Robotics Nanodegree Project2: Robotic Arm, Pick and Place

Amin Ghafari

October 16, 2017

1 Forward Kinematics and Modified DH Parameters

The annotation for the robot is shown in Fig.1 and it is down based on the explanation from the project walk through. The Modified DH parameters are obtained using urdf files provided for the robot and using RViz to make sure about the length of the links.

2 Transformation Matrices

Here the transformation from link i-1 to link i is shown all the way from the base link to the gripper

$$T_{0\to1} = \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 & 0 \\ \sin(\theta_1) & \cos(\theta_1) & 0 & 0 \\ 0 & 0 & 1 & 0.75 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

$$T_{1\to2} = \begin{bmatrix} \cos(\theta_2 - \pi/2) & -\sin(\theta_2 - \pi/2) & 0 & 0.35 \\ 0 & 0 & 1 & 0 \\ -\sin(\theta_2 - \pi/2) & -\cos(\theta_2 - \pi/2) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

$$T_{2\to3} = \begin{bmatrix} \cos(\theta_3) & -\sin(\theta_3) & 0 & 1.25 \\ \sin(\theta_3) & \cos(\theta_3) & 0 & 0 \\ 0 & 0 & 1 & \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

$$T_{3\to4} = \begin{bmatrix} \cos(\theta_4) & -\sin(\theta_4) & 0 & -0.054 \\ 0 & 0 & 1 & 1.5 \\ -\sin(\theta_4) & -\cos(\theta_4) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

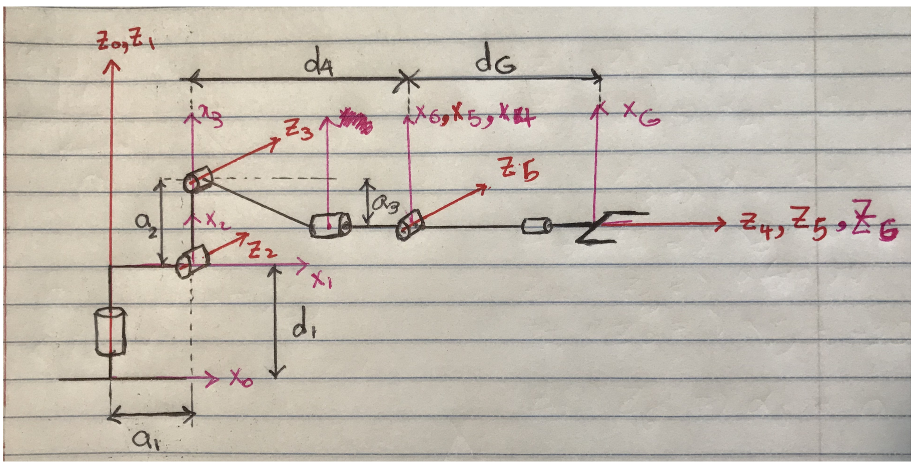

Figure 1: Links of the Robot

Links	α_{i-1}	a_{i-1}	d_{i-1}	$\theta i-1$
$0 \to 1$	0	0	0.75	θ_1
$1 \to 2$	$-\pi/2$	0.35	0	$-\pi/2+\theta_2$
$2 \to 3$	0	1.25	0	θ_3
$3 \to 4$	$-\pi/2$	-0.054	1.5	θ_4
$4 \to 5$	$\pi/2$	0	0	θ_5
$5 \to 6$	$-\pi/2$	0	0	θ_6
$6 \to G$	0	0	0.303	0

Table 1: Modified DH Parameters

$$T_{4\to 5} = \begin{bmatrix} \cos(\theta_5) & -\sin(\theta_5) & 0 & 0 \\ 0 & 0 & -1 & 0 \\ \sin(\theta_5) & \cos(\theta_5) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

$$T_{5\to 6} = \begin{bmatrix} \cos(\theta_6) & -\sin(\theta_6) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\sin(\theta_5) & -\cos(\theta_5) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6}$$

$$T_{6\to G} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0.303 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{7}$$

For the direction homogeneous transformation from the base link to the end effector using the position and orientation of it we have

$$\begin{bmatrix} \cos x_2 \cos x_3 & -\cos x_2 \sin x_3 & \sin x_2 & px \\ \cos x_1 \sin x_3 + \cos x_3 \sin x_1 \sin x_2 & \cos x_1 \cos x_3 - \sin x_3 \sin x_1 \sin x_2 & -\cos x_2 \sin x_1 & py \\ \sin x_1 \sin x_3 - \cos x_1 \cos x_3 \sin x_2 & \cos x_3 \sin x_1 - \cos x_1 \sin x_2 \sin x_3 & -\cos x_2 \sin x_1 & pz \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8}$$

where the position is

$$P_G = \begin{bmatrix} p_x \\ p_y \\ p_x \end{bmatrix} \tag{9}$$

and the angles are x1 : roll, x2 : pitch, x3 : yaw.

3 Inverse Kinematic

Now by having a good sense of the geometry of the robot we can start the Invers Kinematic problem.
Having the position of the grip and its orientation we can find the position of the wrist as it is explained in the project walk through.

$$P_G = \begin{bmatrix} p_x \\ p_y \\ p_x \end{bmatrix} \tag{10}$$

And the orientation of the Gripper:

$$P_G = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \tag{11}$$

We can get the rotation matrix from the writs to the gripper:

$$R_{W\to G} = \begin{bmatrix} l_x & m_x & n_x \\ l_y & m_y & n_y \\ l_z & m_z & n_z \end{bmatrix} R_{corr} \tag{12}$$

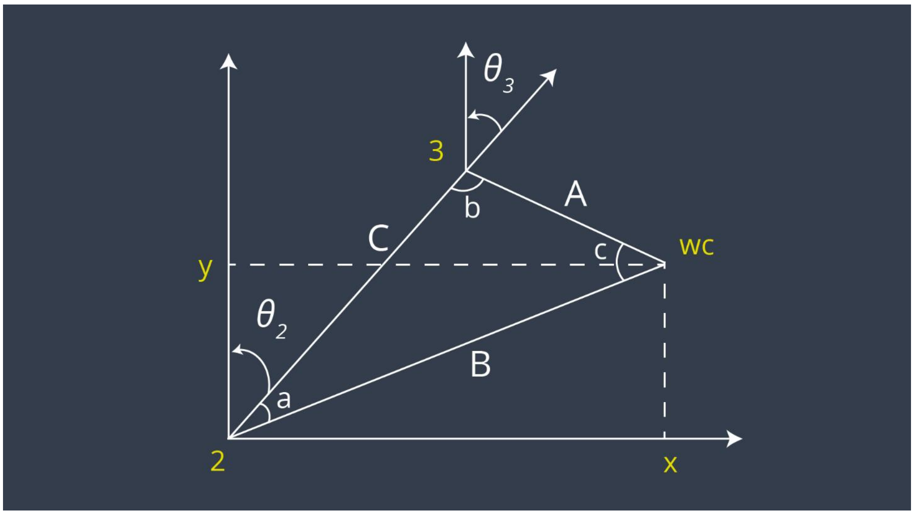

Figure 2: Link 2 and 3 of the robot for IK calculations

or

$$R_{W\to G} = \begin{bmatrix} \cos x_2 \cos x_3 & -\cos x_2 \sin x_3 & \sin x_2 \\ \cos x_1 \sin x_3 + \cos x_3 \sin x_1 \sin x_2 & \cos x_1 \cos x_3 - \sin x_3 \sin x_1 \sin x_2 & -\cos x_2 \sin x_1 \\ \sin x_1 \sin x_3 - \cos x_1 \cos x_3 \sin x_2 & \cos x_3 \sin x_1 - \cos x_1 \sin x_2 \sin x_3 & -\cos x_2 \sin x_1 \end{bmatrix} R_{corr} \tag{13}$$

Where R_{corr} is the correction matrix as it is introduced in the project. And the wrist position can be found by

$$W_x = p_x - d_6.n_x \qquad W_y = p_y - d_6.n_y \qquad W_z = p_z - d_6.n_z \tag{14}$$

Now by having the writs position we can easily find the first 3 angle of rotation for the links. Looking at a figure from project walk through, Fig4,we can find the angle for θ_1 using an arctan argument

$$\theta_1 = \arctan(W_y, W_y) \tag{15}$$

The angle of a, b, c can be calculated using cosine rules:

$$a = \arccos(\frac{L_C^2 + L_B^2 - L_A^2}{2 * L_B * L_C}) \quad b = \arccos(\frac{L_A^2 + L_C^2 - L_B^2}{2 * L_A * L_C}) \quad c = \arccos(\frac{L_A^2 + L_B^2 - L_C^2}{2 * L_A * L_B}) \tag{16}$$

where L_A, L_B, L_C are the length of the triangle sides.

$$L_A = \sqrt{d_4^2 + a_3^2} \quad L_B = \sqrt{Wy_2^2 + Wx_2^2} \quad L_C = a_2 \tag{17}$$

where

$$Wy_2 = \sqrt{W_x^2 + W_y^2} - a_1 \qquad Wx_2 = W_x - d_1 \tag{18}$$

Using this and the angle of the side B with x axis we have

$$\theta_2 = \pi/2 - a - \arctan(Wy_2, Wx_2) \tag{19}$$

And for θ_3 using the b angle the inclination of link 3 toward down we can have

$$\theta_3 = \pi/2 - b - \arctan(-a_3, d_4) \tag{20}$$

Now that we know that the rotation from link 0 to 3 can be calculated by having $\theta_1, \theta_2, \theta_3$ and the fact that the whole rotation matrix from link 0 to the gripper is equal to what we have from roll, yaw and rotation matrix; then, we can have:

$$R_{0\to 6} = R_{0\to 3} * R_{3\to G} = R_{rpy} \tag{21}$$

The next 3 unknowns are hidden in $R_{3\to G}$ that we can computed as a function of $\theta_1, \theta_2, \theta_3$.

Figure 3: An example of the robot arm picking the sample

$$R_{3\to G} = inv(R_{0\to 3}) * R_{rpy} \tag{22}$$

The three rotation matrices are combination of 3 Euler angle rotation. The resultant multiplication is

$$R_{3\to G} \begin{bmatrix} c_4c_6 - s_4s_5s_6 & -c_6s_4s_5 - c_4s_6 & -c_5s_4 \\ c_5s_6 & -c_5c_6 & -s_5 \\ c_4s_5s_6 + c_6s_4 & c_4c_6s_5 - s_4s_6 & c_4c_5 \end{bmatrix} \tag{23}$$

$$\theta_4 = \arctan(R_{3\to G}(3,3), -R_{3\to G}(1,3)) \tag{24}$$

$$\theta_5 = \arctan(\sqrt{R_{3\to G}(3,3)^2 + R_{3\to G}(1,3)^2}, R_{3\to G}(2,3)) \tag{25}$$

$$\theta_6 = \arctan(-R_{3\to G}(2,2), R_{3\to G}(2,1)) \tag{26}$$

4 Project Implementation

The implementation of the project has been done in file IK_server.py and attached in the project. There are some comments in the file to make the reader familiar with the process of implementation of these calculations in a python code. Then, this codes is tested using Ross commands and Gazebo and RViz to make sure that it is working properly.

Although in most cases the robot arm is able to pick and place the sample, the motion of the robot and the last three arms is not smooth and consistent when the trajectory that is determined has a sharp curve or when the arm has a small space to move and reach to the sample. Sometimes extra rotation of the arm can be seen which is mostly due to the angles being close to 0 where due to numerical computations it will turn out to be -0.0001 or +0.0001. This can easily force the arm to have some extra rotation before getting to the desired space.

Figure 4: After dropping off the sample