Kai Bing Goh, William Delaporte, Yi Gao & Yixiang Gao
981436, 915100, 1208446, 990803
Department of Mechanical Engineering
The University of Melbourne
VIC 3010, Australia
kaibingg@student.unimelb.edu.au
wdelaporte@student.unimelb.edu.au
yigao4@student.unimelb.edu.au
yixiang1@student.unimelb.edu.au

Robotics Systems Assignment 1 Report
Forward and Inverse Kinematics of Robot

MCEN90028 : Dr. Justin Fong & Dr. Tomislav Bacek
Due –*Friday, April 1, 2022*

Table of Content

1 Introduction

The purpose of this project is design and build a robotic arm that can manipulate chess pieces by moving them around a board. This reports covers the derivation of the forward and inverse kinematics for the manipulator, as well as determinations of the desired link lengths.

The forward kinematics are derived using the Denavit-Hartenberg Convention. The resulting transformations are then used to formulate the transformation matrices used to shift between frames attached to each joint. MATLAB is then used to verify these matrices by plotting a series of configurations.

To determine the desired link lengths, estimates are made based on the measured dimensions of the chess board. Then, MATLAB is used to generate graphs of the reachable workspace in the X-Y and Z-X planes. The estimates are then iteratively tweaked until the entire workspace could be covered.

The inverse kinematics are derived geometrically using trigonometric properties. The results are verified by using MATLAB to calculate the joint angles required for end effector to reach two different squares of the chessboard.

2 Forward Kinematics

2.1 Kinematic Definition with Schematic Diagram in "Zero Position"

Our robot has 4 joins and 1 gripper. The sketch and the coordinate frame shows in figure 1

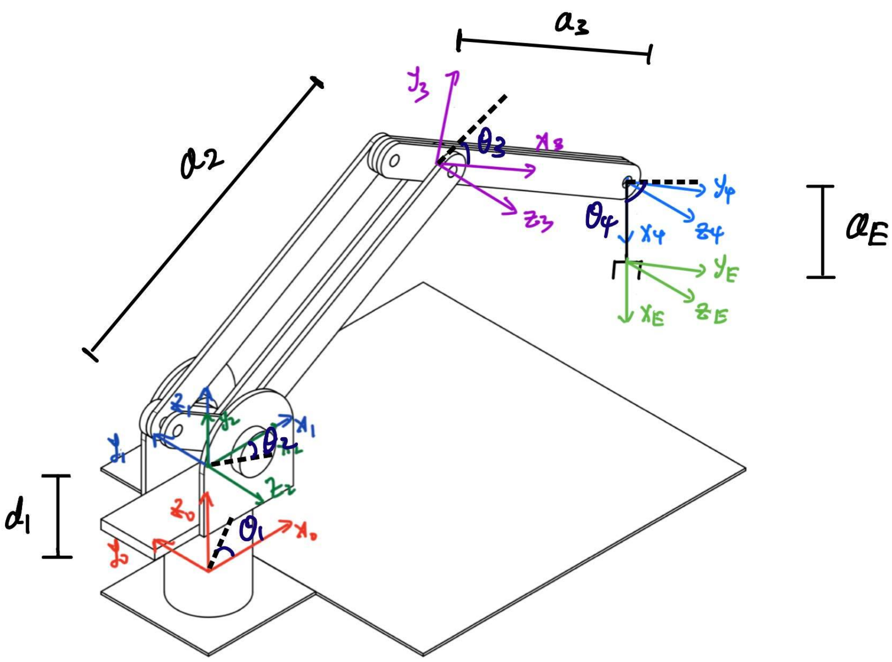


Figure 1: The schematic diagram

2.2 DH Table with Parameters

a_{i-1}	α_{i-1}	d_i	θ_i
0	0	d_1	Q_1
0	90	0	$90 - Q_2$
a_2	0	0	$-Q_3$
a_3	0	0	$-Q_4$
a_E	0	0	0

Table 1: DH table with parameters

The Denavit-Hartenberg table shows the series of transformations required to move from the inertial frame to the end-effector frame.

2.3 Derivation of Transformation Matrix

The function to determine the transformation matrix ${}^{i-1}_{i}T$ is shown in equation 1.

$$ {}^{i-1}_{i}T = D_X(a_i - 1) \times R_X(\alpha_{i-1}) \times D_Z(d_i) \times R_X(\alpha_{i-1}) \quad (1) $$

Using this, we can generate all the transformation matrices:

$$ {}^{0}_{1}T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \text{cos(0)} & \text{-sin(0)} & 0 \\ 0 & \text{sin(0)} & \text{cos(0)} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \text{cos(Q1)} & \text{-sin(Q1)} & 0 & 0 \\ \text{sin(Q1)} & \text{cos(Q1)} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d1 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{1}_{2}T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \text{cos(90)} & \text{-sin(90)} & 0 \\ 0 & \text{sin(90)} & \text{cos(90)} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \text{cos(90-Q2)} & \text{-sin(90-Q2)} & 0 & 0 \\ \text{sin(90-Q2)} & \text{cos(90-Q2)} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{2}_{3}T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \text{cos(0)} & \text{-sin(0)} & 0 \\ 0 & \text{sin(0)} & \text{cos(0)} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & \text{a}_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \text{cos(-Q3)} & \text{-sin(-Q3)} & 0 & 0 \\ \text{sin(-Q3)} & \text{cos(-Q3)} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{3}_{4}T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \text{cos(0)} & \text{-sin(0)} & 0 \\ 0 & \text{sin(0)} & \text{cos(0)} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & \text{a}_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \text{cos(-Q4)} & \text{-sin(-Q4)} & 0 & 0 \\ \text{sin(-Q4)} & \text{cos(-Q4)} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{4}_{E}T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \text{cos(0)} & \text{-sin(0)} & 0 \\ 0 & \text{sin(0)} & \text{cos(0)} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & \text{a}_E \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \text{cos(0)} & \text{-sin(0)} & 0 & 0 \\ \text{sin(0)} & \text{cos(0)} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

The final transformation matrix ${}^0_ET$ can be found via the equation ${}^0_ET = {}^0_1T \times {}^1_2T \times {}^2_3T \times {}^3_4T \times {}^4_ET$. The resulting matrix is given below:

$$\begin{pmatrix} \sigma_2 & \cos(Q_4)\,\sigma_8 + \sin(Q_4)\,\sigma_7 & \sin(Q_1) & a_3\,\sigma_7 + a_E\,\sigma_2 + a_2\cos(Q_1)\,\sigma_9 \\ \sigma_1 & \cos(Q_4)\,\sigma_6 + \sin(Q_4)\,\sigma_5 & -\cos(Q_1) & a_3\,\sigma_5 + a_E\,\sigma_1 + a_2\,\sigma_9\sin(Q_1) \\ -\cos(Q_4)\,\sigma_3 - \sin(Q_4)\,\sigma_4 & \cos(Q_4)\,\sigma_4 - \sin(Q_4)\,\sigma_3 & 0 & d_1 - a_2\,\sigma_{10} - a_E\,(\cos(Q_4)\,\sigma_3 + \sin(Q_4)\,\sigma_4) - a_3\,\sigma_3 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

where

$$\sigma_1 = \cos(Q_4)\,\sigma_5 - \sin(Q_4)\,\sigma_6$$

$$\sigma_2 = \cos(Q_4)\,\sigma_7 - \sin(Q_4)\,\sigma_8$$

$$\sigma_3 = \cos(Q_3)\,\sigma_{10} + \sigma_9\sin(Q_3)$$

$$\sigma_4 = \cos(Q_3)\,\sigma_9 - \sin(Q_3)\,\sigma_{10}$$

$$\sigma_5 = \cos(Q_3)\,\sigma_9\sin(Q_1) - \sin(Q_1)\,\sin(Q_3)\,\sigma_{10}$$

$$\sigma_6 = \cos(Q_3)\,\sin(Q_1)\,\sigma_{10} + \sigma_9\sin(Q_1)\,\sin(Q_3)$$

$$\sigma_7 = \cos(Q_1)\,\cos(Q_3)\,\sigma_9 - \cos(Q_1)\,\sin(Q_3)\,\sigma_{10}$$

$$\sigma_8 = \cos(Q_1)\,\cos(Q_3)\,\sigma_{10} + \cos(Q_1)\,\sigma_9\sin(Q_3)$$

$$\sigma_9 = \cos\left(Q_2 - \tfrac{\pi}{2}\right)$$

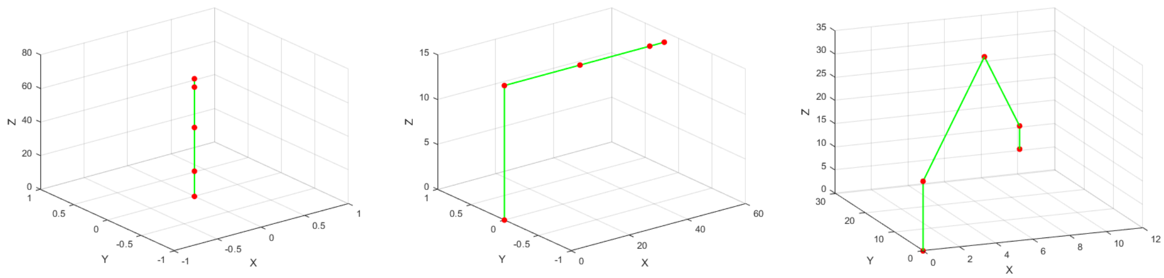
$$\sigma_{10} = \sin\left(Q_2 - \tfrac{\pi}{2}\right)$$

2.4 Forward Kinematics Solution Justification

In this part, we will provide a stick figure plot to showing the positions of limbs when the joints are set to various angles. Each joint's position is calculated using the original point (0, 0, 0) multiplied by the transformation matrices. The equations 2 7 show the five point position function. To draw the figure, we set the $d_1 = 15cm$, $a_2 = 27cm$, $a_3 = 23cm$, $a_E = 10cm$. Furthermore, joint 5 represents the gripper.

$$\begin{aligned} joint1 &= {}^0_1T \times original && (2)\\ joint2 &= {}^0_1T \times {}^1_2T \times original && (3)\\ joint3 &= {}^0_1T \times {}^1_2T \times {}^2_3T \times original && (4)\\ joint4 &= {}^0_1T \times {}^1_2T \times {}^2_3T \times {}^3_4T \times original && (5)\\ joint5 &= {}^0_1T \times {}^1_2T \times {}^2_3T \times {}^3_4T \times {}^E_4T \times original && (6)\\ & && (7) \end{aligned}$$

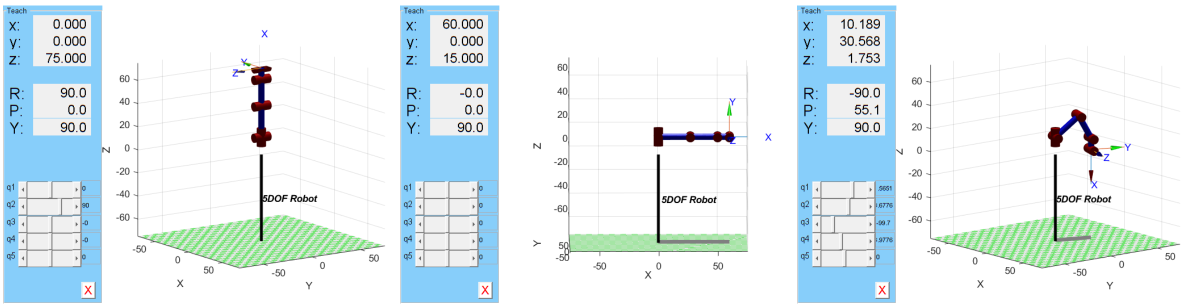
The positions of the arm when q = [0 0 0 0 0], q = [0 90 0 0 0], and q = [71 51 99 29 0] are shown in figure 2.



q = [0 0 0 0 0] q = [0 90 0 0 0] q = [71 51 99 29 0]

Figure 2: Stick figure

The first figure shows the the position of the robot arm in its natural state. When we want to turn the second join 90°, the arm will change to the second figure. Similarly, when we set $Q_1 = 71°$ $Q_2 = 51°$ $Q_3 = 99°$ $Q_4 = 29°$, the arm will turn the reflect angle, the final position is (8, 30, 1). When compared with the solutions generated by MATLAB's robotic system toolbox, shown in figure 3, the results are the same. Therefore, the forward kinematics transformation matrix is correct.



q = [0 0 0 0 0] q = [0 90 0 0 0] q = [71 51 99 29 0]

Figure 3: Stick figure in toolbox

3 Determination of Link Lengths

Before deciding on appropriate link lengths for the robotic arm, the group has measured some important dimensions that would define the necessary specifications required for a valid workspace.

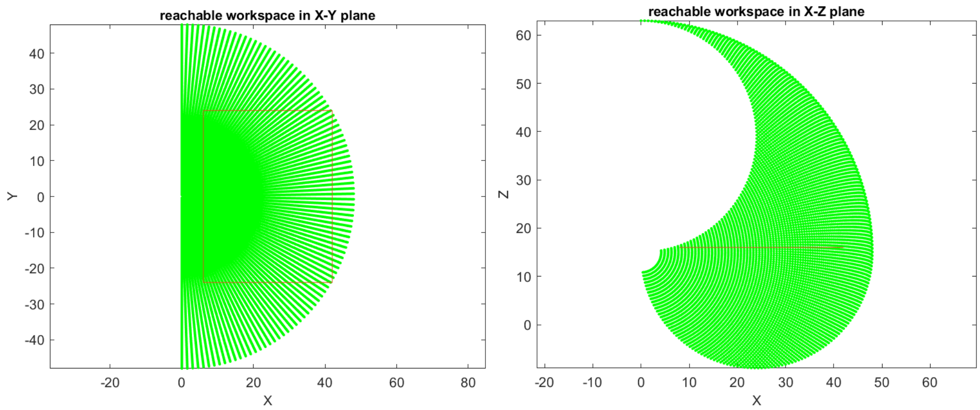
- Chess board width and height: 360mm (8 x 40mm squares + outer blank frame)
- Chess board height: 2.5mm
- Height of tallest chess piece: 60mm (Queen)
- Robot base radius: 60mm
- Robot first joint height: 150mm

3.1 Estimating the Workspace

Initially, the distance of the furthest corners from the center of the base is calculated to estimate the maximum distance the arm is required to reach.

$$\begin{aligned} d_{max} &= \sqrt{(board\_width + base\_radius)^2 + (board\_width/2)^2} \\ d_{max} &= \sqrt{(0.36 + 0.06)^2 + 0.18^2} = 0.4569 \end{aligned} \tag{8}$$

To avoid the arm needing to straighten out completely at the elbow, the lengths of the two largest links were set to 240mm.



(a) (b)

Figure 4: (a)XY Plane (b)XZ Plane

Figure 4 shows the reachable workspace of the robot in the XY and XZ planes. This also includes a red outline representing the size of the chessboard with an extra margin on each side to place pieces that are removed from the board. The current link length dimensions are enough to cover the required space. However, the margin for error is slim, and tolerance issues when manufacturing may render some of the workspace unreachable. Moreover, this configuration would need to flatten out to reach the furthest distance as in figure 4(a). This is not ideal as it would produce the largest amount of torque on the motors. For figure 4(b), we did account that the arms cannot be folded into itself by excluding 10 degrees as implemented in the code.

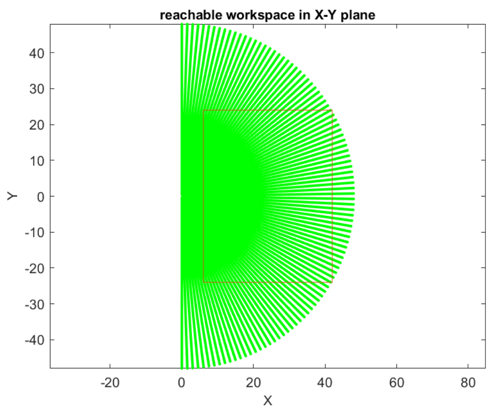
NOTE: Since our gripper (l4 in MATLAB) is assumed to be 10cm long and always facing down, we added that l4 height to the chess height of 6cm. Hence, in the plots, the red line depicting the workspace required is 6cm(chess piece height) + 10cm(gripper height) above ground, where our joint controlling the gripper (green dots) should cover the red lines.

3.2 Iterations to Finalise Link Lengths

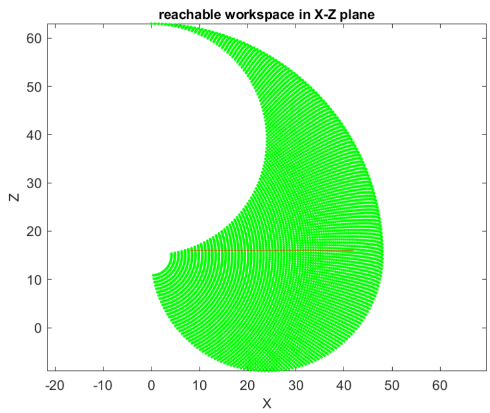
To improve, we have decided to implement a four bar parallelogram linkage for the elbow (3rd joint). This design allows for more support on the first arm, while relocating the weight of the servomotor away from the 3rd joint to the robot base instead, reducing the overall torque on the 2nd joint.

Another improvement was to increase the first arm length, since the weight before the 3rd joint has more support. By doing so, we can reduce the length of the second arm as long as it produces the sufficient reachable workspace. However, an important note regarding this iteration is that the second arm can't be too much shorter relative to the first arm. The larger the ratio of second arm length to first arm length, the larger the inner semi-circular area between the chess board and robot is removed from the reachable workspace. Hence, this introduces the possibility of being unable to pick up pieces on D1/E1 (playing white) or D8/E8 (playing black) squares and should be considered when selecting the parameters.

Therefore, due to the reasons above, an arm length of 270mm is selected for the first arm and 230mm for the second arm. As a result, a workspace with some freedom while reaching all the required workspace is achieved as shown in figure 5.



(a)



(b)

Figure 5: (a)XY Plane (b)XZ Plane

3.3 Remark on the Link Lengths

The over length robotic arm is implicitly constrained. This is because a longer arm would imply a larger arm mass, which in turn means a larger torque acting on the joints as the distance from the end effector grows. This could lead to issues. For example, the stress and strain caused by torque from the load and its own mass could exceed the yield strength of the material, causing mechanical failure. In another case, the increased torque, if exceeds the maximum torque stated in the servomotor's specification sheet, it could overload and burnout the servomotors, causing motor failure.

On the other hand, if the arm is too short, the required reachable workspace for a fully functioning chess robot could not be fully realised, where the resulting workspace may not cover certain areas of the chess board.

4 Inverse Kinematics

4.1 Derivation of Inverse Kinematics

θ_1 is defined as shown in figure 6, which shows the X-Y plane view of the robot. θ_2, θ_3, and θ_4 are defined as shown in figure 7, which is the X-Z plane view of the robot.
In order to keep the gripper is perpendicular to the ground, the constraint $\theta_2 + \theta_3 + \theta_4 = 180°$ is set. Then, the following equations can be found using the trigonometric identities.

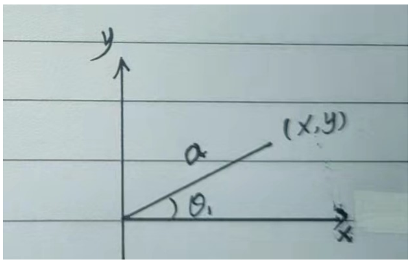


Figure 6: X-Y projection

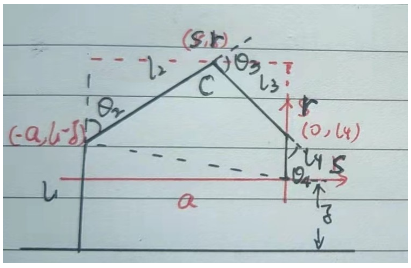


Figure 7: X-Z projection

$$\theta_1 = \arctan(\frac{y}{x}) \tag{9}$$

$$\theta_2 = \arctan(\frac{a+s}{r-l_1+z}) \tag{10}$$

$$\theta_4 = \arctan(\frac{-s}{r-l_4}) \tag{11}$$

$$\theta_3 = 180^\circ - \theta_2 - \theta_4 \tag{12}$$

Here, (r, s) is the location of joint3 in s-r coordinates, which is also the intersection point of two circles with radius l_2 and l_3. Let $a = \sqrt{x^2+y^2}$ be the projection length of robot in X-Y plane. Then:

$$s^2 + (r-l_4)^2 = l_3^2 \tag{13}$$

$$(s+a)^2 + (r-l_1+z)^2 = l_2^2 \tag{14}$$

Solving these equations, we can find the intersection point and get r and s.

$$s = m * r + n \tag{15}$$

$$r = \frac{-p \pm \sqrt{p^2-4q}}{2} \tag{16}$$

$$m = \frac{-z+l_1-l_4}{a} \tag{17}$$

$$n = \frac{l_2^2 - l_3^2 + l_4^2 - (z-l_1)^2 - a^2}{2a} \tag{18}$$

$$p = \frac{2mn - 2l_4}{m^2+1} \tag{19}$$

$$q = \frac{n^2 + l_4^2 - l_3^2}{m^2+1} \tag{20}$$

4.2 Justification

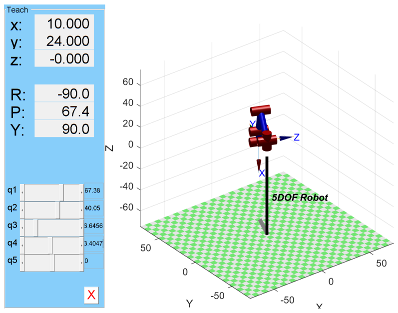


Figure 8: Justification test 1

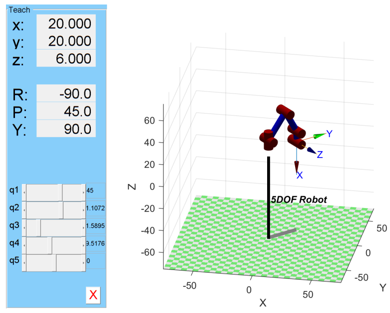


Figure 9: Justification Test 2

Test 1: $(x, y, z) = (10, 24, 0)$

Using MATLAB and the equations above, we get $r = 32.909$ or -8.6228. Our design, we set limits such that $\theta_2 \leq 90°$, where r is always positive. Hence, we neglect the negative value and get $\theta_1 = 67.3801°$, $\theta_2 = 49.9471°$, $\theta_3 = 116.6456°$, $\theta_4 = 13.4047°$.

These values can be used in the DH table to obtain the appropriate joint angles: $q_1 = 67.3801°$, $q_2 = 90 - \theta_2 = 40.05°$, $q_3 = -\theta_3 = -116.6456°$, $q_4 = -\theta_4 = 13.4047°$. Substituting the joint angles into the forward kinematics, we obtain the position of end-effector, which is $(10.000, 24.00, 0.000)$ as shown in Figure 8, proving that our derivation of inverse kinematics is correct.

Test 2: $(x, y, z) = (20, 20, 6)$

Similarly, we neglect the negative r value and calculate the angles $\theta_1 = 45°$, $\theta_2 = 38.8928°$, $\theta_3 = 111.5895°$, $\theta_4 = 29.5176°$. Next, calculate q values and substitute into forward kinematics and we arrive at the position of end-effector to be $(20, 20, 6)$ as shown in Figure 9.

5 Conclusion

This report has detailed the process of deriving and verifying the kinematic properties of a robotic arm designed to move chess pieces around on a board. It also estimates a set of joint parameters that will allow the robot to reach the full workspace.

The next step in the process of designing the arm is to calculate the dynamics. Of particular importance are the torques that each joint will experience. This will allow us to verify that the motors will be able to output enough torque to operate the arm.