Robotics Systems Assignment 1

Xinyu Shen, Jarod Tang, Zachary Lee

March 2024

Contents

1 Introduction **1**

2 Forward Kinematics **2**
2.1 Kinematic Definition 2
2.2 DH Table 3
2.3 Transformation Matrix 3
2.4 Validation 4

3 Determination of Link Lengths **5**
3.1 Workspace Estimation 5
3.2 Link Length Selection 9
3.3 Considerations 11

4 Inverse Kinematics **12**
4.1 Derivation 12
4.2 Validation 17

1 Introduction

Forward and inverse kinematics are fundamental concepts in robotics, governing the motion and control of robotic systems. Forward kinematics calculates the position and orientation of the robot's end effector (such as a gripper) based on the given joint parameters, such as angles and link-lengths, effectively describing how the motion of individual joints translates into the

motion of the entire robot using transformation matrices. In contrast, inverse kinematics solves the reverse problem, determining the necessary joint configurations to achieve a desired position and orientation of the end effector.

2 Forward Kinematics

2.1 Kinematic Definition

Calculating the transformation matrices that gives the relationships between each coordinate frames is an important process in designing the robotic arm. This can be done with the Denavit-Hartenberg (DH) convention, as it provides a systematic way to determine transformations between 2 coordinate frames. Suppose there are two coordinate frames, frame $\{i\}$ and the preceding frame $\{i-1\}$. There are 4 key parameters: a_{i-1}, α_{i-1}, d_i, θ_i.

- α_{i-1}: Rotation about x_{i-1} to align axes z_{i-1} and z_i
- a_{i-1}: Translation along x_{i-1} to align the origin of frames $\{i-1\}$ and $\{i\}$
- d_i: Translation along z_i to align the origin of frames $\{i-1\}$ and $\{i\}$
- θ_i: Rotation about z_i to align axes x_{i-1} and x_i

For this convention to hold:

- Assign the z_i axis of joint i along the axis of rotation or translation.
- Assign the x_{i-1} axis to intersect and be perpendicular to both z_i and z_{i-1}.
- Axis y_i can be found using the cross product between z_i and x_i. $y_i = z_i \times x_i$

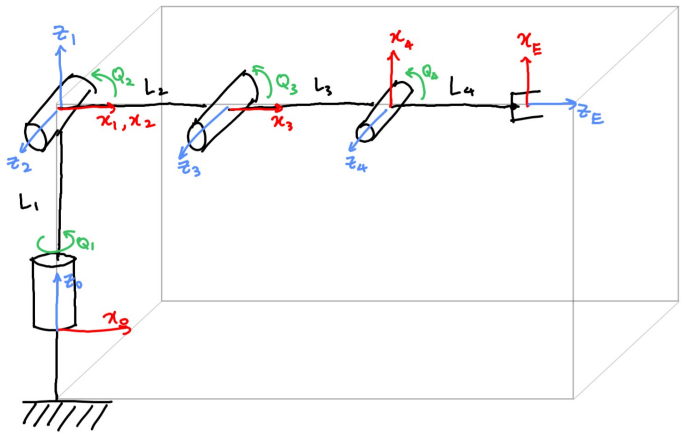


Figure 1: Robot arm configuration

2.2 DH Table

i	a_{i-1}	α_{i-1}	d_i	θ_i
1	0	0	L_1	Q_1
2	0	$\frac{\pi}{2}$	0	Q_2
3	L_2	0	0	Q_3
4	L_3	0	0	$Q_4 + \frac{\pi}{2}$
E	0	$\frac{\pi}{2}$	L_4	0

Table 1: DH Table

2.3 Transformation Matrix

From the DH table, transformation matrix between frames can be found with the following equation:

$$^{i-1}_{i}T = R_x(\alpha_{i-1})D_x(a_{i-1})R_z(\theta_i)D_z(d_i) \tag{1}$$

With equation 1, The following transformation matrices between subsequent frames can be easily obtained.

$$
{}^{0}_{1}T = \begin{bmatrix} \cos(Q_1) & -\sin(Q_1) & 0 & 0 \\ \sin(Q_1) & \cos(Q_1) & 0 & 0 \\ 0 & 0 & 1 & L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}
$$

$$
{}^{1}_{2}T = \begin{bmatrix} \cos(Q_2) & -\sin(Q_2) & 0 & 0 \\ 0 & 0 & -1 & 0 \\ \sin(Q_2) & \cos(Q_2) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}
$$

$$
{}^{2}_{3}T = \begin{bmatrix} \cos(Q_3) & -\sin(Q_3) & 0 & L_2 \\ \sin(Q_3) & \cos(Q_3) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}
$$

$$
{}^{3}_{4}T = \begin{bmatrix} \cos(Q_4 + \frac{\pi}{2}) & -\sin(Q_4 + \frac{\pi}{2}) & 0 & L_3 \\ \sin(Q_4 + \frac{\pi}{2}) & \cos(Q_4 + \frac{\pi}{2}) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}
$$

$$
{}^{4}_{E}T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & -L_4 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6}
$$

Finally, to obtain the forward kinematics of ${}^{0}_{E}T$, the transformation matrices can be chain multiplied:

$$
{}^{0}_{E}T = {}^{0}_{1}T\,{}^{1}_{2}T \cdots {}^{n-1}_{n}T\,{}^{n}_{E}T \tag{7}
$$

2.4 Validation

To test the correctness of forward kinematics obtained, a simple MATLAB 3D plot of the robotic arm can be used. Assuming all link lengths are 1 unit long, and all motor displacement angles are set to 0.

As seen from figure 2, the MATLAB model using default configurations results in the same robotic arm position as figure 1. Meaning that forward kinematics is done correctly.

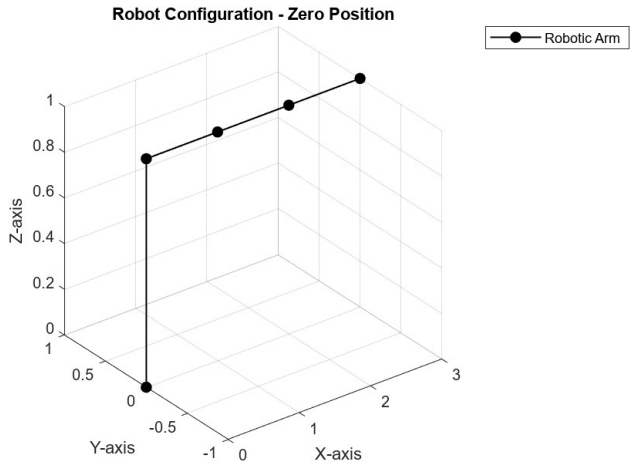


Figure 2: Initial configuration

3 Determination of Link Lengths

3.1 Workspace Estimation

In this section, the workspace of the robot is numerically estimated by plotting the reachable workspace of the robot end-effector. Firstly, the workspace requirements are as follows:

1. Each location on the chessboard must be reachable
2. The robot must be able to reach the height of each chess piece
3. The robot must be able to lift each piece to a height of at least twice the height of a piece to move the pieces

The workspace is visualised in two planes; XY and XZ. The XY workspace represents all the possible end-effector positions on the chessboard plane. The XZ workspace represents the vertical cross-section of the reachable workspace. To ensure the validity of the estimated workspace, the dimensions of the chessboard and the heights of the smallest and largest pieces have been measured. This ensures that the robot will be able to reach both the lowest and highest heights necessary to fulfil the workspace requirements listed above. The measurements are as follows:

- Chessboard dimensions: 36x36cm

- Chessboard height: 3.04mm
- Pawn height (smallest piece): 31.85mm
- King height (tallest piece): 56.3mm
- Twice the height of the tallest piece: 112.6mm

One important point of consideration is the position of the robot in the XY plane. Considering the robot in its initial configuration, located along the y-axis, if the robot was to only rotate 360° about joint 1, then the area of coverage would be a circle, where the radius would be the sum of l_2, l_3 and l_4. Assuming the chessboard is located only in the quadrant where "x" is positive and connected to the y-axis, the area of coverage would be a semi-circle. Therefore, for optimal coverage of the chessboard, it is advisable to position the robot at the base of the board, behind either the black or white side, in the center. This positioning ensures maximum reach and coverage. Refer to Figures 3 and 4 for visual representations.

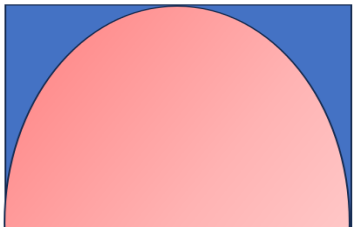


Figure 3: Robot at Chessboard Center

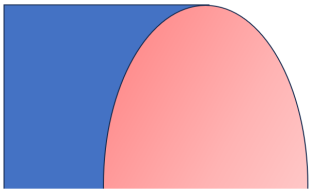


Figure 4: Robot at Chessboard Corner

To estimate the workspace, link lengths are initialised arbitrarily so that the full workspace can be visualised. The values are as follows:

- $l_1 = 150$mm
- $l_2 = 300$mm
- $l_3 = 300$mm
- $l_4 = 120$mm

The value of l_1 was chosen to $150mm$ to ensure that the robot in its initial configuration would clear all the pieces. The length $120mm$ for l_4 was chosen so that, in its initial configuration, the robot would be able to reach the smallest piece. The minimum required sum of l_2 and l_3 to reach the far corners of the board from the optimal position of the robot in its initial configuration is 40.25cm ($\sqrt{36^2 + 18^2}$). Therefore, the link lengths for l_2 and l_3 were both set to $300mm$ as a starting value.

In light of the use-case of the robot, which is one of a "pick and place" machine, a few mechanical and practical considerations must be kept in mind. These are as follows:

- Only inverse kinematic solutions belonging to the "elbow up" manifold are considered. An "elbow down" manifold is impractical for the required task as the robot would have no way of picking up and placing the chess pieces.
- At joint 1, there will be a motor and other electrical components to articulate rotation about that joint. Therefore, the base of the robot must be displaced a certain amount to account for this physical limitation, otherwise, it will begin to overlap with the chessboard. In the plots that follow, the base of the robot is displaced a placeholder amount (5cm) negatively along the x axis and positively along the z axis. This placeholder amount is added to the link lengths to account for this displacement.
- Scenarios where the elbow is completely straight is to be avoided and are not included in the final workspace estimation.
- Only solutions where $x \geq 0$ and $z \geq 0$ are to be considered. This is because it would be physically impossible for the robot to move through the platform on which the chessboard is. Additionally, the chessboard

is only on the right-hand side of the x-axis. Therefore, solutions which account for the left-hand side of the x-axis are redundant. Note that figures 5 and 6 will ignore these constraints so that the full workspace can be visualised.

- The most convenient orientation for the robot to pick and place chess pieces will be when the end-effector is pointed downwards and is orthogonal to the chessboard. Therefore, the sum of joints 2, 3 and 4 must be equal to $-90°$ ($\theta_2 + \theta_3 + \theta_4 = -\frac{\pi}{2}$).

When plotting the full XY workspace, the joint constraints to impose an "elbow up" manifold are as follows:

- Q_1: $-\frac{\pi}{2} < \theta_1 < \frac{\pi}{2}$
- Q_2: $0 < \theta_2 < \pi$
- Q_3: $-\pi < \theta_3 < 0$
- Q_4: $-\left(\frac{\pi}{2} + \theta_3 + \theta_2\right)$

With these constraints, Figure 5 is produced and is shown below. Note that since both l_1 and l_4 are perpendicular to the chessboard, only l_2 and l_3 will impact the reachability in the XY plane. This is reflected in both Figures 5 and 7. The chessboard has been superimposed onto this plot.

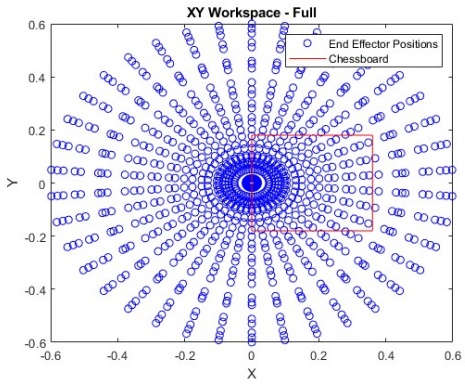


Figure 5: No Constraints XY Workspace

In the construction of the full XZ workspace, the joint limits for Q_2, Q_3 and Q_4 remain the same. Q_1 is set to 0 as only the vertical cross-section is desired. The joint constraints are as follows:

- Q_1: $\theta_1 = 0$
- Q_2: $0 < \theta_2 < \pi$
- Q_3: $-\pi < \theta_3 < 0$
- Q_4: $-\left(\frac{\pi}{2} + \theta_3 + \theta_2\right)$

Figure 6 shown below is the full XZ workspace. The chessboard length and height, as well as the smallest and tallest chess piece heights, have been imposed onto the plot.

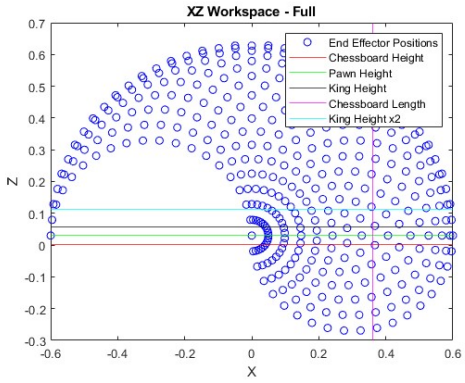


Figure 6: No Constraints XZ Workspace

The practical reachable workspace of both the XY and XZ workspaces are depicted below in figures 7 and 8. Based on the plots produced, the initial choice of link lengths is not suitable to produce the required workspace. Based on Figure 6, both l_1 and l_4 appear to be sufficiently long to reach the tallest and smallest required heights, whilst not being too short. Figure 7 shows that the chessboard can be reached by the arm however both l_{12} and l_3 appear to be too long, as they can extend beyond the chessboard. Moreover, they extend roughly 19cm beyond the chessboard, which is too excessive for the required workspace.

3.2 Link Length Selection

As discussed in Section 3.1, link lengths l_2 and l_3 are not suitable. Therefore, the following link length changes were made.

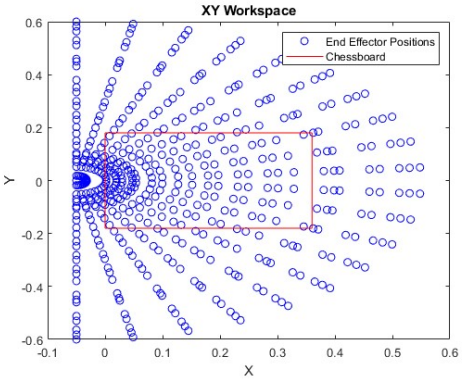


Figure 7: Initial XY Workspace

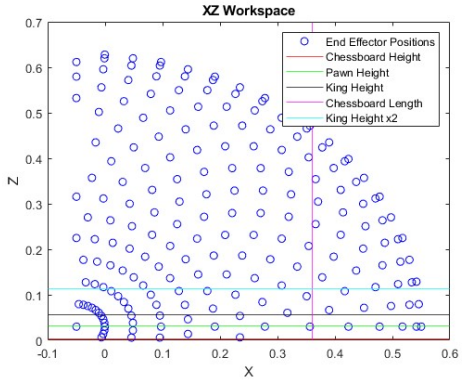


Figure 8: Initial XZ Workspace

- $l_2 = 0.25$m
- $l_3 = 0.2$m

The XY and XZ reachable workspace after the link length changes are depicted below. In Figure 9, the end-effector can reach all positions on the chessboard. Moreover, the combined length of links 2 and 3 is not excessive. Whilst the combination of l_2 and l_3 do extend 4cm beyond the board, this is more acceptable than the previous combination of link lengths extending 24cm beyond the board. Therefore, the lengths set for l_2 and l_3 are suitable for this workspace.

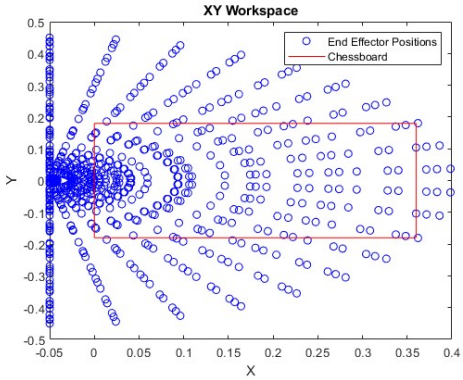


Figure 9: Final XY Workspace

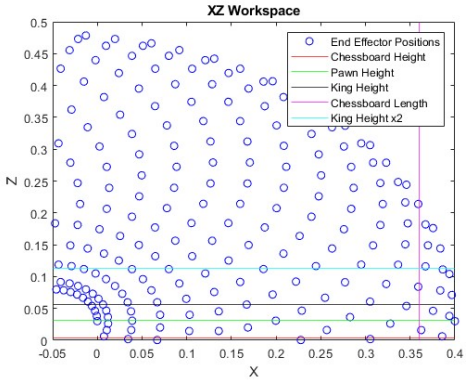


Figure 10: Final XZ Workspace

3.3 Considerations

Links lengths should neither be too long or too short for the following reasons:

1. Reachability: If the links are too short, then the robot will have a reach not suitable for its given workspace. Conversely, links that are too long may cause inefficiencies in motion planning and cause the motors to work harder.

2. Mechanical Stability: Links that are too short may cause structural instability and a reduction in load-bearing ability. Links that are too long may introduce mechanical flexibility, compliance, and vibrations, reducing accuracy in positioning.

4 Inverse Kinematics

Inverse Kinematics is the process of computing the joint angles required to achieve a given end-effector pose (see Equation 8). For 2 DOF planar manipulators, the derivations are trivial, whereby the equations for Q_1 and Q_2 can be obtained by solving the equations for x and y (from Section 2) simultaneously. However, as the degrees of freedom increase, so does the complexity of computing the joint angle equations. Using knowledge of the geometric configuration of the robotic arm, several simplifying assumptions can be made to ultimately obtain the full set of Inverse Kinematics equations.

$$\mathbf{Q} = IK(\mathbf{X}) \tag{8}$$

4.1 Derivation

The process of the subsequent derivation of the Inverse Kinematics is as follows:

1. Identify that the arm can be decomposed into two sections: one controlling the end-effector's gross movement, and one its gross orientation,
2. Use this simplifying assumption to derive the equations for Q_1, Q_2 and Q_3.
3. Using these values, obtain an equation for the end-effector orientation calculated with respect to Q_4, and
4. Compare this equation with the given end-effector orientation from the pose to numerically calculate Q_4.

Firstly, as can be seen in Section 3.1, the 4 DOF configuration of the robotic arm commands a full coverage of the x, y and z axes of the chessboard. This is achieved chiefly from the 1R Axial – 2R Planar configuration of the base 3 joints, i.e. a radius (as given by Q_2 and Q_3) and angle (as given by Q_1) which defines the complete 3D space in cylindrical coordinates. Thus, it can be identified that the gross movement of the end-effector is controlled using these joints. Furthermore, it can be said that the point that intersects the joint 4 frame is not affected by Q_4, and so can be used to determine the 3 base joint angles required to achieve a given end-effector position with

respect to the 3rd joint (this assumption will be validated later). This point is subsequently referred to as the 'wrist position' (W) with coordinates given in Equation 9 with respect to the inertial frame.

$$^0P_{0W} = [x_w, y_w, z_w]^T \tag{9}$$

Now, the 3 base joints can be treated as a 3 DOF arm with an end-effector at the wrist and the Inverse Kinematics calculated as such. First, joints 2 and 3 are viewed as a 2R planar arm in the frame attached to joint 1 as in Figure 11, from which Q_3 can be calculated using the Cosine Rule (see Equation 12). Similarly, Q_2 can be calculated using the Sine Rule (see Equation 17). Note that the wrist position is defined as (x'_w, z_w) where x'_w is the distance of the wrist from the origin of frame 1, which relates to x_w and y_w (the x and y coordinates of the wrist given in the inertial frame) through Equation 18.

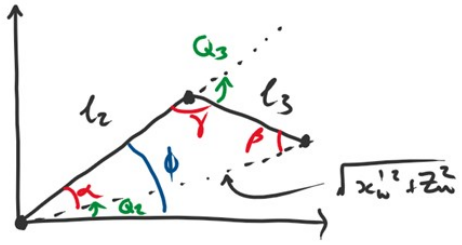


Figure 11: Analysis of Joint 2 and 3 Using Cosine Rule

$$(x'_w)^2 + z_w^2 = l_2^2 + l_3^2 - 2l_2l_3\cos\gamma \tag{10}$$

where

$$\cos\gamma = \cos(\pi + Q_3) = -\cos Q_3 \tag{11}$$

$$Q_3 = \pm\cos^{-1}\left(\frac{(x'_w)^2 + z_w^2 - (l_2^2 + l_3^2)}{2l_2l_3}\right) \tag{12}$$

Therefore,

$$Q_3^1 = -\cos^{-1}\left(\frac{(x'_w)^2 + z_w^2 - (l_2^2 + l_3^2)}{2l_2l_3}\right) \tag{13}$$

Derivation of Q_3 Using Cosine Rule

$$\frac{l_3}{\sin\alpha} = \frac{\sqrt{(x'_w)^2 + z_w^2}}{\sin\gamma} \tag{14}$$

where

$$\sin\gamma = \sin(\pi - Q_3) = \sin Q_3 \tag{15}$$

$$Q_2 = \phi - \alpha \tag{16}$$

$$Q_2^1 = \tan\left(\frac{z_w}{x'_w}\right) - \sin^{-1}\left(\frac{l_3 \sin Q_3^1}{\sqrt{(x'_w)^2 + z_w^2}}\right) \tag{17}$$

Derivation of Q_2 Using Sine Rule

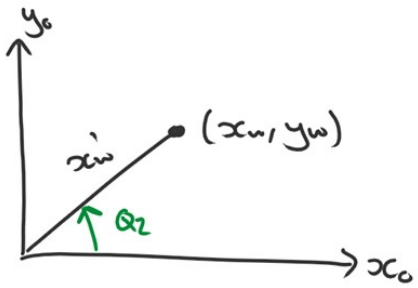


Figure 12: Derivation of Q1 Using End-Effector Position

$$x'_w = \sqrt{x_w^2 + y_w^2} \tag{18}$$

Wrist Position Equation

Note that in Equations 12 and 17, there are two solutions to Q_2 and Q_3 corresponding to either the 'elbow up' or 'elbow down' configuration – i.e. whether Q_3 is positive or negative, and subsequently whether Q_2 is positive or negative based on Q_3. In this case, the elbow up configuration (denoted using the trailing superscript '1') is taken since the chess pieces will always be grasped from above.

Secondly, Q_1 can be computed directly based on the wrist position as seen in Figure 12, to be Equation 18.

$$Q_1 = \tan^{-1}\left(\frac{y_w}{x_w}\right) \tag{19}$$

Derivation of Q_1

Thirdly, now that Q_1 to Q_3 are known with respect to the wrist position, they must be calculated with respect to the given end-effector position. This can be done using another simplifying assumption, now related to the final joint. As mentioned before, the movement of joint 4 has no effect on the wrist position – which implies that in the frame of the end-effector, the wrist position remains constant. This can be seen in Figure 13, where the point w is always at a distance of l_4 in the negative z axis of the end-effector frame. Thus, the wrist position can be calculated using a vector sum of the known end-effector position, orientation, and link length as in Equation 20.

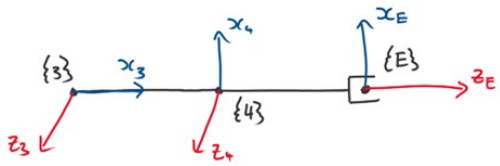


Figure 13: End-Effector Frame

$$\begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} = {}^0P_{0E} + {}^E_0R \begin{bmatrix} 0 \\ 0 \\ -l_4 \end{bmatrix} \tag{20}$$

Wrist Position Calculation

Substituting Equation 20 into Equations 12, 17, and 4.1 yields Q_1 to Q_3 as a function of the end-effector pose. Note that the representation of the end-effector orientation given in ${}^0_ER$ as an input parameter (denoted as 'pitch') is achieved using Euler Angles. That is, the orientation with respect to the inertial frame is given as a sequence of rotation matrices firstly in the z axis, then in y, and finally in x. This is simplified by recognizing that the configuration of joint 4 restricts the rotational degrees of freedom to purely a rotation about the y axis of frame 1; since the overall orientation of the end-effector about the z axis is determined by joint 1, and there is no joint which affects rotation in the x axis (meaning this rotation matrix becomes the identity matrix).

The final joint angle Q_4 can be solved by computing the transformation matrix from joint 3 to the end effector (see Figure 13) using the given transformation matrix from the inertial frame to the end-effector, ${}^{0}_{E}T$, and the calculated transformation matrix from joint 3 to the inertial frame as given in Equation 4.1 ${}^{3}_{0}T$. Note that the latter matrix can be obtained as a function of Q_1 to Q_3 using the matrix transpose given in Equation 22.

$$ {}^{3}_{E}T = {}^{3}_{0}T \cdot {}^{0}_{E}T \tag{21} $$

$$ {}^{3}_{0}T = \left[\begin{array}{ccc|c} & {}^{0}_{3}R^{T} & & {}^{0}_{3}R^{T}\,{}^{0}P_{03} \\ \hline 0 & 0 & 0 & 1 \end{array}\right] \tag{22} $$

Finally, this set of calculations yields ${}^{3}_{E}T$ (and specifically ${}^{3}_{E}R$), a numerical matrix as a function of the known end-effector pose and calculated joint angles Q_1 to Q_3 (also as functions of the known end-effector pose). Then, using the DH table to obtain the transformation matrix ${}^{3}_{E}T$ as given in Equation 21, yields a rotation matrix from the frame at joint 3 to the end-effector frame as a function of Q_4.

$$ {}^{3}_{E}T = {}^{3}_{4}T\,{}^{4}_{E}T, \text{ from Table 1} \tag{23} $$

Thus, the rotation matrix ${}^{3}_{E}R$ as calculated using Q_1 to Q_3 and the given end-effector orientation can be compared with that calculated from the DH table to solve for Q_4 as in Equation 24.

$$ Q_4 = \tan^{-1}\left(\frac{-\,{}^{3}_{E}T_{1,3}}{{}^{3}_{E}T_{1,1}}\right) \tag{24} $$

Therefore, the final set of Inverse Kinematic equations computing the joint angles required to achieve a given end-effector pose is given in Equation 25.

$$ \mathbf{Q} = \begin{bmatrix} \tan^{-1}\left(\frac{y_w}{x_w}\right) \\ \tan\left(\frac{z_w}{x'_w}\right) - \sin^{-1}\left(\frac{l_3 \sin Q_3^1}{\sqrt{(x'_w)^2+z_w^2}}\right) \\ -\cos^{-1}\left(\frac{(x'_w)^2+z_w^2-(l_2^2+l_3^2)}{2l_2l_3}\right) \\ \tan^{-1}\left(\frac{-\,{}^{3}_{E}T_{1,3}}{{}^{3}_{E}T_{1,1}}\right) \end{bmatrix} \tag{25} $$

Where, $\begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix}$ and x'_w can be found with equations 20 and 18 respectively.

4.2 Validation

To validate equation 25, a MATLAB simulation can be used, with a specified end-effector configuration.

For testing purposes, set end-effector coordinates to $\begin{bmatrix} 0.3 \\ 0 \\ 0.3 \end{bmatrix}$, and pitch down by $\frac{\pi}{2}$. Length units are in meters and angles are in radians. This gives:

$$ {}^{0}_{E}T = \left[\begin{array}{ccc|c} 1 & 0 & 0 & 0.2 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0.3 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] \tag{26} $$

Figure 14 shows that the inverse kinematics is valid.

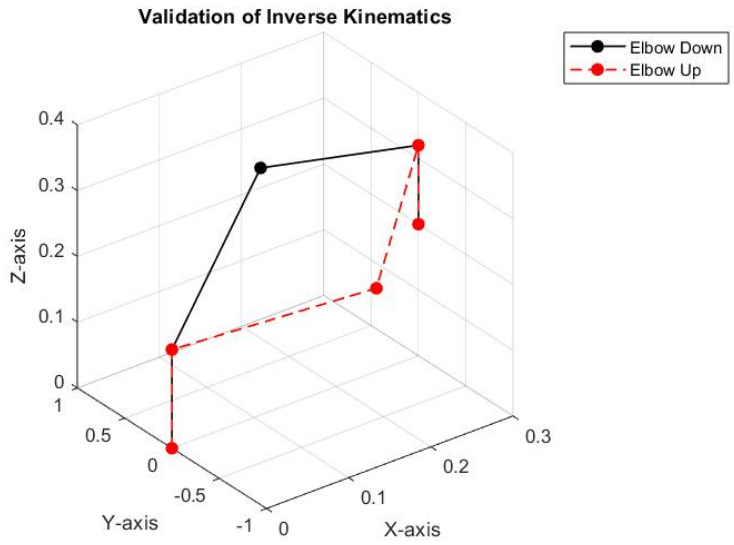


Figure 14: Robotic arm end effector following specification