Lab 2: Inverse Kinematics of the Lynx

MEAM 520, University of Pennsylvania

October 9th, 2020

Venkata Gurrala, Sheil Sarda

Table of Contents

Introduction 2
Method 2
  Kinematic decoupling 2
  Determining rotation matrices 3
  Understanding the wrist 5
  Considering DOF limitations 6
Results 8
  Target 1: Infeasible orientation 8
  Target 2: Feasible orientation 9
  Target 3: Infeasible orientation 9
  Target 4: Infeasible orientation 10
Analysis 10

Introduction

The objective of this lab was to determine the inverse kinematic equations for the robot from the previous lab. These equations were used to relate the Transformation matrix to the joint positions and orientations using python code. The end goal of the lab included being able to output a matrix of all the solutions possible given a transformation matrix.

Method

Kinematic decoupling

To make kinematic calculations easier, the robot was split into two sections. One section comprised of the joints from the base to the wrist. The 2nd section was composed of the wrist, rotating wrist, and end effector. This is called kinematic decoupling and it was used find closed form inverse kinematic solutions. In the prelab, the objective was to determine the closed form equations from the world frame (frame 0) to the wrist joint as a function of the joint variables. These were found with geometric relations using the symbolic diagram in figure 1.

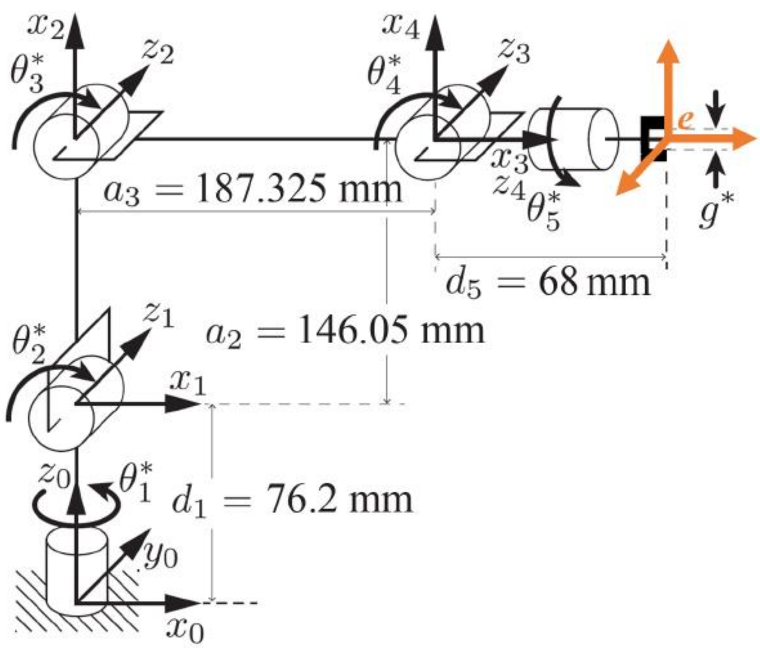


Figure 1. 3D symbolic representation of robot used

$$\theta_1 = \text{atan}\,(\frac{y}{x})$$

<u>Derivation of θ_3:</u> Consider the triangle formed by the (1) origin of frame 1, (2) center of the wrist and the (3) origin of frame 2. Applying law of cosines on the angle associated with frame 2 yields:

$$2a_2a_3 * cos(\theta_3 + \pi/2) + a_2^2 + a_3^2 = x_w^2 + y_w^2 + (z_w - d_1)^2$$

Similarly, if we consider the triangle created by reflecting the above described triangle about the line connecting frame 1 and the center of the wrist, we arrive at an elbow down solution using law of cosines

$$2a_2a_3 * cos(-\theta_3 - \pi/2) \ + a_2^2 + a_3^2 = x_w^2 + y_w^2 + \ (z_w - d_1)^2$$

Simplifying the above expressions yields two cases for theta3:

$$\theta_3 = -\frac{\pi}{2} - \cos^{-1}(\frac{x_w^2 + y_w^2 + (x_w - d_1)^2 - a_2^2 - a_3^2}{2a_2a_3})$$

$$\theta_3 = -\frac{\pi}{2} + \cos^{-1}(\frac{x_w^2 + y_w^2 + (x_w - d_1)^2 - a_2^2 - a_3^2}{2a_2a_3})$$

<u>Derivation of θ_2:</u> Consider the triangle with the vertices (1) center of frame 1, (2) wrist center and (3) the projection of the wrist center onto the x-y plane translated in the z_0 direction by d_1. The leg of this triangle lying parallel to the x-y plane has length $\sqrt{x_w^2 + y_w^2}$. The leg of the triangle perpendicular to the x-y plane has length $z_w - d_1$. The angle of the frame 1 vertex of the triangle can be calculated as $arctan2(z_w - d_1, \sqrt{x_w^2 + y_w^2})$.

Consider adding that angle to the angle $\theta_2 - \pi/2$. This sum represents the angle between the x-y plane translated up by d_1 and the link a_2 extending from frame 1. Call this angle alpha (α).

Consider the right triangle whose hypotenuse is the line segment bounded by the origin of frame 1 and the wrist center. The third point of the triangle is chosen to form the right triangle. The side opposite alpha can be written as $a_3\sin{(-\pi/2 - \theta_3)}$. The side adjacent to alpha can be represented by $a_2 + a_3\cos{(-\pi/2 - \theta_3)}$.

Using the above geometrical intuition, we know that the tangent of the above fraction represents alpha. Once we have an expression for alpha, decompose the expression to solve for θ_2:

$$\theta_2 = \frac{\pi}{2} - atan2\left(\frac{z_w - d_1}{\sqrt{x_w^2 + y_w^2}}\right) + atan2(\frac{a_3 \sin(-\pi/2 - \theta_3)}{a_2 + a_3 \cos(-\pi/2 - \theta_3)})$$

Determining rotation matrices

Once angles θ_1, θ_2 and θ_3 are found, rotation matrices were formed using the DH convention. The matrices can be used to back solve for the angles θ_4 and θ_5. The DH table obtained is shown in table 1.

Link number	a_i	α_i	d_i	θ_i
1	0	$-\pi/2$	76.2	${\theta_1}^*$
2	146.05	0	0	$\theta_2 - pi/2^*$
3	187.325	0	0	$\theta_3 + pi/2^*$
4	0	$-\pi/2$	0	$\theta_4 - pi/2^*$
e	0	0	68	${\theta_5}^*$

Table 1. DH table used to find rotation matrices

The rotation matrices for each pair of joints is shown below:

$$R_1^0 = \begin{vmatrix} \cos(\theta_1) & 0 & -\sin(\theta_1) \\ \sin(\theta_1) & 0 & \cos(\theta_1) \\ 0 & -1 & 0 \end{vmatrix}$$

$$R_2^1 = \begin{vmatrix} \cos(\theta_2 - \frac{\pi}{2}) & -\sin(\theta_2 - \frac{\pi}{2}) & 0 \\ \sin(\theta_2 - \frac{\pi}{2}) & \cos(\theta_2 - \frac{\pi}{2}) & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

$$R_3^2 = \begin{vmatrix} \cos\left(\theta_3 + \frac{\pi}{2}\right) & -\sin\left(\theta_3 + \frac{\pi}{2}\right) & 0 \\ \sin\left(\theta_3 + \frac{\pi}{2}\right) & \cos\left(\theta_3 + \frac{\pi}{2}\right) & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

$$R_4^3 = \begin{vmatrix} \cos(\theta_4 - \frac{\pi}{2}) & 0 & -\sin(\theta_4 - \frac{\pi}{2}) \\ \sin(\theta_4 - \frac{\pi}{2}) & 0 & \cos(\theta_4 - \frac{\pi}{2}) \\ 0 & -1 & 0 \end{vmatrix}$$

$$R_e^4 = \begin{vmatrix} \cos(\theta_5) & -\sin(\theta_5) & 0 \\ \sin(\theta_5) & \cos(\theta_5) & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

After determining all the individual matrix rotation matrices, the following multiplication was done to find $R_3^0$with the calculated θ_2 and θ_3 values.

$$\boldsymbol{R_3^0 = R_1^0 R_2^1 R_3^2}$$

The R_3^0 is then used to find R_e^3 using the following formulation:

$$\boldsymbol{R_e^3 = {R_3^0}^T R_e^0}$$

The numerical output to R_e^3 is then compared to its general form shown below.

$$R_e^3 = \begin{vmatrix} \cos\left(\theta_4 - \frac{\pi}{2}\right)\cos(\theta_5) & -\cos\left(\theta_4 - \frac{\pi}{2}\right)\sin(\theta_5) & -\sin\left(\theta_4 - \frac{\pi}{2}\right) \\ \sin\left(\theta_4 - \frac{\pi}{2}\right)\cos(\theta_5) & -\sin\left(\theta_4 - \frac{\pi}{2}\right)\sin(\theta_5) & \cos\left(\theta_4 - \frac{\pi}{2}\right) \\ -\sin(\theta_5) & -\cos(\theta_5) & 0 \end{vmatrix}$$

Where the numbered matrix can be represented as the following:

$$R_e^3 = \begin{vmatrix} V_1 & V_2 & V_3 \\ V_4 & V_5 & V_6 \\ V_7 & V_8 & V_9 \end{vmatrix}$$

When the two matrices are made equal to each other, the following angles, θ_4 and θ_5 can be found in the following ways:

$$\theta_4 = \arctan\left(\frac{-V_3}{V_6}\right) + \frac{\pi}{2}$$

$$\theta_5 = \arctan\left(\frac{V_7}{V_8}\right)$$

Understanding the wrist

Since there are several joints that exist to make the end effector reach a point in space, there can often be multiple solutions for angles that can satisfy the given position. Figure 2 shows an example of a point in space (x,y,z) in which joints 2 and 3 can move to satisfy.

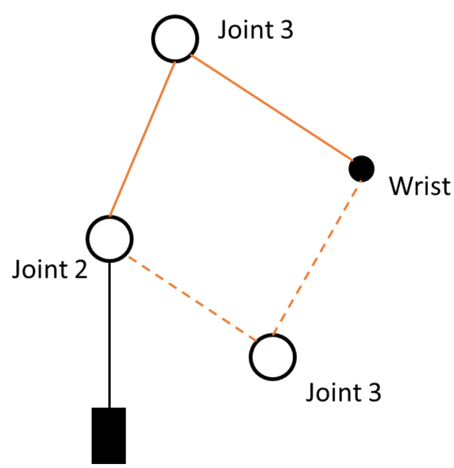


Figure 2. (x,y,z) point in space satisfied with two joint orientations

In order to simplify this, an equation was determined that would relate the end effector position to the wrist's position shown below.

$$X_{end\ effector} = X_{Wrist\ position} + 68\begin{vmatrix} z_e * x_0 \\ z_e * y_0 \\ z_e * z_0 \end{vmatrix}$$

The matrix used in this equation is from the third column of the given transformation matrix and it represents the rotation of the end effectors frame with respect to the 0th frame. The matrix is multiplied by 68 since it is the length of the link connecting the wrist and end effector. Both X variables in the equation represent the $(x_{0,}, y_{0,}, z_{0,})$ coordinates of end effector and wrist position. While using figure 2 while considering joints 4-e, it can be noticed that the end effector's position is fixed with respect to the wrists position. This makes it convenient to use the position and orientation of the end effector to find the wrists position.

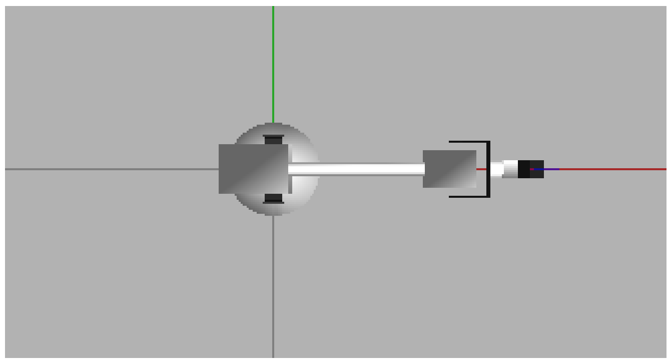


Figure 3. Top view of robot

Considering DOF limitations

While this robot's reachable workspace is very large and can have infinite orientations to reach many points in the reachable workspace, it has 5 degrees of freedom whereas rigid bodies normally have 6 degrees of freedom. This makes certain orientations of the end effector impossible to reach with the current design. From a top view, the robot can be simply represented as a straight line at every single input of a given q and this becomes an important observation as we attempt to represent infeasible end effector orientations. Figure 4 shows this top view of the robot.

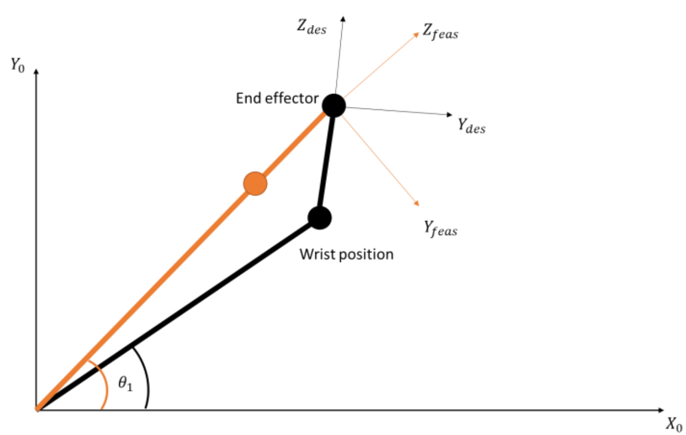


Figure 4. Top view representation of an infeasible end effector orientation

The figure represents a hypothetical situation where the given transformation matrix will suggest that the end effector is not in line with the wrist position. Since our robot does not have the ability to move in his axis, this is not possible. The orange line represents the new feasible position of the robot from the top view. This is the position where the end effector is able to reach the target position while keeping itself in line with the wrist.

In the situation presented, it would be necessary for the robot to recognize that the end effector will not be able to follow a given orientation. In figure 4, it can be noticed that the wrist's position is not on the same line as the position of the end effector. In a given end effector position and orientation, both the end effector and the wrist should have the same angle with respect to the x and y axis. So a mathematical check can be developed to ensure that the robot recognizes when an orientation is not feasible by calculating θ_1 from both the wrist position and the end effector positions since their coordinates are known using the last column of the transformation matrix. If the two angles measured using the arctan of the x and y coordinates of the wrist and end effectors are not the same, it is considered to have an infeasible orientation.

When the orientation is determined to be infeasible, the isPos=0 and the robot will determine the closest orientation that the end effector it can reach. The z-axis is prioritized since it represents the approach orientation of the end effector. Since the robot is assumed to reach the target end effector position, there is no possible movement in the x or y direction that is possible to satisfy the original infeasible orientation. However, with the assistance of angles $\theta_2 - \theta_4$, the end effector's orientation in the z direction can be matched.

In figure 5, the Z_{des} is at an angle to Z_e whose angle will be eliminated in the feasible orientation space. To do this, the projection of Z_{des} in the normal direction to Z_e is found. To do this, a vector is found which is orthogonal to the Z_e as shown in the figure below with a highlighted arrow. This normal is the Z_3 vector taken from the 3rd column of the R03 matrix. The matrix is found to be the following:

$$\vec{n} = \begin{vmatrix} -\sin(\theta_1) \\ \cos(\theta_1) \\ 0 \end{vmatrix}$$

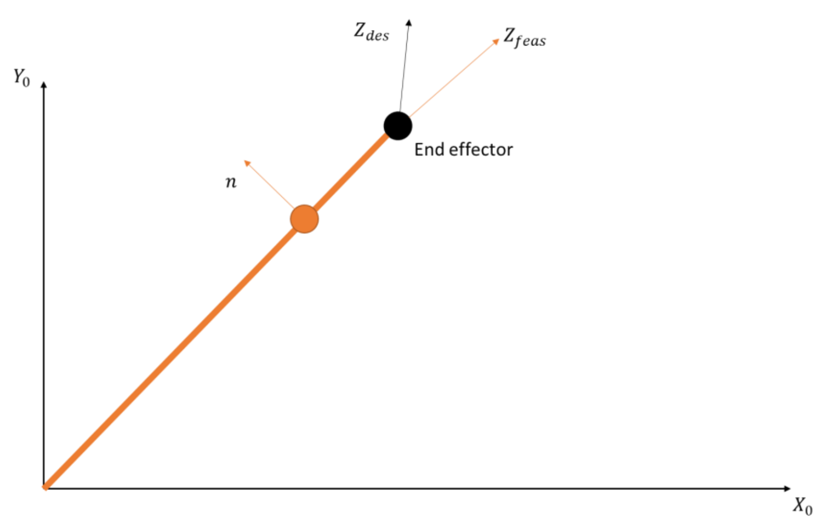


Figure 5. Top view representation used to find feasible orientation

The projection of the Z_{des} vector in the vector n can be represented by the following:

$$proj_n Z_{des} = \frac{Z_{des} \cdot n}{\|n\|^2} n$$

Once this projection was found, the following calculation was done to remove the yaw component of the orientation:

$$Z_{feas} = Z_{des} - proj_n Z_{des}$$

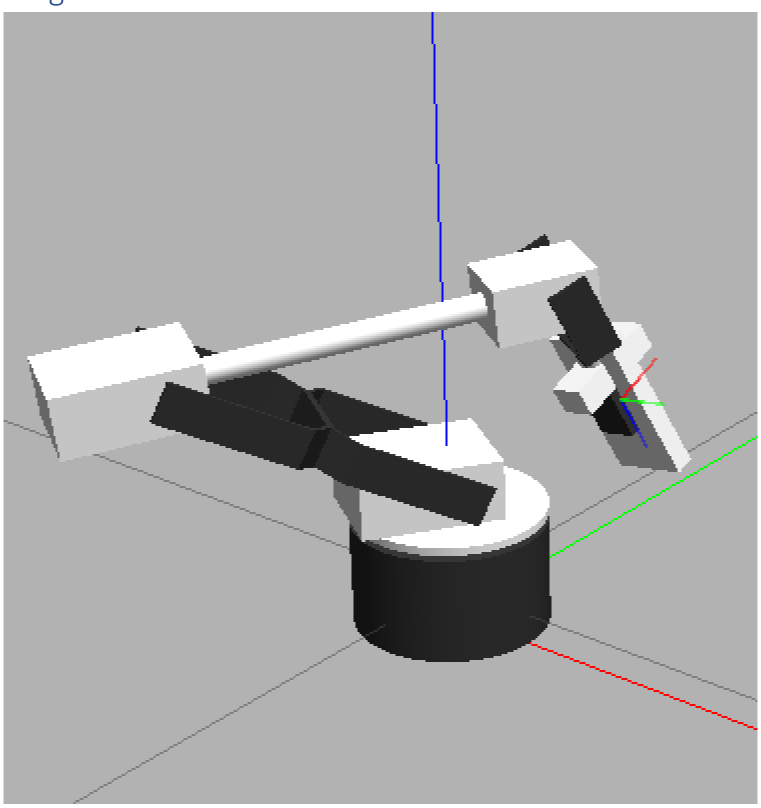
Once the Z_{feas} was found, the next orientation that was modified was the Y. This was done in a similar method to finding the feasible Z. Since the Z_{feas} is orthogonal to the Y_{feas}, it can be the normal vector on which to project the Y_{des}. The projection of the Y_{des} onto Z_{des} can be subtracted from the Y_{des} to lead to the Y_{feas}:

$$proj_{Z_{des}} Y_{des} = \frac{Y_{des} \cdot Z_{des}}{\|Z_{des}\|^2} Z_{des}$$

$$Y_{feas} = Y_{des} - proj_{Z_{des}} Y_{des}$$

Results

Target 1: Infeasible orientation



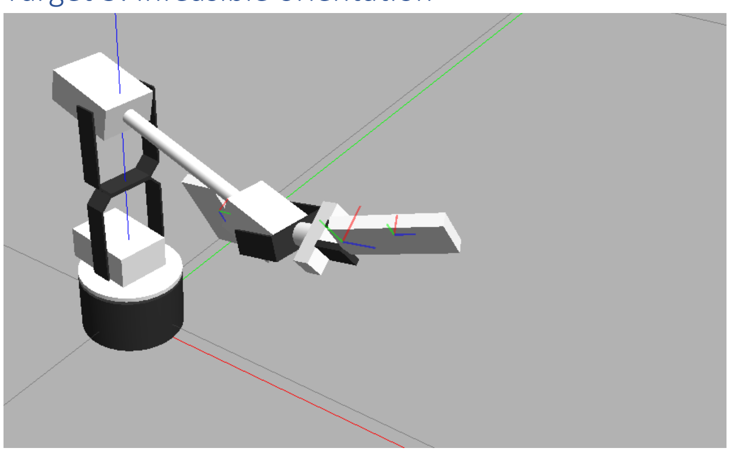
```
('The joint values are', [1.0, -1.1, 1.0, 1.201, -0.499, 7.416])

('The joint velocities are', [-0.0, -0.0, -0.0, 0.0, 0.0, -22.584])
Simulation T0e =
[[   0.02      0.969     0.245    47.001]
 [   0.917    -0.115     0.381    73.198]
 [   0.397     0.217    -0.892   100.504]
 [   0.        0.        0.         1.   ]]
Target T0e =
[[   0.019     0.969     0.245    47.046]
 [   0.917    -0.115     0.382    73.269]
 [   0.398     0.217    -0.891   100.547]
 [   0.        0.        0.         1.   ]]
isPos =
True
q =
[[ 0.99999  -1.10024   1.00012   1.19983  -0.499475  0.        ]]
```

Target 2: Feasible orientation

Target 2 was not met by the robot's end effector

Target 3: Infeasible orientation



```
('The joint values are', [-0.041, -0.009, 0.055, -0.118, -0.229, -0.71])

('The joint velocities are', [-0.198, -0.288, 1.482, -5.857, -8.295, -30.71])
Simulation T0e =
[[  -0.057   -0.05     0.997  253.594]
 [   0.211   -0.977   -0.037   -9.494]
 [   0.976    0.208    0.066  218.967]
 [   0.       0.       0.        1.   ]]
Target T0e =
[[  -0.341   -0.107    0.934  282.96 ]
 [   0.784   -0.58     0.219  -48.302]
 [   0.518    0.807    0.282  235.071]
 [   0.       0.       0.        1.   ]]
isPos =
True
q =
[[-0.16907  -0.03589   0.22262  -0.495755 -0.931857  0.       ]
 [-0.169073  2.06502  -3.364216 -0.495755 -0.931857  0.       ]]
meam520@meam520VM:~/meam520_ws/src/python_code/Workspace/Lab2$
```

Target 4: Infeasible orientation

The target was not met by the robot due to joint limit restrictions

Analysis

The results from the targets show that 2 of the 4 targets were physically met by the robot. This was due to problems with joint limits. This could have been fixed by incorporating a larger q array with solutions which could be narrowed down by the end of the algorithm.

Generally, there were some positions in which the robot had generally followed the point in but the simulation showed a slight offset in certain positions. This prevent it to reach all positions accurately. These inaccuracies could have been the result of floating point inaccuracies where some of the numbers after the decimal could have been truncated. Additionally, if the simulation models friction, torque limits and the effects of gravity, the effects could add to create a slight variation. To mitigate some of these issues, the joint velocity can be lowered so that there is a smaller effect on momentum. With some testing, the effects of momentum from gravity can be tested. A factor can be added along with the final q values that will cancel the small effects of gravity. This factor may only need to be added with certain joint travel or orientations.

- The algorithm starts out by indexing the last column of the T0e to find desired wrist position (line 45)
- The wrist position is calculated (line 49)
- Theta 1 is calculated in line 56
- The feasibility check occurs in line 64 by comparing theta 1 values from the end effector and wrist positions
- Theta 3 solutions are calculated using the two formulas mentioned from **lines 74-100**
- The corresponding theta 2 values are found in **lines 99-104**
- Lines 111-149 have the rotation matrix forms for each frame
- The rotation matrix is taken from the given T0e in lines 151-155
- An if statement is used to check whether the isPos is true in line 159
- If isPos is true, Theta 4 and 5 is calculated as mentioned before (lines 160-185)
- If the isPos is false, the else branch is used to determine the feasible position (lines 188-260)
- Once the feasible position is found, the new T0e is returned (line 262)
- If isPos was true, all of the solutions (q) will be filtered with the joint limits specified in lines 267-274)