Utilizing Inverse Kinematics for Precise Guidance in Planning 6-DoF Robot End-Effector Movements

Heru Suwoyo
Department of Electrical Engineering, Universitas Mercu Buana, Indonesia

Andi Adriansyah
Department of Electrical Engineering, Universitas Mercu Buana, Indonesia

Julpri Andika
Department of Electrical Engineering, Universitas Mercu Buana, Indonesia

Muhammad Hafizd Ibnu Hajar
Department of Electrical Engineering, Universitas Mercu Buana, Indonesia

Rizky Dinata
Department of Product Design, Universitas Mercu Buana, Indonesia

Thathit Gumilar Triwidya Mochtar
Department of Electrical Engineering, Universitas Mercu Buana, Indonesia

Muhammad Yusuf
Department of Electrical Engineering, Universitas Mercu Buana, Indonesia

Fajri Rezki Hutomo
Department of Electrical Engineering, Universitas Mercu Buana, Indonesia

Abstract: The solution of the kinematic inverse determines a substantial part of the robotic arm's control accuracy. Researchers frequently employ standard problem-solving techniques such as numerical, algebraic, iterative, and geometric methods. Although geometric like trigonometrical method has been widely studied, and their application is strongly dependent on the shape and dimensions of the robot. The complexity of the steps makes this approach difficult for researchers. In order to give a clearance and easiness, the step-by-step features of inverse kinematics are described in this research. The study begins with forward kinematics and refers to the DH-parameter in Homogeneous Matrix Transformations calculation. The

Correspondents Author:
Julpri Andika, Department of Electrical Engineering, Universitas Mercu Buana, Indonesia
Email: julpri.andika@mercubuana.ac.id

Received May 17, 2023; Revised November 5, 2023; Accepted December 7, 2023; Published March 1, 2024

existence of specific elements applied to mathematical derivation constituted the basis of forward kinematic discussions. And based on geometrical analysis, the inverse kinematic is then derived. Furthermore, simulations are performed to demonstrate the actual implementation of IK and the solution is then used to initiate the path planning process.

Keywords: Accuracy, Path Planning, Forward and Inverse Kinematic, DH-Parameters, Homogeneous Transformation Matrix.

Introduction

Today's aberrant robotic arm behavior is common. Instead of being able to function as anticipated, labor faults will really harm its users. In his activity as a carving machine, for example, faults induced by imprecision actively exacerbate objects (Putra & Risfendra, 2022; Silfia & Risfendra, 2022; Ekarinda et al., 2021; Arifin, 2017). This is not due to the robot's fault, but rather to a lack of adequate process planning. These issues typically include the robot's inability to maintain the direction of travel due to excessive load from the end-effector, restricted range in specific directions, and hard material that inhibit movement (Andika & Salamah, 2018; Frankovský et al., 2016). As a result, it is believed that actions should be taken to reduce such incidents. One method is to create simulators and observe the behavior of the arm robot that you want to operate.

Simulators are required for users to better comprehend path planning algorithms, arm movements, and so forth. The necessity for simulators will allow learning the properties of a robot prior to actual use easier. Furthermore, by simulating, faults in realization may be eliminated, and motions on the robot can be appropriately planned. In general, the movement of manipulator robots required the use of kinematics systems (Syukranullah et al., 2019; Iskandar et al., 2020; Purwoto, 2020; Kopei et al., 2020; Megalingam et al., 2018; Munadi, 2013). In general, kinematics is separated into two categories: forward and reverse kinematics. Each of its applications is to forecast the position of the end-effector (Nugraha, 2021) when all angular motions are given and to determine the necessity for angular motion to reach the desired end-effector position. Thor, the 6 DoF robot arm, must be mathematically analyzed and applied using shapes and information connected to its dimensions (Laksana et al., 2022; Nurkholik et al., 2022; Prasetyo & Sutopo, 2018; Zaki, 2019). Based on its function, this analysis should incorporate both forward and inverse kinematics at the same time. The viability of both forward and inverse solutions must be established. Furthermore, understanding of assembly is highly crucial to do (Prabanegara et al., 2015; Sun et al., 2022; Cecil et al., 2005). Although just employing software such as CAD, the assembly process will provide logical insight into the robot arm's agility. This endeavor was undertaken because of the urgency and motivation. The DH parameters (Luo & Andika, 2018; Utomo & Munadi, 2013) and the Homogeneous Transformation Matrix (Zurendra et al., 2020; Ye et al., 2020; Zhang et al., 2022; Cashbaugh & Kitts, 2018) are used for forward kinematics. While numerical approaches to inverse kinematics are common. Furthermore, both are implemented with the Python programming language based on these calculations, which includes the building of a Graphical Unit Interface for forward and end-effector prediction for kinematic inverse.

Finally, both will be confirmed by comparing them to FreeCad-Assisted RoboDK simulations. A path planning is carried out after the simulation has been successful. On roboDK, the 6 Dof arm robot is simulated using online and offline programming. In the case of online programming, all targets are determined by hand. The robot is moved as requested during offline programming by using RoboDK-compatible API-python. This is a preliminary implementation that is carried out in a sequential manner. The purpose of a project like this is to optimize the way the 6 DoF robotic arm is used.

Research Method

This 6-DOF robotic manipulator includes six joints, all of which rotate or revolute. Each joint is linked by a big link that is customized for the robot. Figure 1 shows a 6 DoF robotic arm utilized in this research.

$$L_1 =, L_2 =, L_3 = \ , L_4 = \qquad (1)$$

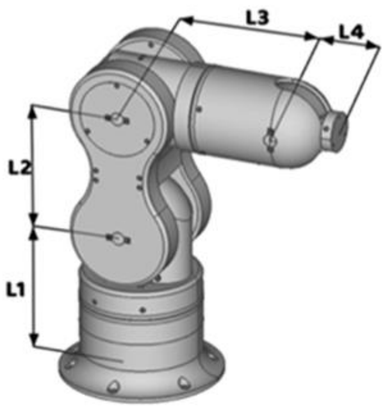

Figure 1 Model CAD Robot

Furthermore, the robot's representation in Figure 1 is regarded the home position.

Forward Kinematic

In a nutshell, the Denavit-Hartenberg Convention can be used to solve forward kinematics. The steps are as follows: Set frames in accordance with Denavit- Hartenberg (DH) rule 4.

Create a parameter DH table, replace the DH parameters in the Homogeneous Transformation Matrix (HTM), and multiply all matrices. The Frame Assignment in Figure 2 is obtained by following the first rule and applying the following DH-specific rules.

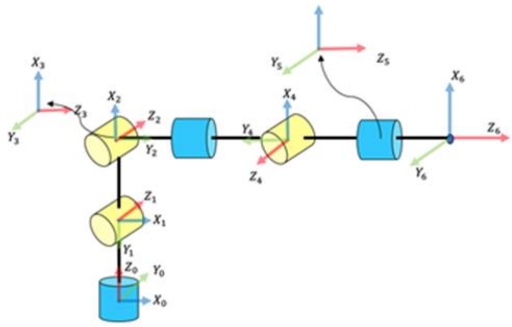

Figure 2 Home Position Frame Assignment

Furthermore, the values of all parameters listed in Table 1 are obtained based on the Frame Assignment and DH parameter descriptions presented at the end of this subsection.

Table 1 D-H Parameters

***i-th* joint**	θ	α	**r**	**d**
1	$0+\theta_1$	-90	0	L_1
2	$-90+\theta_2$	0	L_2	0
3	$0+\theta_3$	-90	0	0
4	$0+\theta_4$	90	0	L_3
5	$0+\theta_5$	-90	0	0
6	$0+\theta_6$	0	0	L_4

Then, using the following mathematical definition of HTM (2), substitute all of the values in the Table 1.

$$H_i^{i-1} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & \alpha_{i\cos\theta_i} \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & \alpha_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

Then, for each joint relationship, obtain HTM as follows. (3)

$$H_1^0 = \begin{bmatrix} \cos\theta_1 & 0 & -\sin\theta_1 & 0 \\ \sin\theta_1 & 0 & \cos\theta_1 & 0 \\ 0 & 1 & 0 & L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H_2^1 = \begin{bmatrix} \sin\theta_2 & \cos\theta_2 & 0 & L_2\sin\theta_2 \\ -\cos\theta_2 & \sin\theta_2 & 0 & -L_2\cos\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

$$H_3^2 = \begin{bmatrix} \cos\theta_3 & 0 & -\sin\theta_3 & 0 \\ \sin\theta_3 & 0 & \cos\theta_3 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H_4^3 = \begin{bmatrix} \cos\theta_4 & 0 & \sin\theta_4 & 0 \\ \sin\theta_4 & 0 & -\cos\theta_4 & 0 \\ 0 & 1 & 0 & L_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H_5^4 = \begin{bmatrix} \cos\theta_5 & 0 & -\sin\theta_5 & 0 \\ \sin\theta_5 & 0 & \cos\theta_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H_6^5 = \begin{bmatrix} \cos\theta_6 & -\sin\theta_6 & 0 & 0 \\ \sin\theta_6 & \cos\theta_6 & 0 & 0 \\ 0 & \sin\alpha_i & 0 & L_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Finally, a forward solution (4) is achieved by multiplying all of the homogeneous matrices (3).

$$H_6^5 = (H_1^0 H_2^1)(H_3^2 H_4^3)(H_5^4\ H_6^5)$$

$$H_1^0 H_2^1 = \begin{bmatrix} \cos\theta_1 \sin\theta_2 & \cos\theta_1 \cos\theta_2 & -\sin\theta_1 & L_2 \cos\theta_1 \sin\theta_2 \\ \sin\theta_1 \sin\theta_2 & \sin\theta_1 \cos\theta_2 & \cos\theta_1 & L_2 \sin\theta_1 \sin\theta_2 \\ -\cos\theta_2 & \sin\theta_2 & 0 & -L_2 \cos\theta_2 + L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H_3^2 H_4^3 = \begin{bmatrix} \cos\theta_3 \sin\theta_4 & -\sin\theta_3 & \cos\theta_3 \sin\theta_4 & -L_3 \sin\theta_3 \\ \sin\theta_3 \sin\theta_4 & \cos\theta_3 & \sin\theta_3 \sin\theta_4 & L_3 \cos\theta_3 \\ 0 & 0 & -\cos\theta_4 & L_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$H_5^4 H_6^5 = \begin{bmatrix} \cos\theta_5 \cos\theta_6 & -\cos\theta_5 \sin\theta_5 - \sin\theta_5 & 0 & -L_4 \sin\theta_5 \\ \sin\theta_5 \cos\theta_6 & -\sin\theta_5 \sin\theta_6 + \cos\theta_5 & 0 & L_4 \cos\theta_5 \\ \sin\theta_6 & \cos\theta_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Inverse Kinematic

In general, kinematic inverse is a method that maps the end-effector's cartesian information to a great degree of all manipulator joints. Here are the steps taken to solve the problem: Create a kinematic diagram referencing the first three joints, then use a geometric approach to derive the kinematic inverse for the position. Forward kinematic calculations on the first three joints yield component rotation (R0_3). Find the R0_3 matrix's kinematic inverse. Forward kinematics on the last three joints and intriguing component rotation, R3_6 Specifications will be rotated on the following R0_6 matrix, and Give the appropriate X, Y, and Z positions to finish the first three joints from the first stage using kinematic inverse. Following the first stage, the following depiction of the first three joints is obtained:

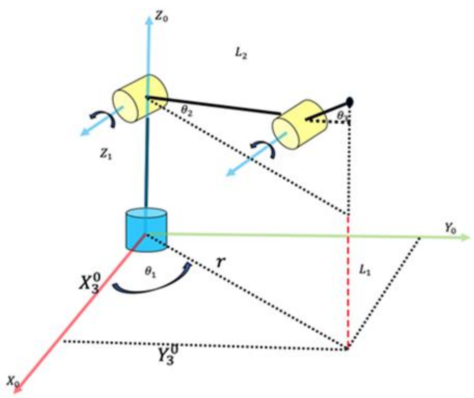

Figure 3 Analysis of the First Three Joints' Basic Geometry

As a result, as illustrated in Figure 4, a thorough picture from a new point of view is obtained.

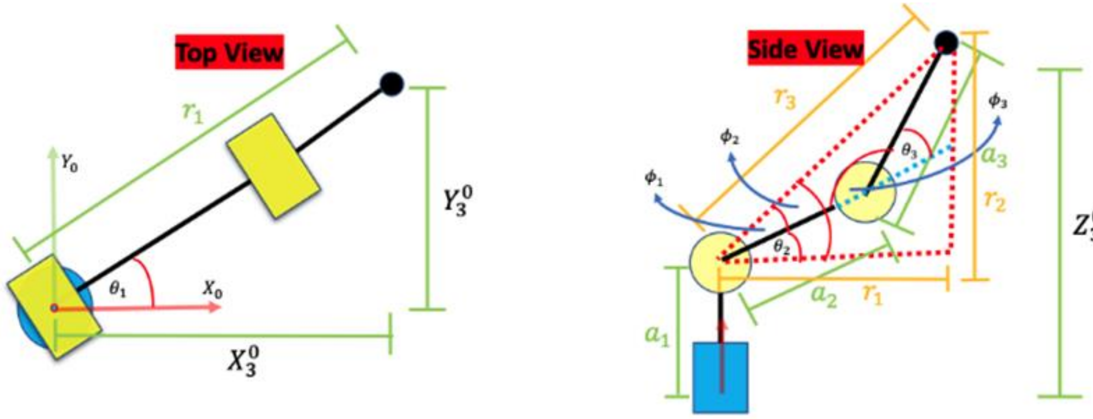

Figure 4 The First Three Joints are Visually Represented

The equation is obtained by referring to the Figure 4.

$$\theta_1 = \tan^{-1}\left(\frac{Y_3^0}{X_3^0}\right) \tag{5}$$

$$r_1 = \sqrt{{X_3^0}^2 + {Y_3^0}^2} \tag{6}$$

$$\theta_2 = \phi_2 - \phi_1 \tag{7}$$

$$\theta_3 = 180º - \phi_3 \tag{8}$$

$$r_2 = Z_3^0 - a_3 \tag{9}$$

$$r_3 = \sqrt{{r_1}^2 + {r_2}^2} \tag{10}$$

$${a_3}^2 = {a_2}^2 + {r_3}^2 - 2a_2r_3\, cos\, \phi_1 \tag{11}$$

$$\phi_1 = \cos^{-1}\left(\frac{{a_3}^2 - {a_2}^2 - {r_3}^2}{-2a_2r_3}\right) \tag{12}$$

$${r_3}^2 = {a_2}^2 + {a_3}^2 - 2a_3a_2\, cos\, \phi_3 \tag{13}$$

$$\phi_3 = \cos^{-1}\left(\frac{{r_3}^2 - {a_2}^2 - {a_3}^2}{-2a_3a_2}\right) \tag{14}$$

$$\phi_2 = \tan^{-1}\left(\frac{r_2}{r_1}\right) \tag{15}$$

The Rotation Matrix is then obtained by using the second step and referring to Table 1 by multiplying the homogeneous matrix of the last three joints, as shown below

$$R_6^3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{16}$$

With the configuration based on this linearity $R = [r_{ij}]$ for $1 \le i \le 3$ and $1 \le j \le 3$, and formally multiply $R_x(\theta_x)R_y(\theta_y)R_z(\theta_z)$, and obtained

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} \cos\theta_4\cos\theta_5\cos\theta_6 - \sin\theta_4\sin\theta_6 & -\cos\theta_4\cos\theta_5\sin\theta_6 - \sin\theta_4\cos\theta_6 & -\cos\theta_4\,\mathrm{s} \\ \sin\theta_4\cos\theta_5\cos\theta_6 + \cos\theta_4\sin\theta_6 & -\sin\theta_4\cos\theta_5\sin\theta_6 + \cos\theta_4\cos\theta_6 & -\sin\theta_4\,\mathrm{si} \\ \sin\theta_5\cos\theta_6 & -\sin\theta_5\sin\theta_6 & \cos\theta_5 \end{bmatrix} \tag{17}$$

The simplest form is given by $r_{33} = cos\,\theta_5$, which means that $\theta_5 = cos\, cos\, (r_{33})$. In light of this result, there are three possibilities:

First Case

If $0 < \theta_5 < \frac{\pi}{2}$ or $0 > \theta_5 > -\frac{\pi}{2}$, so $sin\, sin\, \theta_5 \neq 0$ and $sin\, sin\, \theta_5\, (cos\, cos\, \theta_4, sin\, sin\, \theta_4) = (-r_{13}, -r_{23})$, in this case $\theta_4 = (-r_{13}/-r_{23})$. And $sin\, \theta_5(sin\, sin\, \theta_6, cos\, cos\, \theta_6) = (-r_{32}, r_{31})$ in this case $\theta_6 = (-r_{32}/r_{31})$.

Second Case

If $\theta_5 = 0$, so $sin\, sin\, \theta_5 = 0$ and $cos\, cos\, \theta_5 = 1$. In this case

$$\begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} = \begin{bmatrix} \cos\theta_4\cos\theta_5\cos\theta_6 - \sin\theta_4\sin\theta_6 & -\cos\theta_4\cos\theta_5\sin\theta_6 - \sin\theta_4\cos\theta_6 \\ \sin\theta_4\cos\theta_5\cos\theta_6 + \cos\theta_4\sin\theta_6 & -\sin\theta_4\cos\theta_5\sin\theta_6 + \cos\theta_4\cos\theta_6 \end{bmatrix} \tag{18}$$

$$= \begin{bmatrix} cos\,(\theta_4 + \theta_6) & -(sin\,sin\,(\theta_4 + \theta_6)) \\ sin\,(\theta_4 + \theta_6) & cos\,(\theta_4 - \theta_6) \end{bmatrix}$$

Thereby $\theta_4 + \theta_6 = atan2(r_{21}, r_{11})$. In addition, one of the angles in its implementation might be both constant and user-defined.

Result and Discussion

The following findings are generated in Figure 5 by applying the results of forward kinematic calculations in Python and GUI design. There are six bars, the highest bar is Theta 1 with a range of -200 to 200, theta 2 (-90 to 90), theta 3 (-180 to 0), theta 4 (-200 to 200), theta 5 (-90 to 90) and theta 6 (-200 to 200). Based on this specification, the offline calculation is observed. This is done by generating a series of end-effector's pose mimicking the planned path to be tested. Some points representing the expected position of robot's end-effector are generated. These points are stored in Table 2.

Table 2 Random Point Representing End-Effector's Pose

Sequential Pose	x pose	y-pose	z-pose
1st	76.9436	61.0511	30.3762
2nd	29.3008	72.3268	8.2325
3rd	1.5074	67.8665	25.7454
4th	63.3445	77.5459	21.7731
5th	85.4617	33.5063	36.9817
6th	82.7820	39.9881	19.8512
7th	11.1765	65.8522	31.1772
8th	43.6237	62.0568	6.9790
9th	8.8457	4.2469	0.4633
10th	41.9983	60.9087	25.7513

According to these points, the generated solution is test. It is tested by running (1)-(18) on Python. In order to get easiness, it is modelled as a function with output of all the angle for each joint of robot arm and end-effector's point becomes the input. Regarding to this scenario, the result representing the angle for each joint is obtained as shown in Table 3.

Table 3 Representative Angle for All Joints

Seq.Pose of End-Effector	Theta 1 (Degree)	Theta 2 (Degree)	Theta 3 (Degree)	Theta 4 (Degree)	Theta 5 (Degree)	Theta 6 (Degree)
1st	0. 4906	15.1139	-19.6843	23.6434	15.8982	45.9147
2nd	0. 3549	5.2496	-6.8370	8.2121	5.5220	15.9477
3rd	0. 1139	4.1116	-5.3549	6.4319	4.3249	12.4906
4th	0. 6645	11.9520	-15.5662	18.6970	12.5722	36.3090
5th	0. 5722	17.2791	-22.5042	27.0304	18.1757	52.4921
6th	0. 4921	14.3144	-18.6430	22.3926	15.0572	43.4857

7th	0. 3109	6.2558	-8.1476	9.7862	6.5805	19.0046
8th	0.1123	7.0150	-9.1363	10.9739	7.3790	21.3109
9th	0.0984	1.2781	-1.6645	1.9993	1.3444	3.8826
10th	0.0763	9.6410	-12.5564	15.0818	10.1413	29.2884

Up to this point, each value of angle generated based on the offline approach is then compared with the result shown by simulation. This simulation used in this study is RoboDK. The comparison is then represented in form of graph of mean square error as can e seen in Figure 5.

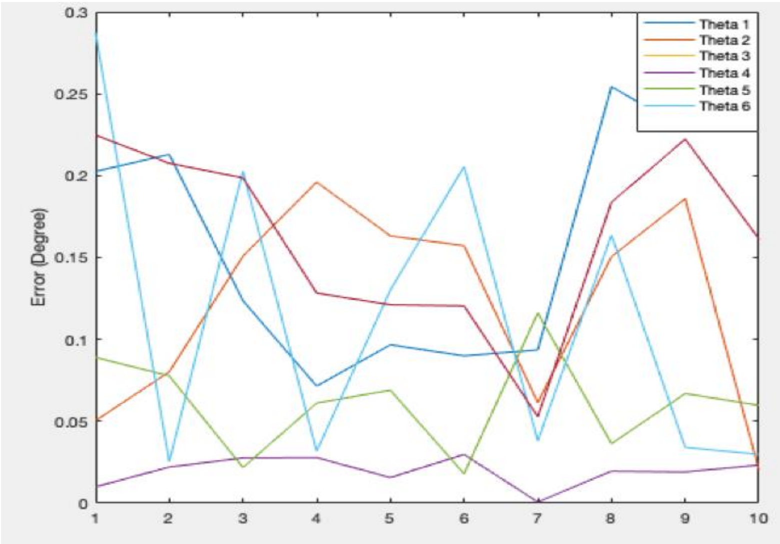

Figure 5 Representative Error

The error presented in Figure 5 is found by considering that the simulation conducted in RoboDK is a basis. As an example, the Figure 6 is presented

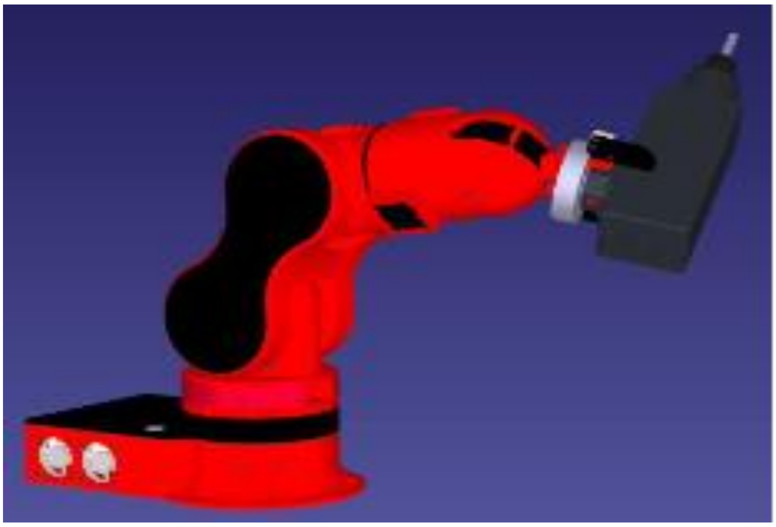

Figure 6 Verify movement with RoboDK

The end-effector’s pose is set to (90,0,100) and the gained angle for each joints are theta1 = 0.0 degree, theta2 = 27.4590902095760008 degree, theta3 = -35.7626088885434396 degree theta4 = 42.955444590727566 degree, theta5 = 28.88401503870459 degree, and theta6 = 83.41804440081549 degree. However, the result given by offline programming are $\theta_1 = 0, \theta_2 = 27.45$, $\theta_3 = -35.76$, $\theta_4 = 42.95$, $\theta_5 = 28.8$, $\theta_6 = 83.41$, in which it shows there is no much error.

As seen in the preceding results, the appropriateness of two distinct calculations has been met. The offline programming process is conducted out using this knowledge base. The goal of this programming is to have the robot behave as a CNC milling machine, with a path chiseled in the shape of the letter “UMB”. Setting goals is one of the most important aspects of planning. This procedure can be summarized as follows:

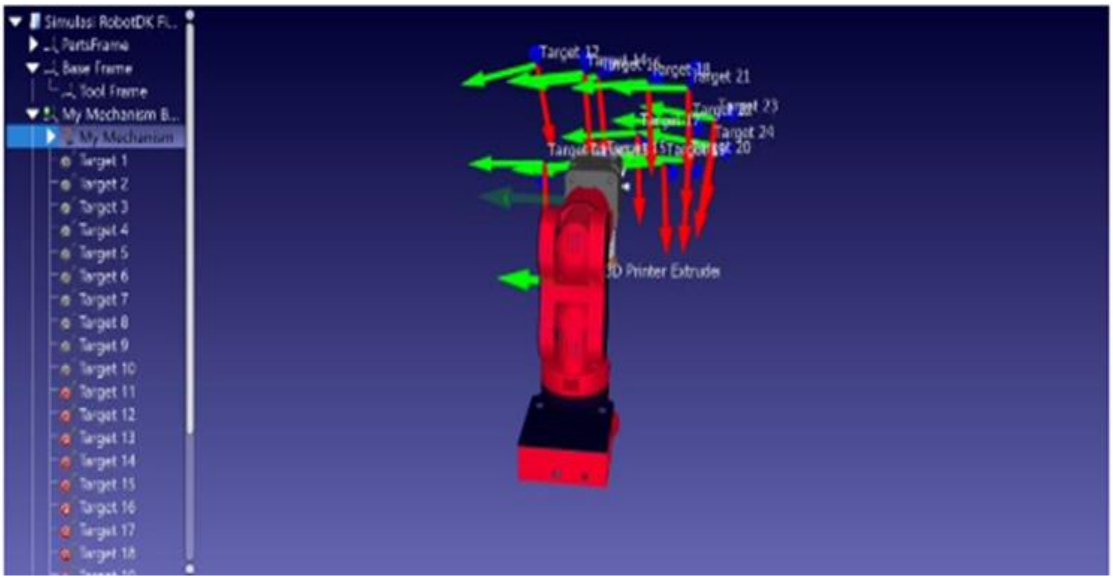

Figure 7 Robotic target recognition

There are 21 well defined targets in this simulation; each target has been measured and is within the manipulator's reach. This procedure enables the path to be generalized, representing the robot's performance in relation to the CNC machine's function at the same time. Figure 8 shows the outcomes of this procedure.

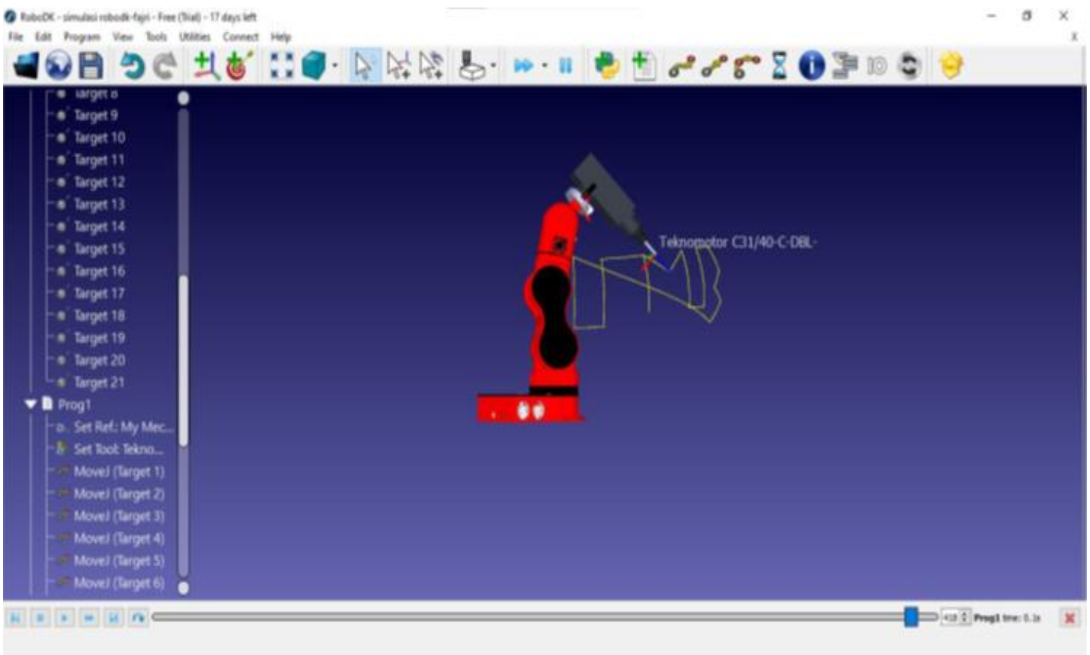

Figure 8 The results of developing robotic targets

Conclusions

In this study the inverse kinematic based on geometrical analysis is presented. By involving DH-parameter through Homogeneous Transformation the forward kinematic is calculated. The foundation of forward kinematic discussions lies in the application of particular elements to mathematical derivations. Subsequently, the inverse kinematics is deduced through a process of geometric analysis. According to this achievement, the simulation is then conducted by running a Python code. The result was then compared to the performance of simulation of RoboDK platform. Based on the comparative result when the end-effector is set, there is no much difference. It is then considered as the basic that the calculation process is conducted well.

Acknowledgements

We acknowledge the financial support received from a Research and Innovation for Advanced Indonesia (RIIM) grant of The Indonesia Endowment Funds for Education (LPDP) and National Research and Innovation Agency (BRIN) with cooperation agreement number 103/IV/KS/11/2022 and 01/065/MOA/XI/2022.

References

Andika, J., & Salamah, K. S. (2018). Analisis Kinematik Pada Robot Hexapod. *Jurnal Teknologi Elektro*, *9*(2), 83-91. https://doi.org/10.22441/jte.v9i2.4072.

Arifin, M. A. S. (2017). Rancang Bangun Prototype Robot Lengan Menggunakan Flex Sensor Dan Accelerometer Sensor Pada Lab Mikrokontroler Stmik Musirawas. *ILKOM Jurnal Ilmiah*, *9*(3), 255-261. https://doi.org/10.33096/ilkom.v9i3.152.255-261.

Cashbaugh, J., & Kitts, C. (2018). Automatic calculation of a transformation matrix between two frames. *IEEE Access*, *6*, 9614-9622. 10.1109/ACCESS.2018.2799173.

Cecil, J., Vasquez, D., & Powell, D. (2005). A review of gripping and manipulation techniques for micro-assembly applications. *International Journal of Production Research*, *43*(4), 819-828.. doi: 10.1080/00207540512331311813.

Ekarinda, H. H., Purwanto, D., & Mardiyanto, R. (2021). Purwarupa Lengan Robot Penyuap Makanan Berbasis Estimasi Posisi Mulut. *Jurnal Teknik ITS*, *10*(1), A44-A49. https://doi.org/10.12962/j23373539.v10i1.59565.

Frankovský, P., Hroncová, D., Delyová, I., & Hudák, P. (2012). Inverse and forward dynamic analysis of two link manipulator. *Procedia Engineering*, *48*, 158-163. https://doi.org/10.1016/j.proeng.2012.09.500.

Iskandar, F. R., Sucahyo, I., & Yantidewi, M. (2020). Penerapan Metode Invers kinematik Pada Kontrol Gerak Robot Lengan Tiga Derajat Bebas. *Inovasi Fisika Indonesia*, *9*(2), 64-71. https://doi.org/10.26740/ifi.v9n2.p64-71.

Kopei, V. B., Onysko, O. R., & Panchuk, V. G. (2019, February). Computerized system based on FreeCAD for geometric simulation of the oil and gas equipment thread turning. In *IOP Conference Series: Materials Science and Engineering* (Vol. 477, No. 1, p. 012032). IOP Publishing. https://doi.org/10.1088/1757-899x/477/1/012032.

Laksana, N. A. S., Ariawan, R., Jati, U. S., & Sodikin, J. (2022). Analisis Kinematik Singularty Pada Manipulator 7 DOF Dengan Software Simulasi ROBOAnalyzer. *Infotekmesin*, *13*(2), 265-271. https://doi.org/10.35970/infotekmesin.v13i2.1538.

Luo, Q., & Andika, J. (2018). Analysis Of Kinematic For Legs Of A Hexapod Using Denavit-Hartenberg Convention. *Sinergi*. https://doi.org/10.22441/sinergi.2018.2.001

Megalingam, R. K., Sivanantham, V., Kumar, K. S., Ghanta, S., Teja, P. S., Gangireddy, R., ... & Gedela, V. V. (2018). Design and development of inverse kinematic based 6 dof robotic arm using ROS. *International Journal of Pure and Applied Mathematics*, *118*, 2597-2603.

Munadi, M. (2013). Analisa Forward Kinematic Pada Simulator Arm Robot 5 Dof Yang Mengintegrasikan Mikrokontroler Arduino-Uno Dan Labview. *ROTASI*, *15*(2), 37-43. https://doi.org/10.14710/rotasi.15.2.37-43.

Nugraha, I. D., & Santika, P. M. (2021). Pendekatan Geometri untuk Perhitungan Inverse Kinematics Gerakan Lengan Robot 4 Derajat Kebebasan. *J. Tek. Mesin ITI*, *5*(1) 1. https://doi.org/10.31543/jtm.v5i1.572.

Nurkholik, Z., Fiolana, F. A., & Kusumastutie, D. A. W. (2022). Robotik Arm Rancangan Bangun Lengan Robot Arm Untuk Menggambar Menggunakan Invers Kinematik. *Jurnal Ilmiah Sistem Informasi*, *1*(3), 59-68. https://doi.org/10.51903/juisi.v1i3.413.

Prabanegara, R. H., Noor, M. F., & Kunia, E. (2015). Rancang Bangun Robot Lengan Pemindah dan Penyeleksi Barang Berdasarkan Warna Berbasis Arduino Uno. *Energy-Jurnal Ilmiah Ilmu-Ilmu Teknik*, *5*(2), 31-40.

Prasetyo, H., & Sutopo, W. (2018). Industri 4.0: Telaah Klasifikasi aspek dan arah perkembangan riset. *J@ ti Undip: Jurnal Teknik Industri*, *13*(1), 17-26.. https://doi.org/10.14710/jati.13.1.17-26.

Purwoto, B. H. (2020). Pemodelan robot kinematik manipulator menggunakan matlab. *Emitor: Jurnal Teknik Elektro*, *20*(2), 141-146. https://doi.org/10.23917/emitor.v20i02.11345.

Putra, G. S., & Risfendra, R. (2022). Sistem kontrol lengan robot berbasis Arduino dengan aplikasi GUI. *JTEIN: Jurnal Teknik Elektro Indonesia*, *3*(2), 387-398. https://doi.org/10.24036/jtein.v3i2.270.

Silfia, R., & Risfendra, R. (2022). Record and Playback Robot Lengan Menggunakan Graphical User Interface. *JTEIN: Jurnal Teknik Elektro Indonesia*, *3*(2), 524-532. https://doi.org/10.24036/jtein.v3i2.293.

Sun, C., Orbik, J., Devin, C. M., Yang, B. H., Gupta, A., Berseth, G., & Levine, S. (2022, January). Fully autonomous real-world reinforcement learning with applications to mobile manipulation. In *Conference on Robot Learning* (pp. 308-319). PMLR.

Syukranullah, S., Bukhari, B., & Amalia, I. (2019). Rancang bangun robot lengan berbasis mikrokontroler arduino uno. *jurnal mesin sains terapan*, *3*(1), 7-10. https://doi.org/10.30811/jmst.v3i1.876.

Utomo, B., & Munadi, M. (2013). Analisa forward dan inverse kinematics pada simulator arm robot 5 derajat kebebasan. *Jurnal Teknik Mesin*, *1*(3), 11-20.

Ye, H., Wang, D., Wu, J., Yue, Y., & Zhou, Y. (2020). Forward and inverse kinematics of a 5-DOF hybrid robot for composite material machining. *Robotics and Computer-Integrated Manufacturing*, *65*, 101961. https://doi.org/10.1016/j.rcim.2020.101961.

Zaki, A., & Khairul Jauhari. (2019). Perhitungan Inverse Kinematics dan Pengukuran Akurasi Pergerakan pada Robot Manipulator Lengan. *JTM-ITI (Jurnal Teknik Mesin ITI)*, *3*(2), 53–53. https://doi.org/10.31543/jtm.v3i2.339

Zhang, H., Xia, Q., Sun, J., & Zhao, Q. (2022, September). A Fully Geometric Approach for Inverse Kinematics of a Six-Degree-of-Freedom Robot Arm. In *Journal of Physics: Conference Series* (Vol. 2338, No. 1, p. 012089). IOP Publishing. https://doi.org/10.1088/1742-6596/2338/1/012089.

Zurendra, R. Z. A., Maulana, R., & Fitriyah, H. (2020). Implementasi Inverse Kinematics Pada Robot Lengan Untuk Pengambilan Benda Dengan Koordinat Awal Acak. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, *4*(2), 632-644.