Robotic Arm: Pick Place

Alfonso Sanchez

March 6, 2018

1 KINEMATIC ANALYSIS

1.1 EVALUATING THE KR210.URDF.XACRO FILE

The table that describes the obtained Denavit-Hartenberg (DH) parameters for the forward kinematics (FK) of the robot shows in figure 1 is presented next.

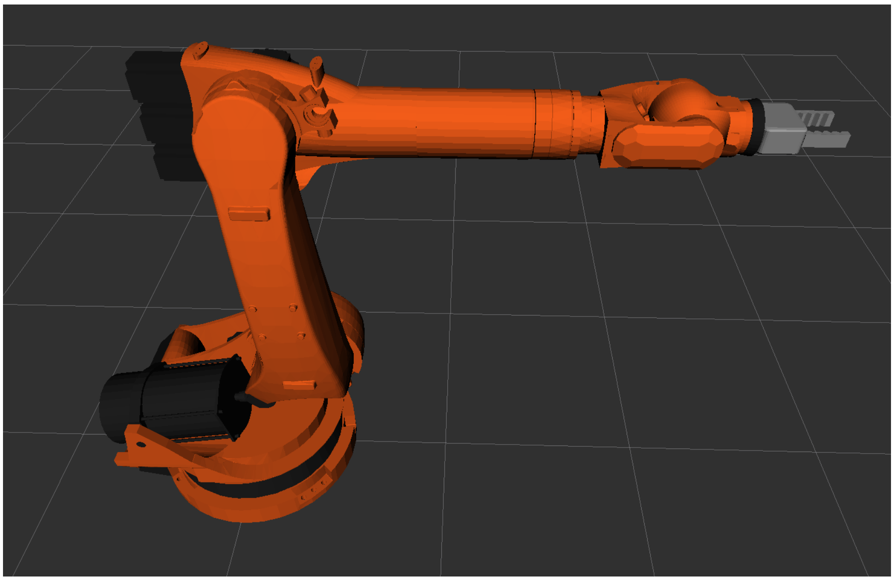

Figure 1: Robot used for the current problem.

i (joint)	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	.75	q_1
2	-$\pi/2$	.35	0	q_2
3	0	1.25	0	q_3
4	-$\pi/2$	-.054	1.5	q_4
5	$\pi/2$	0	0	q_5
6	-$\pi/2$	0	0	q_6
7	0	0	.303	q_7

Table 1: Denavit-Hartenberg parameters.

The alpha parameters were found as the angles between the z_{i-1} and z_i axes, in the x_{i-1} direction following the right sense way. On the other hand, a_{i-1} and d_i are the distances from z_{i-1} and z_i in the x_{i-1} direction and x_{i-1} and from x_i in the z_i direction, respectively. Finally, theta are the angles in which the joints rotate. The nature of these parameters can be seen better in figure 2.

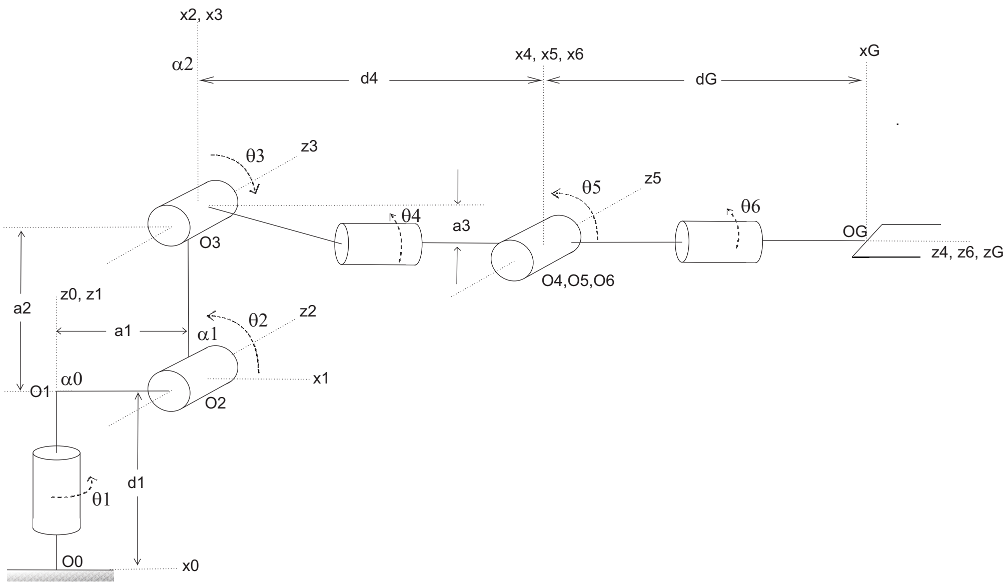

Figure 2: Graphical representation of the DH parameters.

Furthermore, figure 3 is presented in order to introduce a more detailed understanding of the DH parameters and how they were obtained. Additionally, the figure allows to see the links generated between the joints.

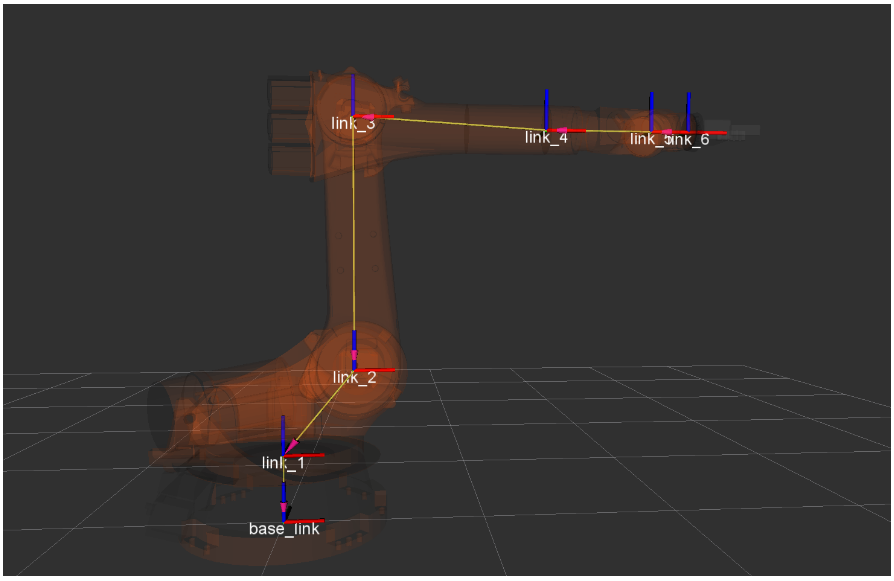

Figure 3: Physical links for the robot used in this problem.

1.2 Rotation and translation transformations

The obtained DH parameters define the initial position for the robot and it is possible to use them to determine the position of each of their joints using rotational transformations and considering the translations between them. The rotations and translations between joints are normally combined in order to obtain homogeneous matrices, also known as homogeneous transformations. These transformations are presented for the current problem in the following matrices, where R_{A-B} represented the homogeneous matrix between the frame for the joint A and the frame for the joint B.

$$R_{0-1} = \begin{bmatrix} cos(\theta_1) & -sin(\theta_1) & 0 & 0 \\ sin(\theta_1) * cos(\alpha_0) & cos(\theta_1) * cos(\alpha_0) & -sin(\alpha_0) & -sin(\alpha_0) * d_1 \\ sin(\theta_1) * sin(\alpha_0) & cos(\theta_1) * sin(\alpha_0) & cos(\alpha_0) & cos(\alpha_0) * d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_{1-2} = \begin{bmatrix} cos(\theta_2) & -sin(\theta_2) & 0 & 0 \\ sin(\theta_2) * cos(\alpha_1) & cos(\theta_2) * cos(\alpha_1) & -sin(\alpha_1) & -sin(\alpha_1) * d_2 \\ sin(\theta_2) * sin(\alpha_1) & cos(\theta_2) * sin(\alpha_1) & cos(\alpha_1) & cos(\alpha_1) * d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_{2-3} = \begin{bmatrix} cos(\theta_3) & -sin(\theta_3) & 0 & 0 \\ sin(\theta_3) * cos(\alpha_2) & cos(\theta_3) * cos(\alpha_2) & -sin(\alpha_2) & -sin(\alpha_2) * d_3 \\ sin(\theta_3) * sin(\alpha_2) & cos(\theta_3) * sin(\alpha_2) & cos(\alpha_2) & cos(\alpha_2) * d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_{3-4} = \begin{bmatrix} cos(\theta_4) & -sin(\theta_4) & 0 & 0 \\ sin(\theta_4) * cos(\alpha_3) & cos(\theta_4) * cos(\alpha_3) & -sin(\alpha_3) & -sin(\alpha_3) * d_4 \\ sin(\theta_4) * sin(\alpha_3) & cos(\theta_4) * sin(\alpha_3) & cos(\alpha_3) & cos(\alpha_3) * d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_{4-5} = \begin{bmatrix} cos(\theta_5) & -sin(\theta_5) & 0 & 0 \\ sin(\theta_5) * cos(\alpha_4) & cos(\theta_5) * cos(\alpha_4) & -sin(\alpha_4) & -sin(\alpha_4) * d_5 \\ sin(\theta_5) * sin(\alpha_4) & cos(\theta_5) * sin(\alpha_4) & cos(\alpha_4) & cos(\alpha_4) * d_5 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_{5-6} = \begin{bmatrix} cos(\theta_6) & -sin(\theta_6) & 0 & 0 \\ sin(\theta_6) * cos(\alpha_5) & cos(\theta_6) * cos(\alpha_5) & -sin(\alpha_5) & -sin(\alpha_5) * d_6 \\ sin(\theta_6) * sin(\alpha_5) & cos(\theta_6) * sin(\alpha_5) & cos(\alpha_5) & cos(\alpha_5) * d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_{6-G} = \begin{bmatrix} cos(\theta_7) & -sin(\theta_7) & 0 & 0 \\ sin(\theta_7) * cos(\alpha_6) & cos(\theta_7 * cos(\alpha_6) & -sin(\alpha_6) & -sin(\alpha_6) * d_7 \\ sin(\theta_7) * sin(\alpha_6) & cos(\theta_7) * sin(\alpha_6) & cos(\alpha_6) & cos(\alpha_6) * d_7 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Moreover, it is possible to determine the position of the gripper as a combination of the DH parameters and the homogeneous matrices, as the product of the homogenous matrices of the frames of the previous joints as following:

$$R_{0-G} = R_{0-1} * R_{1-2} * R_{2-3} * R_{3-4} * R_{4-5} * R_{5-6} * R_{6-G} \tag{1}$$

However, it is necessary to compensate the robot found position with respect to the convention used for the robot as presented in the *urdf* file. Thus, the rectification is implemented rotating the found position in the Y and Z axes. This operation is implemented as the product of the found position and the correction matrix R_{corr} as following:

$$R_{0-G} = R_{0-G} * R_{corr} \tag{2}$$

where the matrix $\mathrm{R}_{corr} isexpressedastheproductbetweenthe \mathrm{R}_{corrY}$ and the R_{corrZ} matrices, defined in equations 4 and 5, respectively.

$$R_{corr} = R_{corrZ} * R_{corrY} \tag{3}$$

$$R_{corrY} = \begin{bmatrix} cos(-\pi/2) & 0 & -sin(-\pi/2) & 0 \\ 0 & 1 & 0 & 0 \\ -sin(-\pi/2) & 0 & cos(-\pi/2) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

$$R_{corrZ} = \begin{bmatrix} cos(\pi) & -sin(\pi) & 0 & 0 \\ sin(\pi) & cos(\pi) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

1.3 DECOUPLING THE INVERSE KINEMATICS PROBLEM

The inverse kinematics problem can be solved as an analytic problem, given the characteristic of the used robot. In this case the wrist center (WC) is defined as the intersection between joints 4, 5 and 6 as presented in figure 4. Then, the solution of the inverse problem can be solved by finding the Cartesian plane that defines the location of the WC and then finding the angles of the joints that determines the orientation of the end effector.

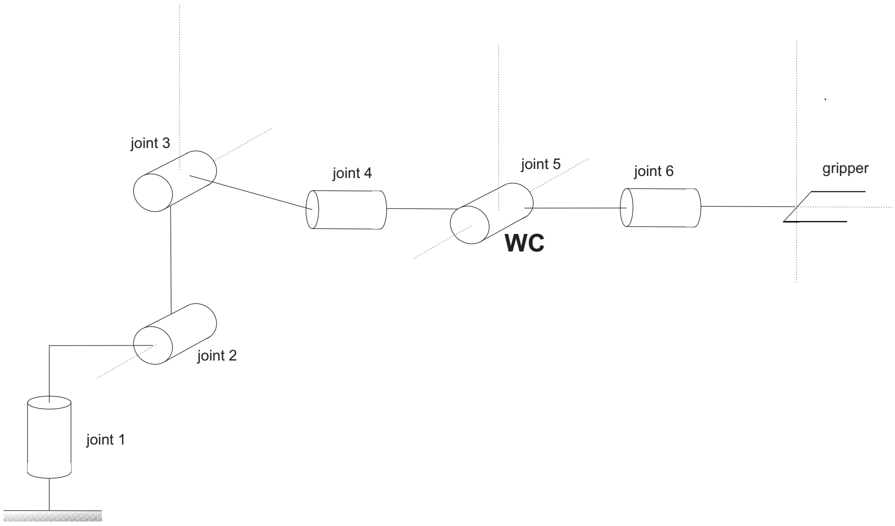

Figure 4: Graphical representation of the wrist center.

The wrist center can be found using equations 6 to 8, where Px, Py and Pz represent the end-effector position, while n_x, n_y and n_z are the components of the orthonormal vector that represents the orientation along the z axes.

$$Wx = Px - (d6 + d7) * n_x \tag{6}$$
$$Wy = Py - (d6 + d7) * n_y \tag{7}$$
$$Wz = Pz - (d6 + d7) * n_z \tag{8}$$

As mentioned in the definition of the problem provided by Udacity, the components of the $\boldsymbol{n}$ vector can be obtained from the Rrpy matrix, presented in equation 16, where the angles roll, pitch and yaw are the rotation angles with respect to the x, y and z axes of the end effector.

The Rot(X, roll), Rot(Y, pitch) and Rot(Z, yaw) matrices can be understood better by observing equations 10, 11 and 12, correspondingly.

$$Rrpy = Rot(Z, yaw) * Rot(Y, pitch) * Rot(X, roll) * R_{corr} \tag{9}$$

$$Rot(X, roll) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & cos(roll) & -sin(roll) & 0 \\ 0 & sin(roll) & cos(roll) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{10}$$

$$Rot(Y, pitch) = \begin{bmatrix} cos(pitch) & 0 & -sin(pitch) & 0 \\ 0 & 1 & 0 & 0 \\ -sin(pitch) & 0 & cos(pitch) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{11}$$

$$Rot(Z, yaw) = \begin{bmatrix} cos(yaw) & -sin(yaw) & 0 & 0 \\ sin(yaw) & cos(yaw) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{12}$$

Once the WC positons is knows, it is possible to determine the expressions for the angles θ_1, θ_2 and θ_3, by observing figures 5 , 6 and 2 in combination with table 1.

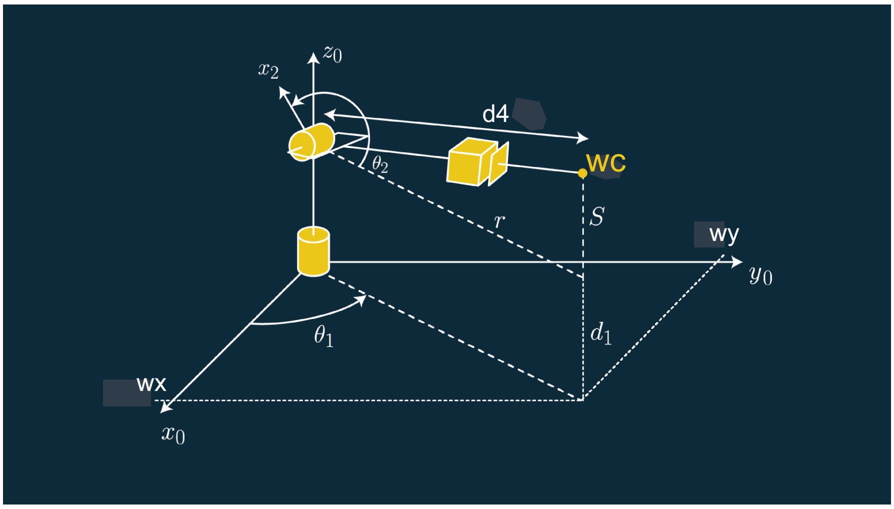

Figure 5: Representation of the rotation angle for the joints 1 and 2, without considering the DH parameter a1.

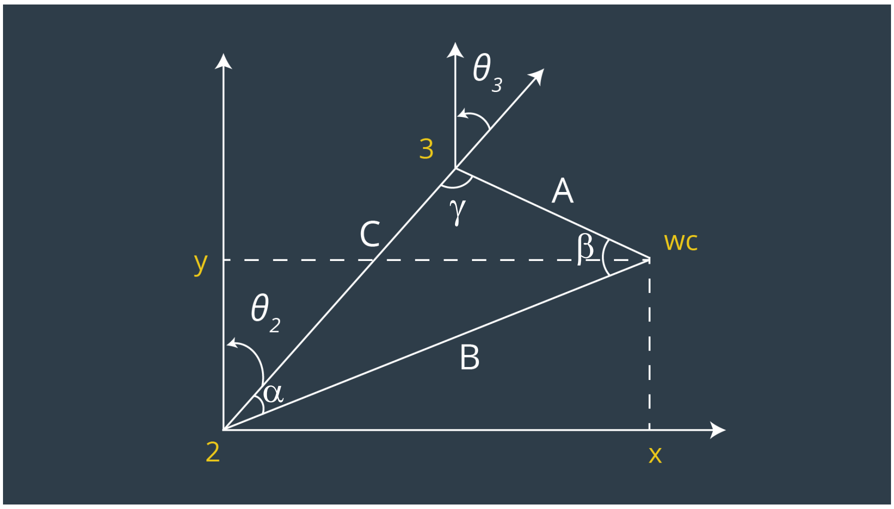

Figure 6: Representation of the rotation angles for joints 2 and 3.

Then, the expressions found for the theta angles are presented in the following equations:

$$\theta_1 = atan2(Wy, Wx) \tag{13}$$
$$\theta_2 = \pi/2 - \alpha - atan2(wz - d_1, r) \tag{14}$$
$$\theta_3 = (\gamma + (\pi - \gamma)) + \theta_2 \tag{15}$$

where $\pi - \gamma$ comes from figure 7

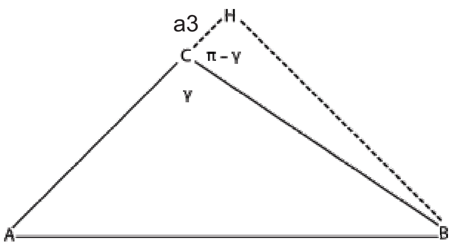

Figure 7: Representation of the angle $\pi - \gamma$, to compensate for the DH parameter a3.

Lastly, the angles that define the positions for the rest of the joints, θ_4, θ_5 and θ_6, are obtained by solving equation 16 for the desired angles.

$$R_{3-6} = inv(R_{0-3}) * Rrpy \tag{16}$$

The expressions found for the desired angles are presented in the following equations:

$$\theta_4 = atan2(R_{3-6}[3,3], -R_{3-6}[1,3]) \tag{17}$$

$$\theta_5 = atan2(\sqrt{R_{3-6}[3,3]^2 + R_{3-6}[1,3]^2}, R_{3-6}[2,3]) \tag{18}$$

$$\theta_6 = atan2(-R_{3-6}[2,2], R_{3-6}[2,1]) \tag{19}$$

2 Project Implementation

2.1 Pick and place process

The pick and place process was implemented as explained in the Udacity nanodegree program, implementing the methods presented in the previous sections of this document. The process starts with the commands that are sent to the robot through the *RViz* program, where the behavior of the robot can be observed in the *Gazebo* window. Thus, the process starts with the following figure 8, which generates the position of the robot presented in figure 9.

Figure 8: Initial state for the robot, visualized in RViz.

Figure 9: Initial state for the robot, visualized in Gazebo.

Then, the RViz program uses the *IK_server.py* file to estimate the inverse kinematics that move the robot to the desire location.

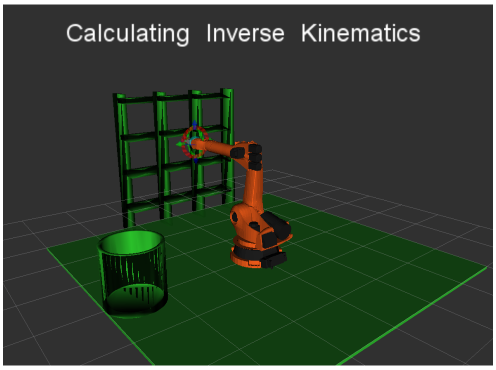

Figure 10: Calculating the inverse kinematics, RViz.

Once the inverse kinematics are calculated, the robot move accordingly to them.

Figure 11: Moving the robot to the target location, RViz.

Figure 12: Moving the robot to the target location, Gazebo.

The next step, after reaching the desired location, is to reach the desired target.

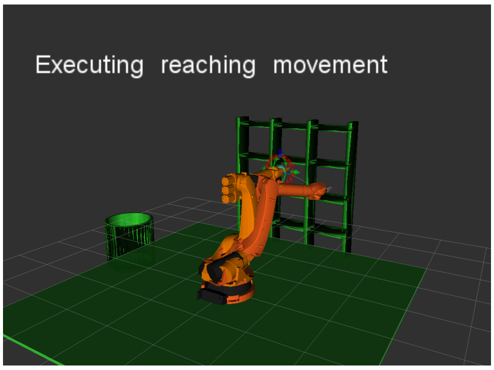

Figure 13: Reaching the desired target, RViz.

After that, the goal is to retrieve the target from its location.

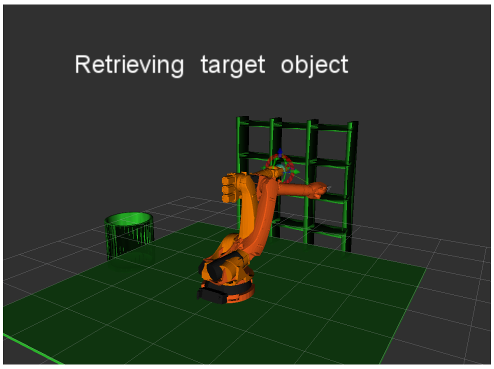

Figure 14: Retrieving the desired target, RViz.

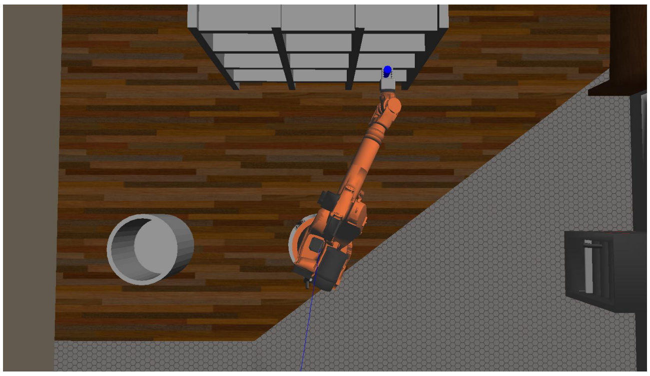

Figure 15: Retrieving the desired target, Gazebo.

Once the target is retrieved, the calculation of the inverse kinematics that determines the movements that the robot needs to implemented in order to place the target in the desired location are calculated.

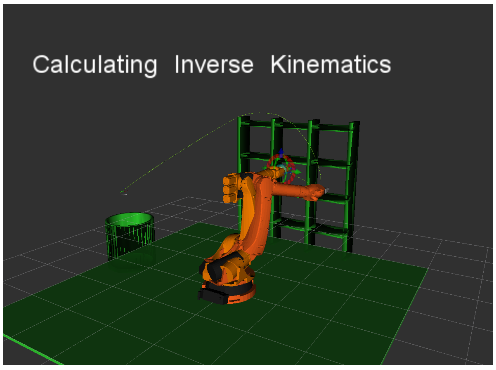

Figure 16: Calculating the inverse kinematics.

The next step is to move the robot to the desired location.

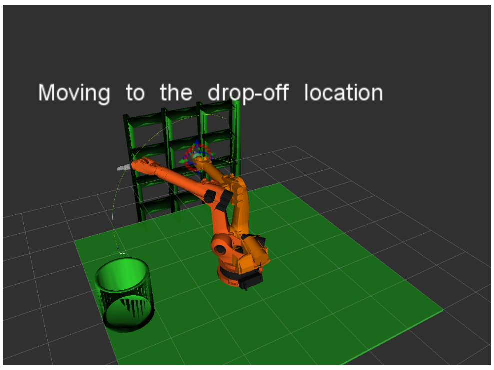

Figure 17: Moving the robot to the release location, RViz.

Figure 18: Moving the robot to the release location, Gazebo.

Finally, once the robot reach the release location, the target object is stored there.

Figure 19: Releasing the target object in the desired location, RViz.

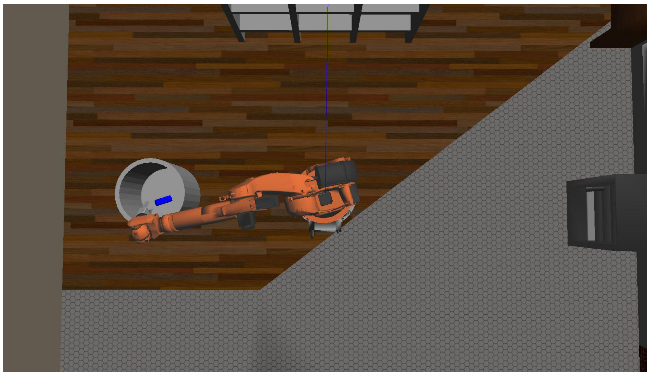

Figure 20: Releasing the target object in the desired location, Gazebo.

2.2 Results

The results obtained from the implementation of the robot presented in figure 1 is shown in the next figure, in which can be observed that the implementation fulfill the requirements of the project. Nonetheless, it is important to remark that the robot behaves in different ways than the expected, especially because of the problems grabbing the targets, even when the gripper was perfectly aligned with them and the found kinematics were smooth.

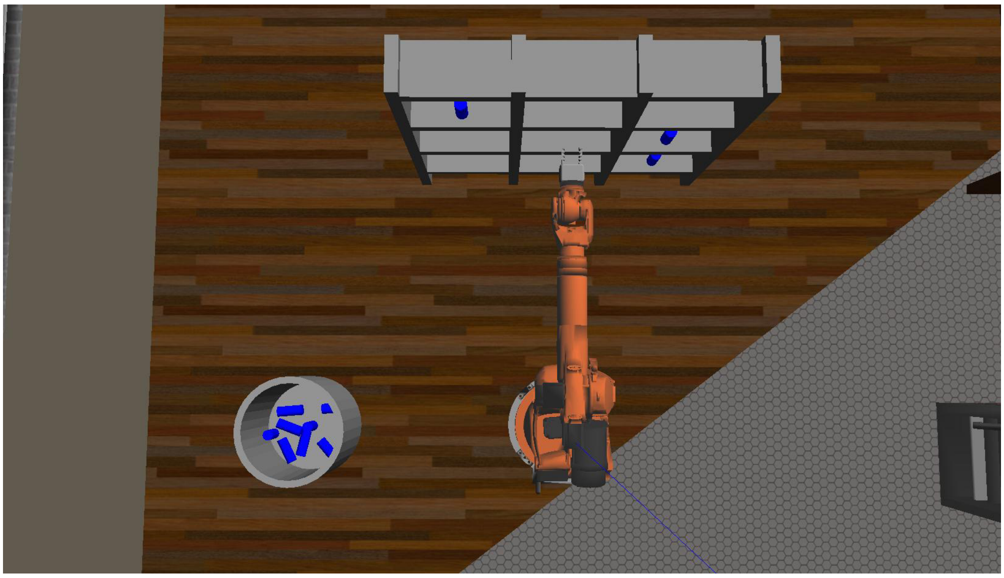

Figure 21: Results, correctly picked and placed 8 targets, out of 10 tries.