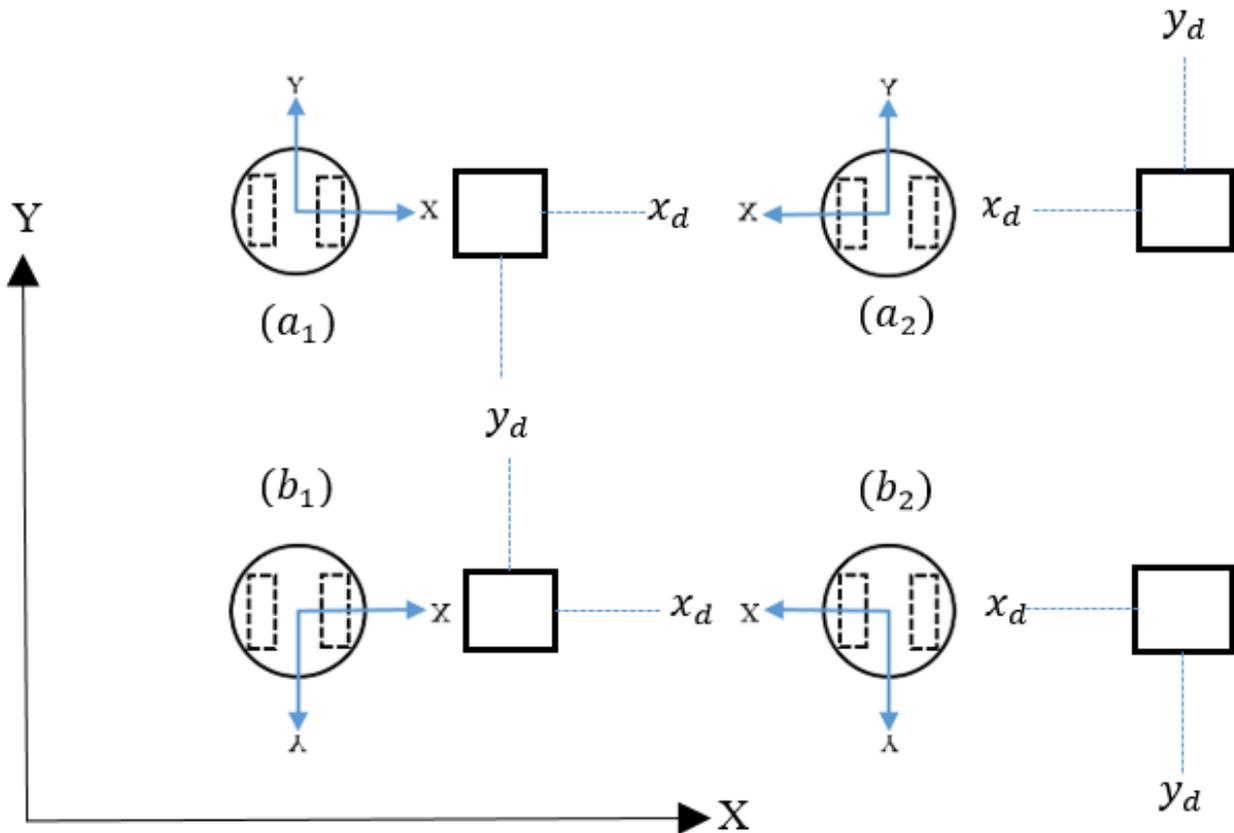


Fig. 25. State "A" conditions and their projections

Table 6 .Angular positions, according to Fig.25 (State “A”)

| From Fig.25 | θ_{dy} | θ_{dx} | $\text{sign}(\theta_{dz})$ |
|-------------------|---------------|---------------|----------------------------|
| (b ₁) | 0 | 0 | + |
| (a ₁) | 0 | π | - |
| (a ₂) | Π | 0 | - |
| (b ₂) | Π | π | + |

Table 7. Summary results for inverse kinematic of the ERA robot

| | |
|------------|---|
| θ_1 | $\alpha_3 - \pi / 2$ |
| θ_2 | 0 |
| θ_3 | $(\alpha + \beta) - (\pi / 2)$ |
| θ_4 | $-\cos^{-1}\left(\left(\frac{R^2 - (L_3^2 + L_3^2)}{2.L_3.L_3}\right)\right)$ |
| θ_5 | $-\theta_3 - \theta_4 + \theta_{dx} - \pi$ |
| θ_6 | $-\theta_{dy}$ |
| θ_7 | $\text{sign}(\theta_{dz}).\theta_1 - \theta_{dz}$ |

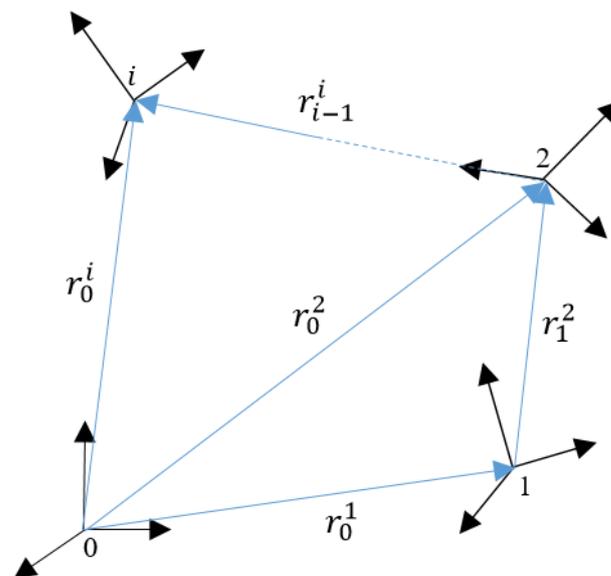


Fig. 26. Vectors and coordinate systems in the transferring operation

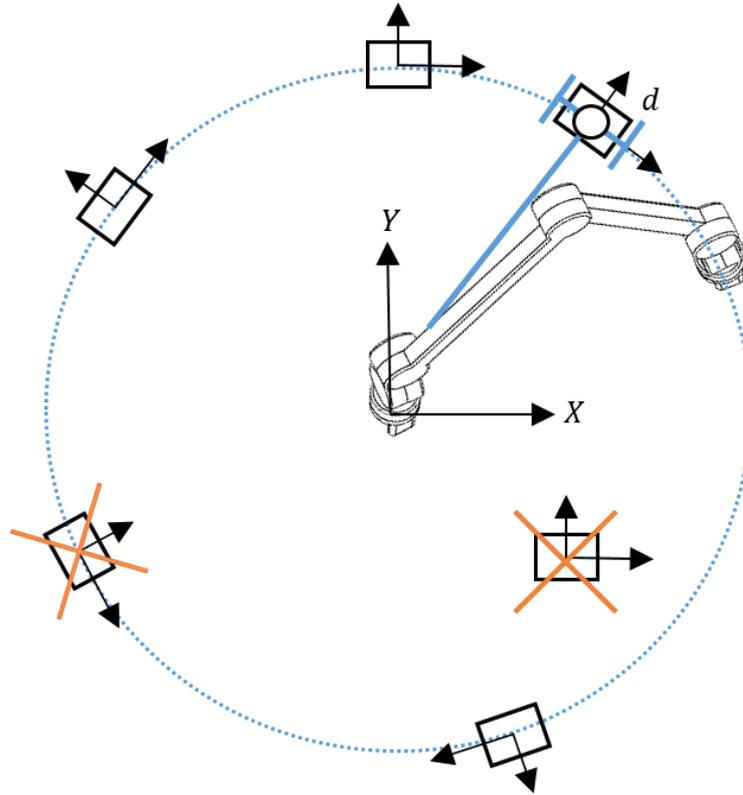


Fig. 27. Projections of the object rotated around x or y or both axis

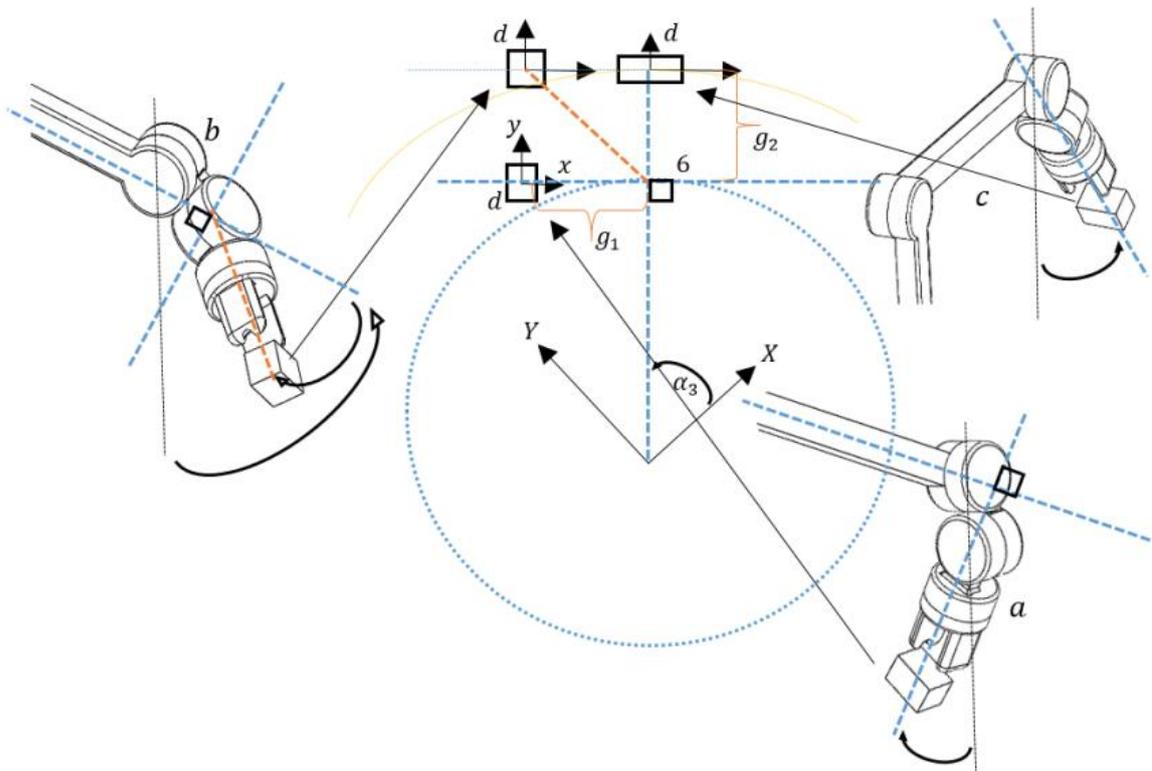


Fig. 28. Three states are presented for the robot to grasp the object, regarding Fig.11

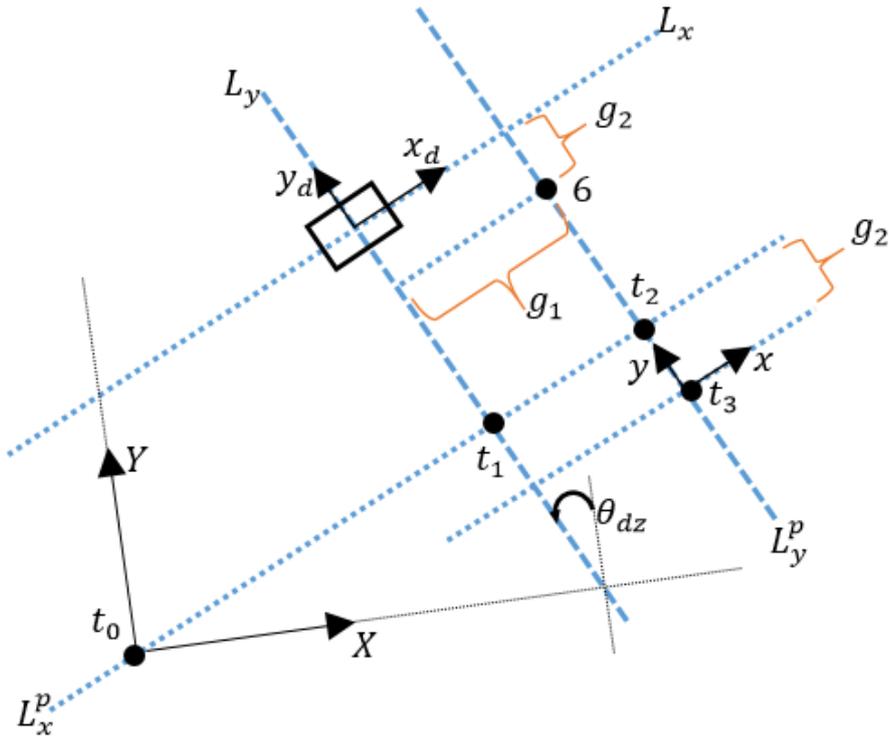


Fig. 29. Lines and their intersections in the reference plane which is a key point for the transferring operation

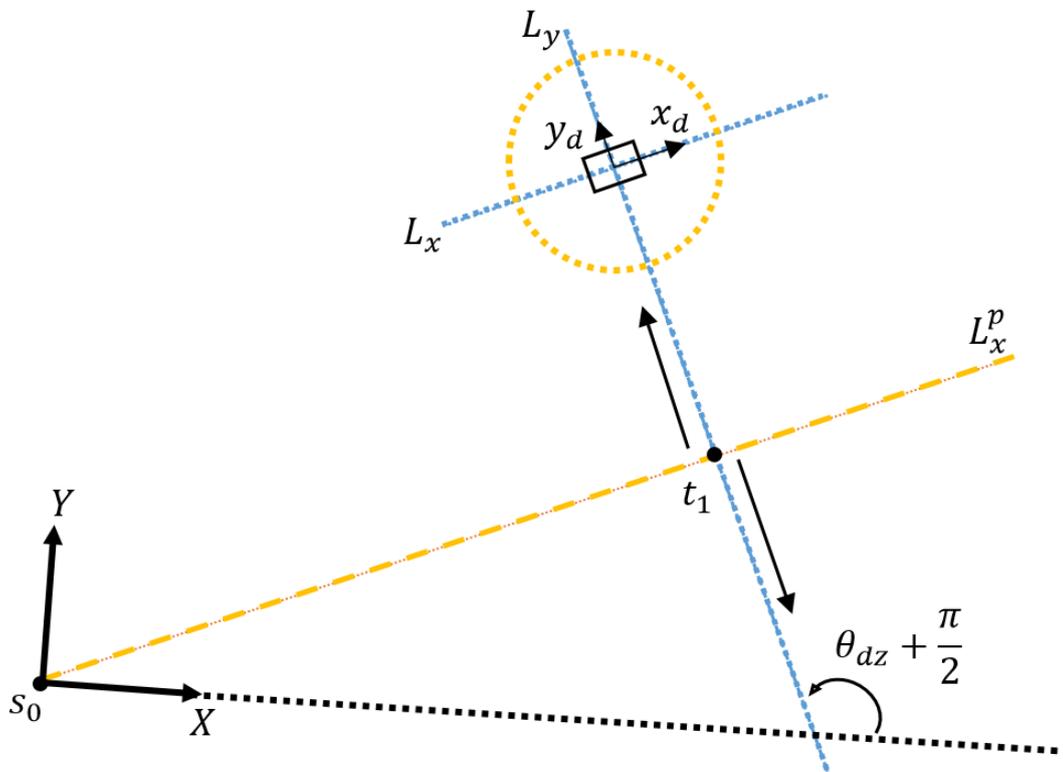


Fig. 30. The general view of the transferring operation to find the appropriate point for joint No. 1

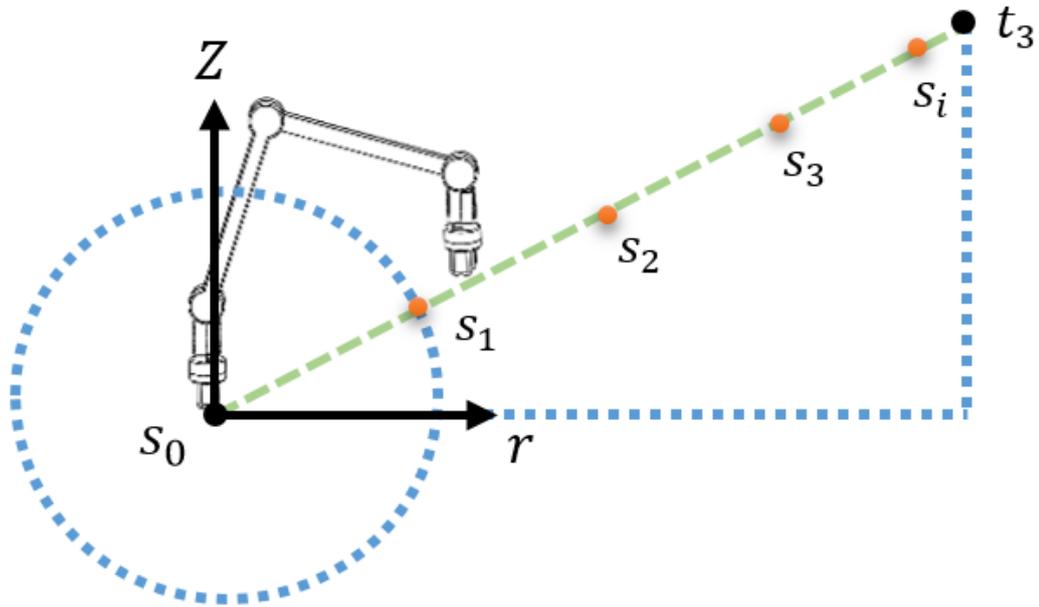


Fig. 31. Transferring operation by $i+1$ steps towards t_3

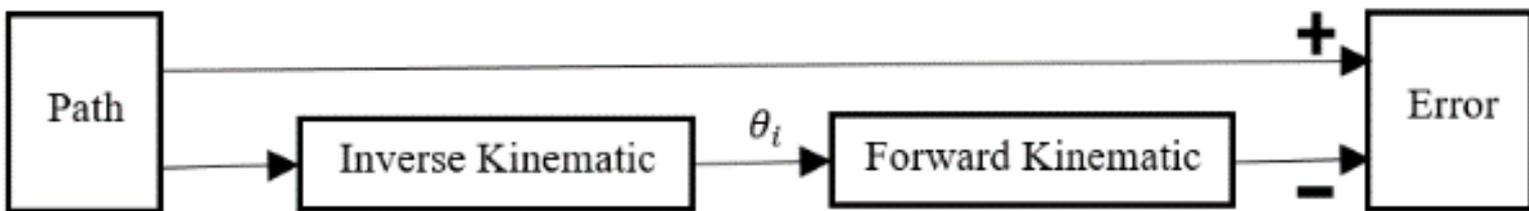


Fig. 32. Method of simulation and validation

Table 8. Robot links length

| parameter | L_1 | L_2 | L_3 |
|-----------|-------|-------|-------|
| (mm) | 21 | 21 | 120 |

Table 9. Object positions and orientations as the input of inverse kinematic algorithm

| coordinate | x | y | z | θ_{dx} | θ_{dy} | θ_{dz} |
|------------|-----|-----|-----|---------------|---------------|---------------|
| (mm) | -10 | 150 | 120 | 40 | 30 | 60 |

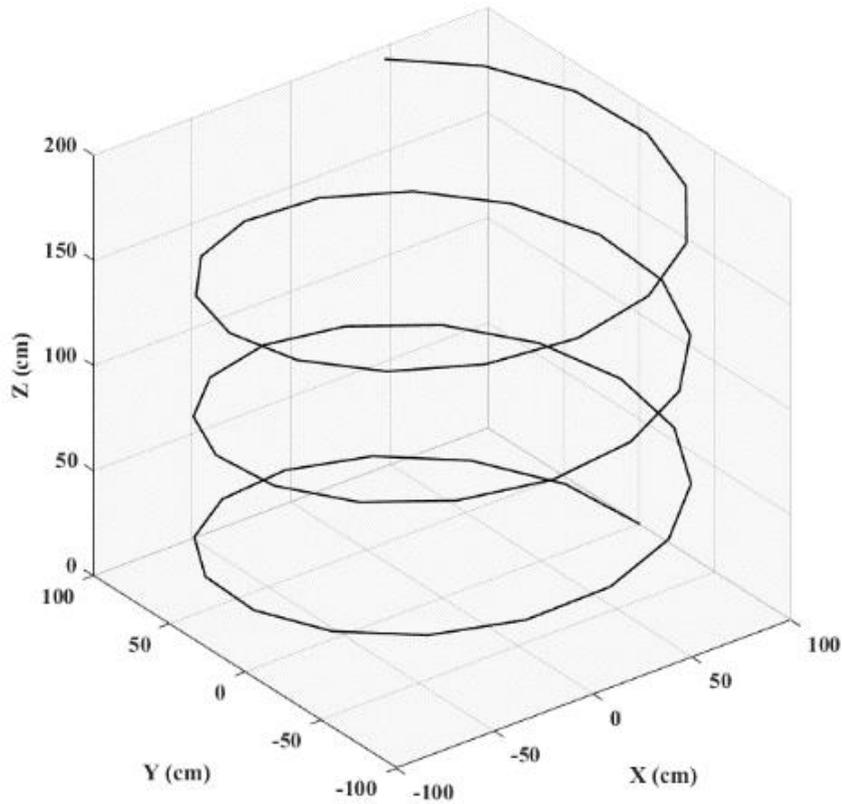


Fig. 34. Path error between the input and output simulation

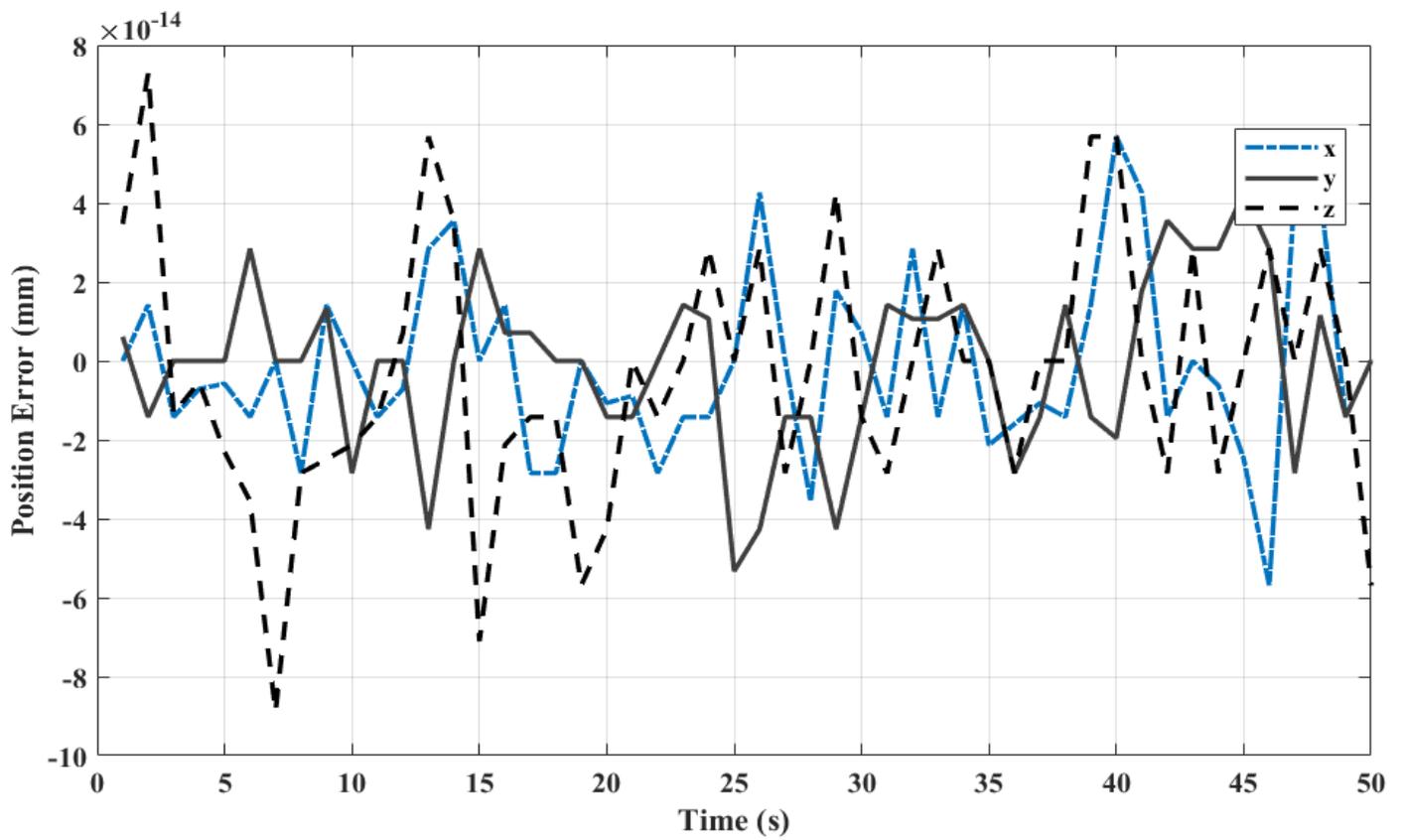


Fig. 34. Path error between the input and output simulation

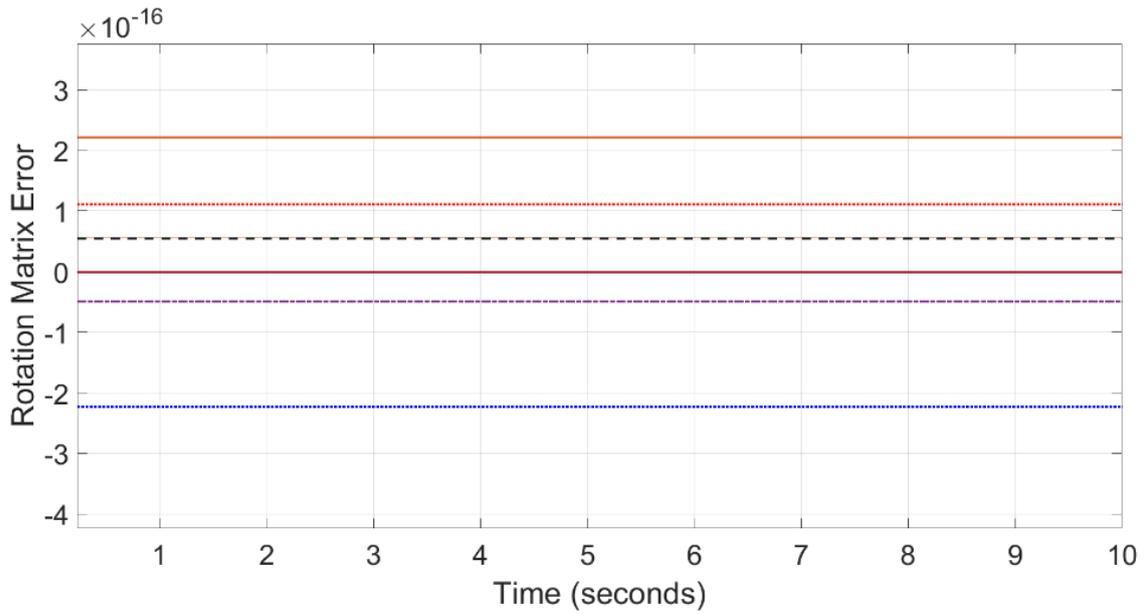


Fig. 35. Orientation error obtained regarding Eqs. (53-5)

Table 10. Object positions and orientations as the input of the inverse kinematic algorithm to perform the transferring operation

| coordinate | x | y | z | θ_{dx} | θ_{dy} | θ_{dz} |
|------------|-------|------|------|---------------|---------------|---------------|
| (mm) | -1000 | 1500 | 1200 | 40 | 30 | 60 |

Table 11. The coordinate of the necessary points for the transferring process

| points | X (mm) | Y (mm) | Z (mm) |
|--------|--------|--------|--------|
| s_1 | -1001 | 1685 | 1385 |
| s_2 | -2002 | 3371 | 2770 |
| s_3 | -3003 | 5056 | 4154 |
| s_4 | -4003 | 6742 | 5539 |
| s_5 | -5004 | 8427 | 6924 |
| s_6 | -6005 | 10113 | 8309 |
| s_7 | -7006 | 11798 | 9693 |
| s_8 | -8007 | 13483 | 11078 |
| t_1 | -8984 | 14414 | 0 |
| t_2 | -8823 | 14692 | 0 |
| t_3 | -8673 | 14605 | 12000 |

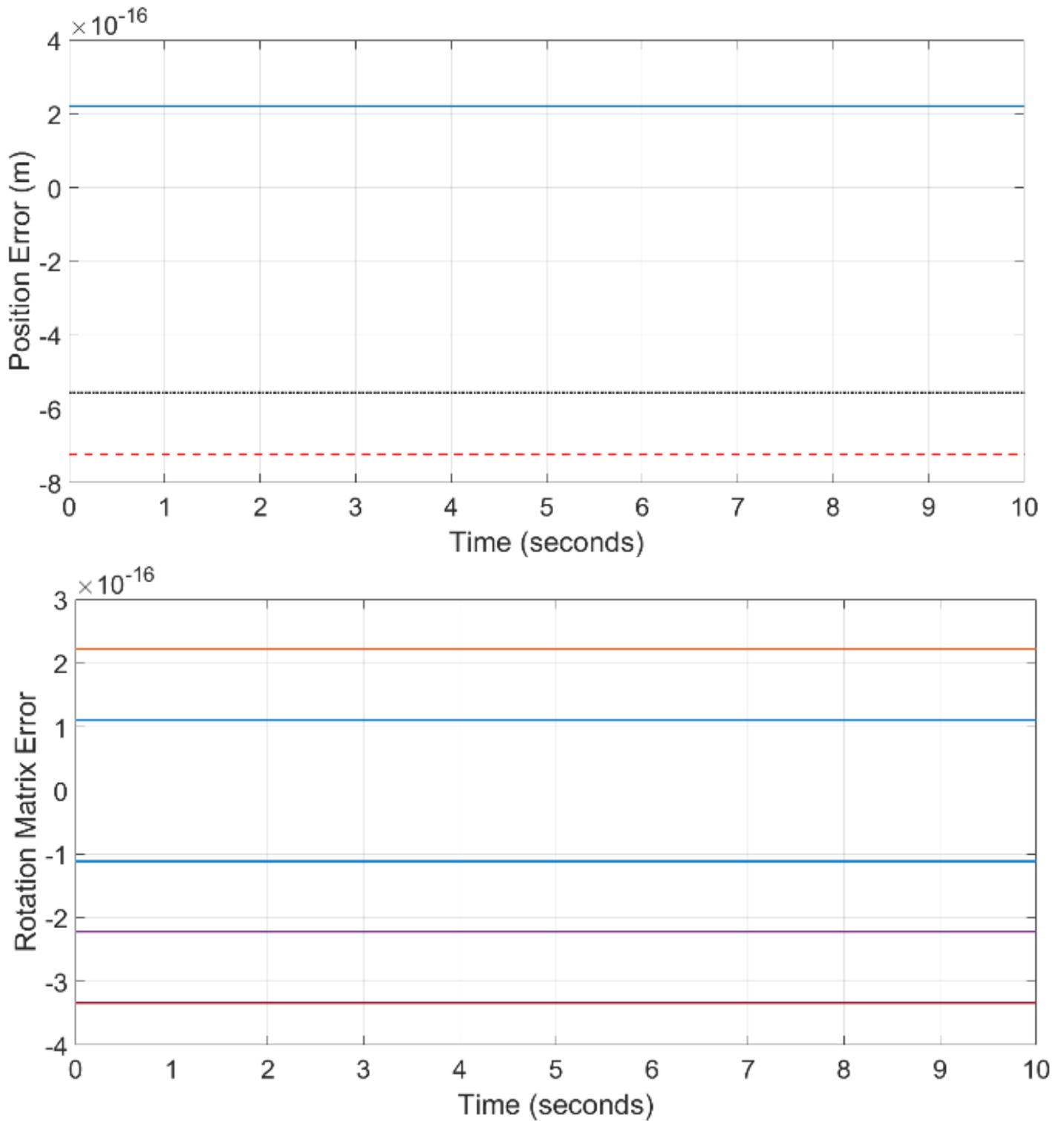


Fig. 36. Position and orientation error for the last step of transferring operation are performed by the inverse kinematic algorithms

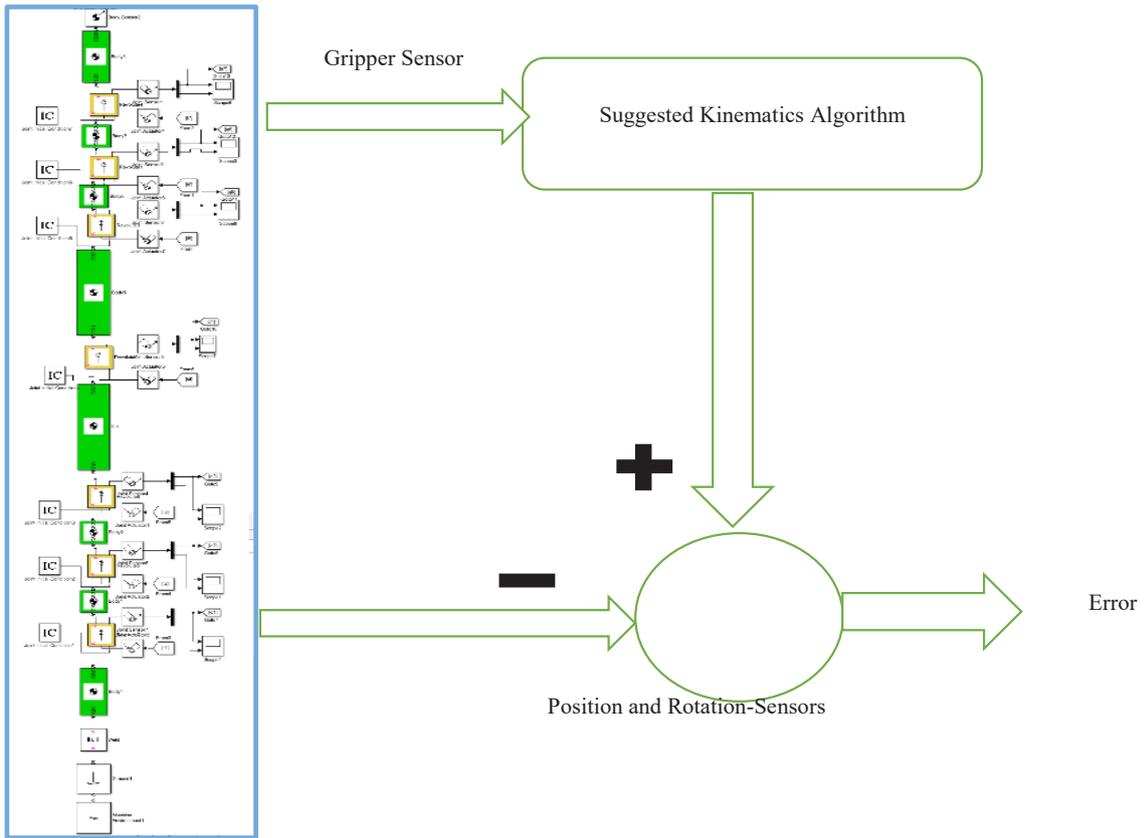


Fig. 37. Flowchart simulated in the MATLAB Simscape to the validation of inverse kinematic algorithm

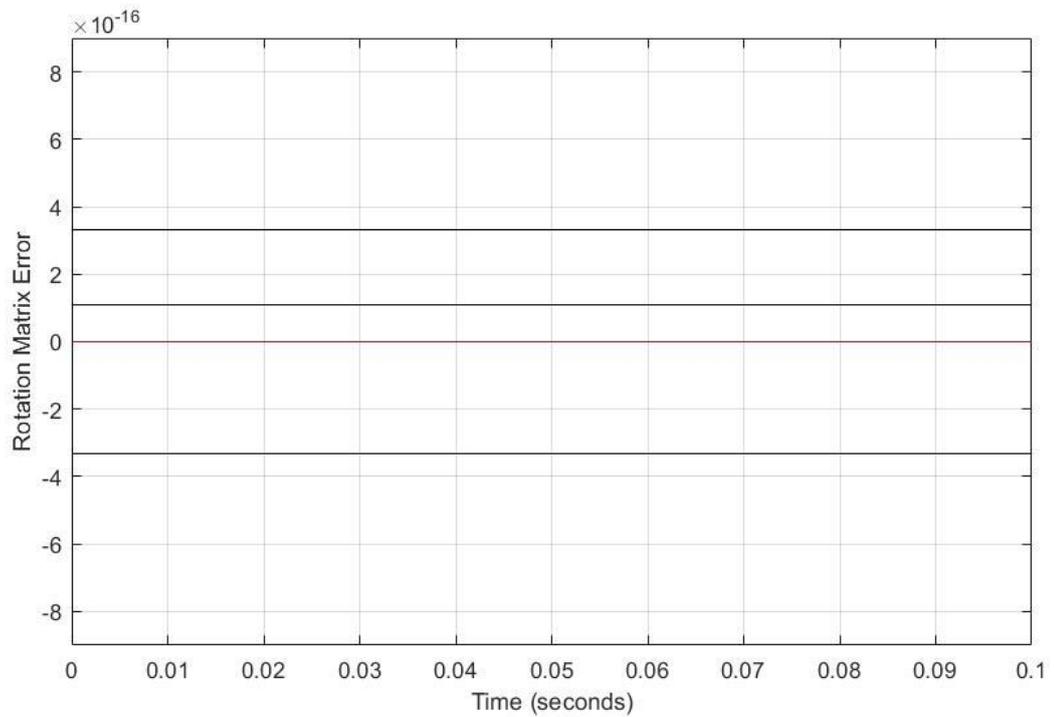


Fig. 38. The rotation matrix error between the inverse kinematic algorithm and the data recorded by sensors

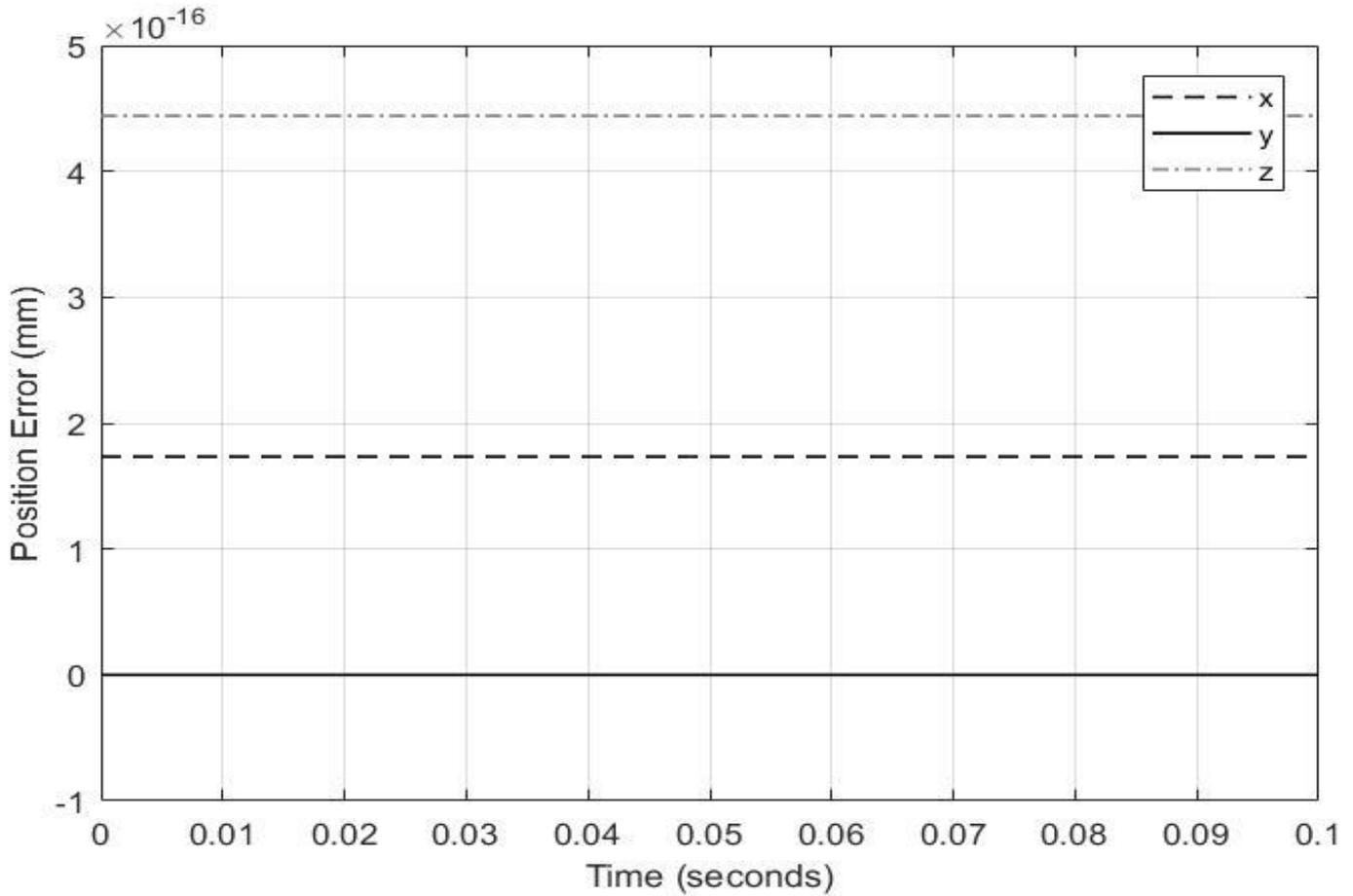


Fig. 39. The position error between the inverse kinematic algorithm and the data recorded by

6. CONCLUSIONS

In this study, a geometrical method for inverse kinematic analysis of a 7-degree freedom manipulator robot is proposed. To this end, the workspace limitations are analyzed. The model of gripper allows us to consider the object orientation. The presented approach makes the robot autonomously follows the desired points and paths. The suggested algorithm for inverse kinematic has two main steps as follows. Step (I): the object is located in the robot workspace, and all the necessary condition of the inverse kinematic algorithm is satisfied. In this case, the object can be located in any direction around the z-axis or 180 Degrees around the x or y-axis. Step (II): there are two cases, which the robot must perform the transferring operation, as follow: (a) the object places in the robot workspace, but the object has rotated around the x or y-axis

(except for 180 Degrees), and (b) the object doesn't place in the robot workspace.

Three planes are designed to verify and validate the inverse kinematics algorithm. The results of the simulations confirm the algorithm and show that the utilized robot tracks the desired trajectories with high accuracy.

Appendix

The forward kinematics of the system defined in section 3 is as follows:

$$r_{\text{target}} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}$$

$$\begin{aligned}
 r_1 = & C_1 S_2 L_2 + (C_1 C_3 S_2 + S_1 S_3) L_3 + \left(\begin{array}{l} -(C_1 S_2 S_3 - C_3 S_1) S_4 \\ + (C_1 C_3 S_2 + S_1 S_3) C_4 \end{array} \right) L_3 \\
 & + \left((C_1 S_2 S_3 - C_3 S_1) C_4 + (C_1 C_3 S_2 + S_1 S_3) S_4 \right) S_5 + \\
 & (-C_1 S_2 S_3 - C_3 S_1) S_4 + (C_1 C_3 S_2 + S_1 S_3) C_4 C_5 L_2 \\
 & + C_1 C_2 S_6 L_2 + \left((C_1 S_2 S_3 - C_3 S_1) C_4 + (C_1 C_3 S_2 + S_1 S_3) S_4 \right) S_5 \\
 & + \left(-(C_1 S_2 S_3 - C_3 S_1) S_4 + (C_1 C_3 S_2 + S_1 S_3) C_4 \right) C_5 C_6 L_3 \\
 & + C_1 C_2 S_6 L_1 + \left(\begin{array}{l} \left(\begin{array}{l} (C_1 S_2 S_3) C_4 + (C_1 C_3 S_2 + S_1 S_3) S_4 \\ - (C_3 S_1) S_4 \end{array} \right) S_5 + \\ \left(\begin{array}{l} -(C_1 S_2 S_3) S_4 + (C_1 C_3 S_2) C_4 \\ - (C_3 S_1) S_4 \end{array} \right) C_5 \end{array} \right) \\
 C_6 L_1 \\
 r_2 = & S_1 S_2 L_2 + \left(\begin{array}{l} C_3 S_1 S_2 \\ - C_1 S_3 \end{array} \right) L_3 + \left(\begin{array}{l} -(S_1 S_2 S_3 + C_1 C_3) S_4 \\ + (C_3 S_1 S_2 - C_1 S_3) C_4 \end{array} \right) L_3 \\
 & + \left(\begin{array}{l} \left(\begin{array}{l} (S_1 S_2 S_3) C_4 + (C_3 S_1 S_2) S_4 \\ + (C_1 C_3) S_4 \end{array} \right) S_5 + \left(\begin{array}{l} -(S_1 S_2 S_3) S_4 \\ + (C_1 C_3) C_4 \end{array} \right) C_5 \\ \left(\begin{array}{l} (S_1 S_2 S_3) C_4 + (C_3 S_1 S_2) S_4 \\ + (C_1 C_3) S_4 \end{array} \right) S_5 + \left(\begin{array}{l} -(S_1 S_2 S_3) S_4 \\ + (C_1 C_3) C_4 \end{array} \right) C_5 \end{array} \right) \\
 L_2 + S_1 C_2 S_6 L_2 + \left(\begin{array}{l} \left(\begin{array}{l} (S_1 S_2 S_3) C_4 + (C_3 S_1 S_2) S_4 \\ + (C_1 C_3) S_4 \end{array} \right) S_5 + \left(\begin{array}{l} -(S_1 S_2 S_3) S_4 \\ + (C_1 C_3) C_4 \end{array} \right) C_5 \\ \left(\begin{array}{l} (S_1 S_2 S_3) C_4 + (C_3 S_1 S_2) S_4 \\ + (C_1 C_3) S_4 \end{array} \right) S_5 + \left(\begin{array}{l} -(S_1 S_2 S_3) S_4 \\ + (C_1 C_3) C_4 \end{array} \right) C_5 \end{array} \right) \\
 C_6 L_3 + S_1 C_2 S_6 L_1 + \left(\begin{array}{l} \left(\begin{array}{l} (S_1 S_2 S_3) C_4 + (C_3 S_1 S_2) S_4 \\ + (C_1 C_3) S_4 \end{array} \right) S_5 + \left(\begin{array}{l} -(S_1 S_2 S_3) S_4 \\ + (C_1 C_3) C_4 \end{array} \right) C_5 \\ \left(\begin{array}{l} (S_1 S_2 S_3) C_4 + (C_3 S_1 S_2) S_4 \\ + (C_1 C_3) S_4 \end{array} \right) S_5 + \left(\begin{array}{l} -(S_1 S_2 S_3) S_4 \\ + (C_1 C_3) C_4 \end{array} \right) C_5 \end{array} \right) \\
 C_6 L_1 \\
 r_3 = & C_2 C_3 L_3 + C_2 L_3 + L_1 + L_2 + (C_2 C_3 C_4 - C_2 S_3 S_4) L_3 + \\
 & ((C_2 C_3 S_4 + C_2 C_4 S_3) S_5 + (C_2 C_3 C_4 - C_2 S_3 S_4) C_5) L_2 \\
 & - S_2 S_6 L_2 + \left(\begin{array}{l} (C_2 C_3 S_4) S_5 + (C_2 C_3 C_4) C_5 \\ + (C_2 C_4 S_3) S_5 + (-C_2 S_3 S_4) C_5 \end{array} \right) C_6 L_2 - S_2 S_6 L_1 \\
 & + ((C_2 C_3 S_4 + C_2 C_4 S_3) S_5 + (C_2 C_3 C_4 - C_2 S_3 S_4) C_5) C_6 L_1
 \end{aligned}$$

REFERENCES

[1] H. Crujissen, M. Ellenbroek, M. Henderson, H. Petersen, P. Verzijden, M. Visser, The European Robotic Arm: A High-Performance Mechanism Finally on its way to Space, in 42nd Aerospace Mechanisms Symposium, NASA Goddard Space Flight Center, 2014

[2] P.J. Lambooy, W.M. Mandersloot, R.H. Bentall, Some mechanical design aspects of the European Robotic Arm, Proceedings of the 29th Aerospace Mechanisms Symposium. Johnson Space Center, Houston, Texas, (1995) 17-29.

[3] R. Boumans, C. Heemskerk, The European robotic arm for the international space station, Robotics and Autonomous Systems, 23(1-2)

(1998) 17-27.

[4] F. Feng, L. Tang, J. Xu, H. Liu, Y. Liu, A review of the end-effector of large space manipulator with capabilities of misalignment tolerance and soft capture, Science China Technological Sciences, 59(11) (2016) 1621-1638.

[5] F. Romanelli, Hybrid control techniques for static and dynamic environments: a step towards robot-environment interaction, in: Robot Manipulators New Achievements, IntechOpen, 29 (2010) 551-576.

[6] F.A.A. Cheein, F. di Sciascio, J.M. Toibero, R. Carelli, Robot Manipulator Probabilistic Workspace Applied to Robotic Assistance, Robot Manipulators New Achievements, IntechOpen, 1(1) (2011).

[7] W. Liu, D. Chen, J. Steil, Analytical inverse kinematics solver for anthropomorphic 7-DOF redundant manipulators with human-like configuration constraints, Journal of Intelligent & Robotic Systems, 86(1) (2017) 63-79.

[8] K.D. Kendricks, A kinematic analysis of the gmf a-510 robot: An introduction and application of groebner basis theory, Journal of Interdisciplinary Mathematics, 16(2-3) (2013) 147-169.

[9] K. Kendricks, Solving the inverse kinematic robotics problem: A comparison study of the Denavit-Hartenberg matrix and Groebner basis theory, PhD Thesis, Auburn University Libraries, Auburn, Alabama, 2007

[10] J.F.A. Diaz, M.S. Dutra, F.A.d.N.C. Pinto, Kinematical and dynamical models of KR 6 KUKA robot, including the kinematic control in a parallel processing platform, Robot manipulators new achievements, IntechOpen, (2010) 601-620.

[11] C. Lee, M. Ziegler, Geometric approach in solving inverse kinematics of PUMA robots, IEEE Transactions on Aerospace and Electronic Systems, (6) (1984) 695-706.

[12] S.N. Nabavi, A. Akbarzadeh, J. Enferadi, A Study on Kinematics and Workspace Determination of a General 6-P US Robot, Journal of Intelligent & Robotic Systems, 91(3-4) (2018) 351-362.

[13] Z. Xie, Kinematic, dynamic and accuracy reliability analysis of 6 degree-of-freedom robotic arm, PhD Thesis, University of Missouri, Columbia, 2013

[14] M. Shahinpoor, M. Jamshidi, Y.T. Kim, Exact solution to the inverse kinematics problem of a standard 6-axis robot manipulator, Journal of robotic systems, 3(3) (1986) 259-280.

[15] J.D. Sun, G.Z. Cao, W.B. Li, Y.X. Liang, S.-D. Huang, Analytical inverse kinematic solution using the DH method for a 6-DOF robot, in: 2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), IEEE, 2017, pp. 714-716.

[16] J. Xie, W. Qiang, B. Liang, C. Li, Inverse kinematics problem for 6-DOF space manipulator based on the theory of screws, in: 2007 IEEE International Conference on Robotics and Biomimetics (ROBIO), IEEE, 2007, pp. 1659-1663.

[17] S. Sagara, Y. Taira, Digital Control of Free Floating Space Robot Manipulators Using Transpose of Generalized Jacobian Matrix, in: Robot Manipulators New Achievements, IntechOpen, (2010) 361.

[18] I. Duleba, Kinematic Models of Doubly Generalized N-trailer Systems, Journal of Intelligent & Robotic Systems, 94(1) (2019) 135-142.

[19] W. Xu, Z. Mu, T. Liu, B. Liang, A modified modal method for solving the mission-oriented inverse kinematics of hyper-redundant space manipulators for on-orbit servicing, Acta Astronautica, 139 (2017) 54-66.

[20] Z.Y. Li, D.J. Zhao, J.S. Zhao, Structure synthesis and workspace analysis of a telescopic spraying robot, Mechanism and Machine Theory, 133 (2019) 295-310.

[21] J. Oh, H. Bae, J.-H. Oh, Analytic inverse kinematics considering the joint constraints and self-collision for redundant 7DOF manipulator, in: 2017 First IEEE International Conference on Robotic Computing (IRC), IEEE, 2017, pp. 123-128.

[22] W. Xu, Z. Hu, L. Yan, H. Yuan, B. Liang, Modeling and planning of a

- space robot for capturing tumbling target by approaching the Dynamic Closest Point, *Multibody System Dynamics*, 47(3) (2019) 203-241.
- [23] R. Ghaedrahmati, A. Raoofian, A. Kamali, A. Taghvaeipour, An enhanced inverse dynamic and joint force analysis of multibody systems using constraint matrices, *Multibody System Dynamics*, 46(4) (2019) 329-353.
- [24] M. Ellenbroek, J. Schilder, On the use of absolute interface coordinates in the floating frame of reference formulation for flexible multibody dynamics, *Multibody system dynamics*, 43(3) (2018) 193-208.
- [25] A. Müller, Screw and Lie group theory in multibody kinematics, *Multibody System Dynamics*, 43(1) (2018) 37-70.
- [26] K. Komoda, H. Wagatsuma, Energy-efficacy comparisons and multibody dynamics analyses of legged robots with different closed-loop mechanisms, *Multibody System Dynamics*, 40(2) (2017) 123-153.
- [27] J.H. Ginsberg, *Advanced engineering dynamics*, Cambridge University Press, 1998.

HOW TO CITE THIS ARTICLE

A. Zamanzadeh, H. Ahmadi, Inverse kinematic of European Robotic Arm based on a new geometrical approach, AUT J. Mech. Eng., 5(1) (2021) 13-48

DOI: [10.22060/ajme.2020.17642.5866](https://doi.org/10.22060/ajme.2020.17642.5866)



