Lab 1: Kinematic Characterization of the Lynx (MATLAB)

UNIVERSITY OF PENNSYLVANIA

WESLEY YEE, SHAUN FEDRICK
MEAM 520
SEPTEMBER 23, 2020

1 Methods

For our experimental setup, we used MATLAB to code our forward kinematics calculations, which communicated to ROS running via Gazebo on a local VM running Ubuntu. A 6-DOF robot arm manipulator was programmed to run in Gazebo. Once the code was run in MATLAB on the host computer, the commands for operating the robot were then sent to Gazebo which then actuated the robot to the desired joint variable positions.

The ROS robot was modeled off of an actual physical robot located at Penn, but was not accessible due to COVID-19.
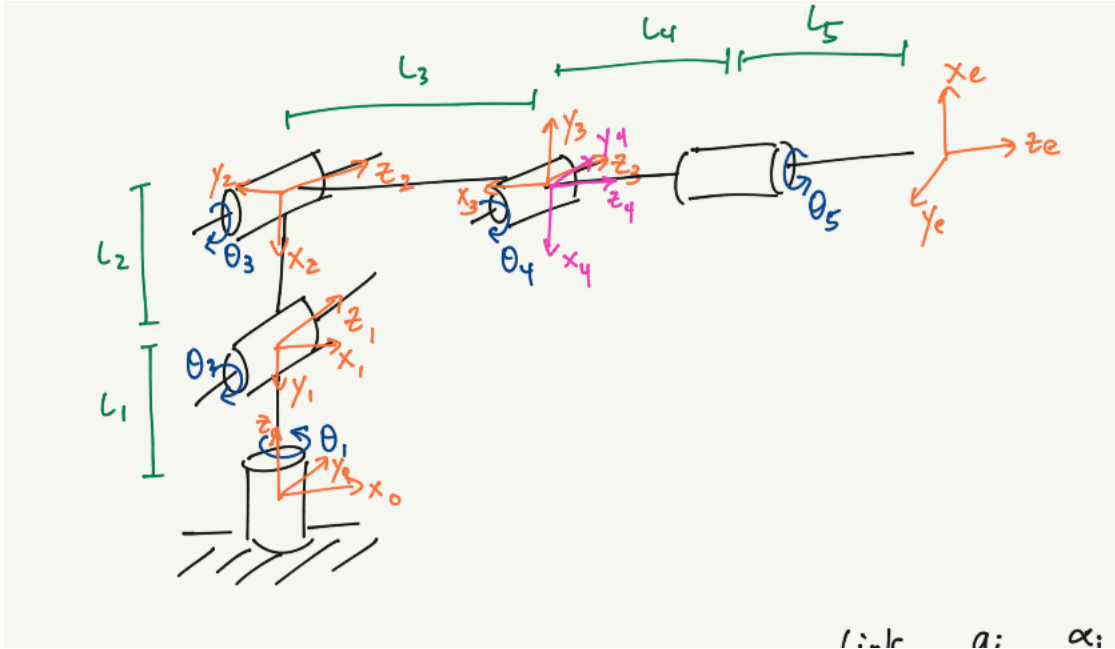

Figure 1: Revised symbolic representation

1.

2. FIX THESE BELOW

$$T_1^0 = \begin{bmatrix} 0 & 0 & 1 & 0mm \\ -1 & 0 & 0 & 0mm \\ 0 & -1 & 0 & 76.2mm \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

$$T_2^1 = \begin{bmatrix} 0 & -1 & 1 & 0mm \\ 1 & 0 & 0 & 146.05mm \\ 0 & 0 & 1 & 0mm \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$$T_3^2 = \begin{bmatrix} -\frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} & 0 & -132.4588mm \\ \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} & 0 & 132.4588mm \\ 0 & 0 & 1 & 0mm \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$$T_4^3 = \begin{bmatrix} 0 & 0 & -1 & 0mm \\ -1 & 0 & 0 & 132.4588mm \\ 0 & 1 & 0 & 0mm \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$T_5^4 = \begin{bmatrix} 0 & 1 & 0 & 0mm \\ -1 & 0 & 0 & 0mm \\ 0 & 0 & 1 & 68mm \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

$$T_5^4 = \begin{bmatrix} -1 & 0 & 0 & 0mm \\ 0 & \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} & 84.3755mm \\ 0 & -\frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} & 14.5255mm \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

3.

4.

2 Results

3 Evaluation

1.

$$T_e^0 = \begin{bmatrix} 0 & 0 & 1 & 255.325mm \\ 0 & -1 & 0 & 0mm \\ 1 & 0 & 0 & 222.25mm \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

2. (a)

$$T_e^0 = \begin{bmatrix} 0 & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & 180.542mm \\ 0 & -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & 180.542mm \\ 1 & 0 & 0 & 222.25mm \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8}$$

(b)

$$T_e^0 = \begin{bmatrix} -1 & 0 & 0 & 0mm \\ 0 & \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} & -180.542mm \\ 0 & -\frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} & 41.708mm \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{9}$$
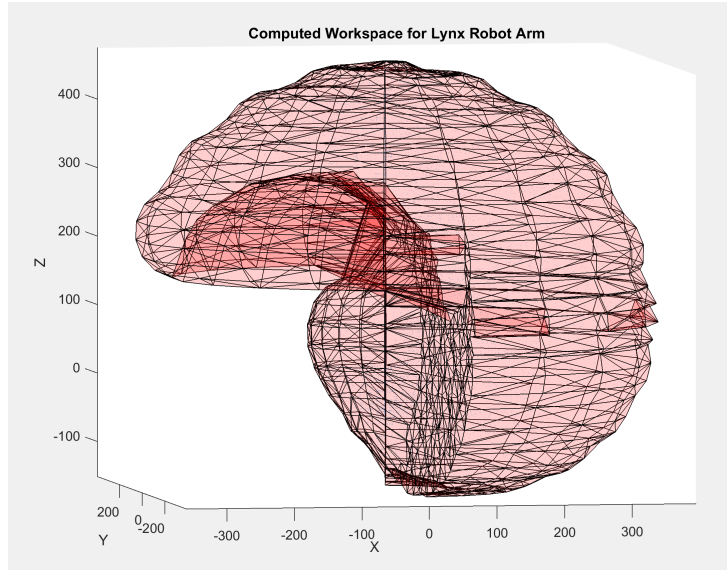
3.

Figure 2: Computed Workspace of Lynx Robot

4.

5. As seen in the code below, the predicted joint positions are almost exactly aligned with the simulated joint positions, aside from a 0.002mm difference in the x position of Joint 3. This is negligible and can be explained due to possible compounded rounding errors in the ROS robot.

However, when examining the T_e^0 matrices, the Simulation T_e^0 records different values of x_0 and y_0 in the end effector frame. We believe that this is due to a mistake in the provided ROS robot code used to calculate Simulation T_e^0. Justification for this conjecture is provided in Problem 1 of the Analysis section.

Listing 1: Outputs of TestFK_Sim.m for $q = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$

```
Simulation  Joint  Positions  =
          0          0          0
          0          0    76.2000
     0.0002     0.0000   222.2500
   187.3252    -0.0002   222.2500
   221.3252    -0.0003   222.2500
   255.3252    -0.0003   222.2500


Predicted  Joint  Positions  =
          0          0          0
          0          0    76.2000
    -0.0000    -0.0000   222.2500
   187.3250    -0.0000   222.2500
   221.3250    -0.0000   222.2500
   255.3250    -0.0000   222.2500


Simulation  T0e  =
     0.0000    -0.0000     1.0000   255.3252
     0.0000     1.0000     0.0000    -0.0003
    -1.0000     0.0000     0.0000   222.2500
          0          0          0     1.0000


Predicted  T0e  =
    -0.0000     0.0000     1.0000   255.3250
     0.0000    -1.0000     0.0000    -0.0000
     1.0000     0.0000     0.0000   222.2500
          0          0          0     1.0000
```

4 Analysis

1. As expected, the results of our evaluation were correct. From the figure below, we use Right-Hand Rule convention to establish that the base frame's red axis is x_0, the green axis is y_0 and the blue axis is z_0. The same axes are applied to the end effector frame, respectively.
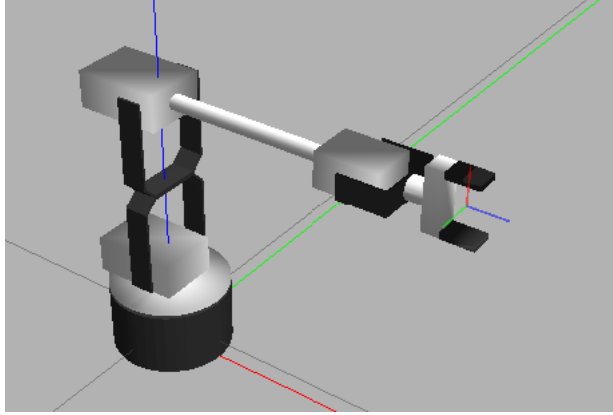

Figure 3: ROS Robot in Zero Configuration

If we use these axes to construct a rotation matrix R by observing each end effector's axis' projection on the base frame's axes, we can compute the following rotation matrix:

$$R = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \tag{10}$$

For example, R states that the projection of the end effector's y axis y_e is in the opposite direction of the base frame's y axis y_0. This value validates the Predicted T_e^0 in Analysis Problem 2.4 and shows that the Simulated T_e^0 is incorrect.

Fortunately, as seen in the same problem, the joint positions were not affected by this error.

2.

5 Appendix