

---

# **Kinematic Analysis and Gazebo Simulation**

## **KUKA 210 Robotic Arm**

---

DATE: AUG 6TH, 2018

MENGHONG FENG

# 1 Overview

This report demonstrates the complete process of kinematic analysis procedure for KUKA 210 robotic arm. KUKA 210 is a 6 degree of freedom (DOF) spherical (RRP) robotic arm which has been widely used in industry. The analysis method presented in this report can also be applied in other types of robotic arm, such as cartesian (PPP), cylindrical (RPP), SCARA (RRP), and so on.

Kinematic analysis consists of two process: forward kinematic and inverse kinematic. Forward kinematic is defined as the process to compute end-effector position from specified joint parameters value. Inverse kinematic, on the contrary, refers to process to recover joint condition, such as movement, angle, and position, by knowing the spatial information of end-effector. In this project, the cylinder object may appear in random location and robotic arm is required to pick it and place it to trash bin. To solve this problem we have to go through the process of forward and inverse kinematic analysis. The steps are summarized below:

1. Find joints and label out their axis orientation
2. Refer to urdf.xacro file to derive DH parameters
3. Use DH parameters to create individual transformation matrices
4. Generate a generalized homogeneous transform between base link and end-effector (gripper)
5. Perform inverse position kinematics and inverse orientation kinematics

## 2 Kinematic Analysis

### 2.1 Forward Kinematic

Figure 1 shows a real KUKA 210 robotic arm and its six joints labeled in figure. Figure 2 is a representation to demonstrate joints' axis and orientation. Wrist center is located in joint 5. An end-effector (gripper) is connected to joint 6 and labeled as joint 7. By referring to the urdf.xacro file from ROS (robotic operation system), the spatial information can be found and collected in table 1. Figure 3 is used to visualize each joint's spatial relation.

Table 1: urdf.xacro joint parameters

$Joint_i$	$x_i$	$y_i$	$z_i$
0	0	0	0
1	0	0	0.33
2	0.35	0	0.42
3	0	0	1.25
4	0.96	0	-0.054
5	0.54	0	0
6	0.193	0	0
7	0.11	0	0

The joint parameters value obtained from urdf.xacro file can be used to compute DH table, which is shown in below. The twist angle  $\alpha_i$  is measured between the  $z_{i-1}$  and  $z_i$  axis of adjacent joints. Link length  $a_i$  is measured between the distance of  $z_{i-1}$  and  $z_i$  axis of adjacent joints. Link offset  $d_i$  is the distance between  $x_{i-1}$  and  $x_i$  axis of adjacent joints. Revolute joint angle  $q_i$  or  $\theta_i$  is the angle of joint, which is the orientation variable appeared later in the homogenous transformation.

Table 2: DH Table Parameters

$\alpha_0: 0$	$a_0: 0$	$d_1: 0.75$	$q_1: q_1$
$\alpha_1: -\pi/2$	$a_1: 0.35$	$d_2: 0$	$q_2: q_2 - \pi/2$
$\alpha_2: 0$	$a_2: 1.25$	$d_3: 0$	$q_3: q_3$
$\alpha_3: -\pi/2$	$a_3: -0.054$	$d_4: 1.5$	$q_4: q_4$
$\alpha_4: \pi/2$	$a_4: 0$	$d_5: 0$	$q_5: q_5$
$\alpha_5: -\pi/2$	$a_5: 0$	$d_6: 0$	$q_6: q_6$
$\alpha_6: 0$	$a_6: 0$	$d_7: 0.303$	$q_7: 0$

With the DH table parameters been collected, now it's time to develop individual transformation matrices and homogenous transform bewteen base link with end-effector (gripper). The transformation matrix is defined as below:

$$_i^{i-1}T = R(x_{i-1}, \alpha_{i-1})T(x_{i-1}, a_{i-1})R(z_i, \theta_i)T(z_i, d_i)$$

$$_i^{i-1}T = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & a_i - 1 \\ \sin \theta_i \cos \alpha_{i-1} & \cos \theta_i \cos \alpha_{i-1} & -\sin \alpha_{i-1} & -\sin \alpha_i - d_i \\ \sin \theta_i \sin \alpha_{i-1} & \cos \theta_i \sin \alpha_{i-1} & \cos \alpha_{i-1} & \cos \alpha_i - d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_{0EE} = T_{01} \times T_{12} \times T_{23} \times T_{34} \times T_{45} \times T_{56} \times T_{6EE}$$

The first equation is individual transformation matric from joint i-1 to joint i, which consists of two translations and two rotations along  $z$  axis and  $x$  aixs. The matric form is shown as second equation. The thrid equation is the homogenous transformation between base link to end-effector(gripper).

One thing we need to be careful about is that the orientation of end-effector (gripper) defined in urdf file is different from the one in DH convention. As shown in figure 4, the green coordinate is urdf reference and orange coordinate is DH reference frame. To compensate the difference, we need to rotate the urdf reference frame to  $180^\circ$  along  $z$  axis and  $-90^\circ$  along  $y$  axis. Three rotation matrices along  $x$ ,  $y$ , and  $z$  axis are defined below along with the orientation correction equation:

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix} \quad (1) \quad R_y(\theta) = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix} \quad (2) \quad R_z(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$$R_{corr} = R_z(180^\circ) \times R_y(-90^\circ)$$

$$T_{0EE} = T_{0EE} \times R_{corr}$$

## 2.2 Inverse Kinematic

Inverse kinematic is the process of recovering joint condition by giving the position and orientation of end-effector (gripper). In this project, the position of end-effector can be retrieved by using ROS command. The joints 4, 5, and 6 form together as spherical wrist, with wrist center at joint 5. Therefore, the first step is to recover the wrist center (WC) position by applying inverse position kinematic analysis. The equation to find WC position is:

$$\begin{bmatrix} WC_x \\ WC_y \\ WC_z \end{bmatrix} = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} - {}^0_6R \begin{bmatrix} 0 \\ 0 \\ d_G \end{bmatrix}$$

Or the matrix form can be expressed as individual equation:

$$WC_x = p_x - (d_6 + l) \cdot n_x$$

$$WC_y = p_y - (d_6 + l) \cdot n_y$$

$$WC_z = p_z - (d_6 + l) \cdot n_z$$

Where  $WC_{x,y,z}$  represent wrist center (joint 5) position,  $p_{x,y,z}$  are the position of end-effector (gripper), which are given in this project.  $d_6 + l$  is the distance between wrist center to end-effector.  $n_{x,y,z}$  form the vector along the z-axis of the gripper link, which can be obtained from end-effector pose  $R_{rpy}$  with respect to the base link by using *eulerfromquaternions()* method.

The end-effector pose with respect to base link can be calculated as below. Correction matrice must be applied to convert reference frame from urdf to DH parameter frame. At last, the  $n_{x,y,z}$  values can be

extracted from  $R_{rpy}$ .

$$R_{rpy} = R_z(yaw) \times R_y(pitch) \times R_x(roll) \times R_{corr}$$

With the wrist center position been found, the inverse position kinematic has been completed. The next step is to perform inverse orientation kinematic to find the rotational angle of each joints. As shown in figure 5, the top view shows that rotation angle of joint 1 is simply the arctan of  $WC_y$  and  $WC_x$ . In figure 6 and 7, the side view shows that rotation angle of joint 2 and joint 3 can be calculated by using the equation below. Please be aware that the account for  $WC_y$  is due to the dynamic rotation movement robotic arm.  $WC_y$  equals zero when roboitc arm overlaps with  $x$  axis.

$$\begin{aligned}\theta_1 &= \tan^{-1} \frac{WC_y}{WC_x} \\ \theta_2 &= 90^\circ - \alpha - \tan^{-1} \frac{WC_z - d_1}{\sqrt{WC_x^2 + WC_y^2} - a_1} \\ \theta_3 &= 90^\circ - \beta - \gamma\end{aligned}$$

Where the angle  $\alpha$ ,  $\beta$ , and  $\gamma$  can be calculated as:

$$\begin{aligned}\alpha &= \cos^{-1} \frac{side_1^2 + side_3^2 - side_2^2}{2 \times side_1 \times side_3} \\ \beta &= \cos^{-1} \frac{side_2^2 + side_3^2 - side_1^2}{2 \times side_2 \times side_3} \\ \gamma &= \tan^{-1} \frac{a_3}{0.96 + 0.54} \\ side_1 &= \sqrt{(\sqrt{WC_x^2 + WC_y^2} - a_1)^2 + (WC_z - d_1)^2} \\ side_2(\theta_2) &= d_4 = \sqrt{\sqrt{0.96^2 + 0.54^2} + 0.054^2} \\ side_2(\theta_3) &= 0.96 + 0.54 \\ side_3 &= a2\end{aligned}$$

The rotation angle of joint 4, 5, and 6 can be calculated by using Euler angle from quaternion method. The first step is to build the rotational matrix from joint 4 to 6, and the rotation angle of each joint can be extracted from it. The calculation steps are shown below:

$$R_{36} = inv(R_{03}) \times R_{rpy}$$

$$R_{36} = R_Z(\theta_4)R_Y(\theta_5)R_Z(\theta_6) = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

$$R_{36} = \begin{bmatrix} \cos(\theta_4)\cos(\theta_5) & \cos(\theta_4)\sin(\theta_5)\sin(\theta_6) - \sin(\theta_4)\cos(\theta_6) & \cos(\theta_4)\sin(\theta_5)\cos(\theta_6) - \sin(\theta_4)\sin(\theta_5) \\ \sin(\theta_4)\cos(\theta_5) & \sin(\theta_4)\sin(\theta_5)\sin(\theta_6) + \cos(\theta_4)\cos(\theta_6) & \sin(\theta_4)\sin(\theta_5)\cos(\theta_6) - \cos(\theta_4)\sin(\theta_5) \\ -\sin(\theta_5) & \cos(\theta_5)\sin(\theta_6) & \cos(\theta_5)\cos(\theta_6) \end{bmatrix}$$

The rotation angle  $\theta_4$ ,  $\theta_5$ , and  $\theta_6$  can be calculated as follow:

$$\begin{aligned} \theta_4 &= \tan^{-1} \frac{r_{21}}{r_{11}} \\ \theta_5 &= \tan^{-1} \frac{-r_{31}}{\sqrt{r_{11}^2 + r_{21}^2}} \\ \theta_6 &= \tan^{-1} \frac{r_{32}}{r_{33}} \end{aligned}$$

### 3 Project Implementation

The pick & place task has been performed 10 times. 8 out of 10 tasks succeed in picking object and placing it in trash bin. 2 out of 10 tasks fail for the same reason. I realize that inverse kinematic calculation process will takes forever if I use simplify function to round off float point results. Also the by assigning numerical values directly in equation will considerably reduce the calculation time. Another issue found in the task is that, though the task succeeds at the end, the end-effector performs several unnecessary rotation during the task. It is because I forgot to correct reference frame for wrist center. After I fix this issue the robotic behaves normally. The screenshots for successfully completing task are attached in the Appendix section.

In Appendix section, figure 8 to figure 20 shows each step of task in KViz monitor. Figure 21 to figure 24 are the screenshots of dynamic simulation in Gazebo. Figure 25 to figure 32 shows the complete command lists in terminal.

## 4 Appendix



Figure 1: KUKA 210 Robotic Arm

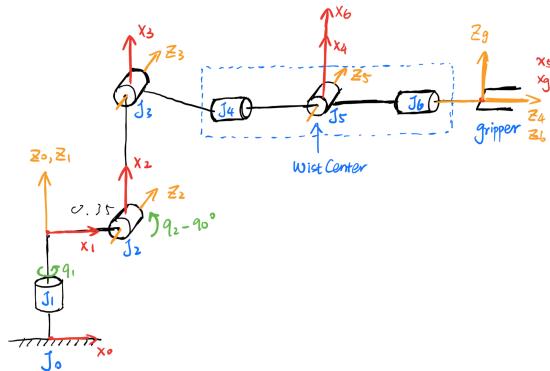


Figure 2: Axis Label and Orientation

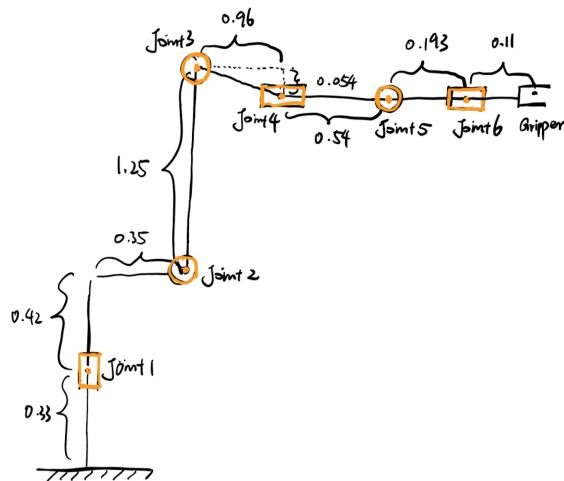


Figure 3: Joint Spatial Relation

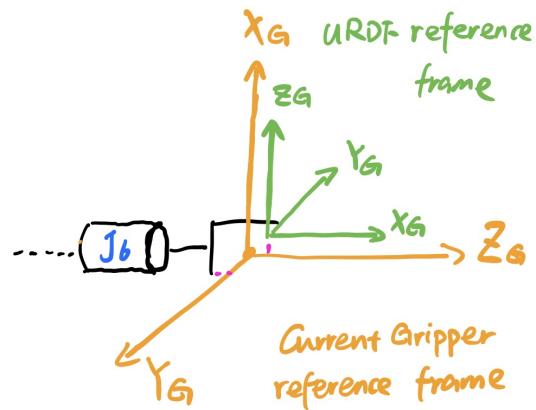


Figure 4: Reference Frame

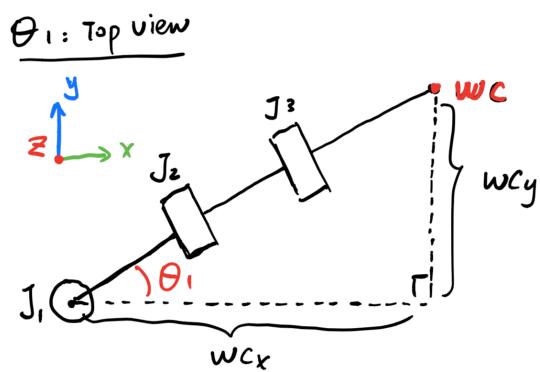


Figure 5:  $\theta_1$

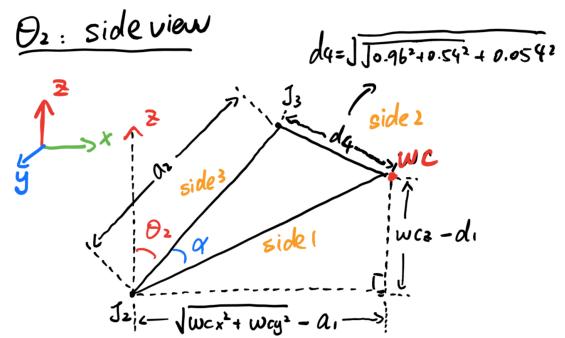


Figure 6:  $\theta_2$

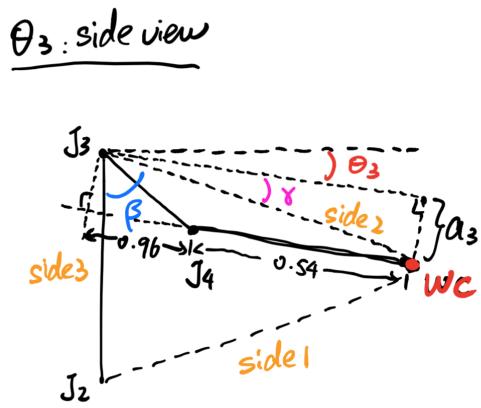


Figure 7:  $\theta_3$

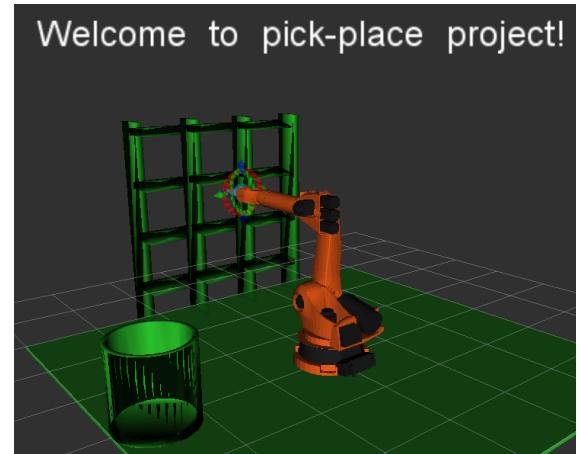


Figure 8: KViz: Start

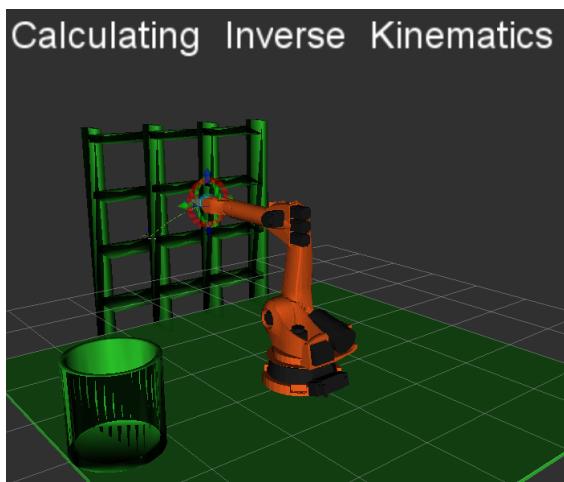


Figure 9: KViz: IK calculation

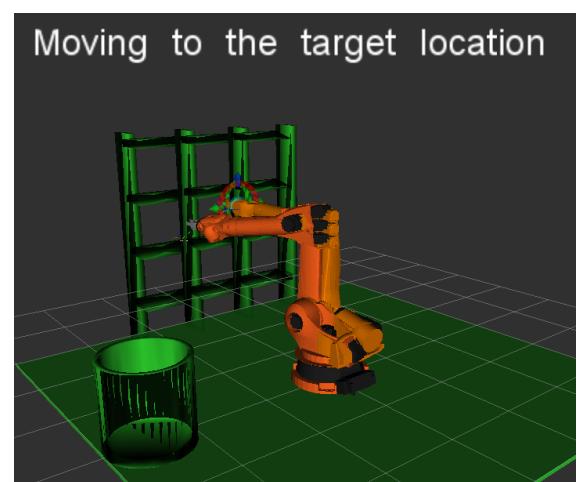


Figure 10: KViz: Moving to Target Location

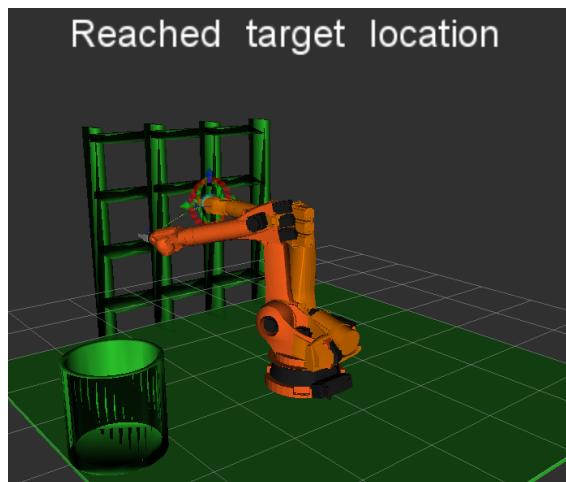


Figure 11: KViz: Reach to Target Location

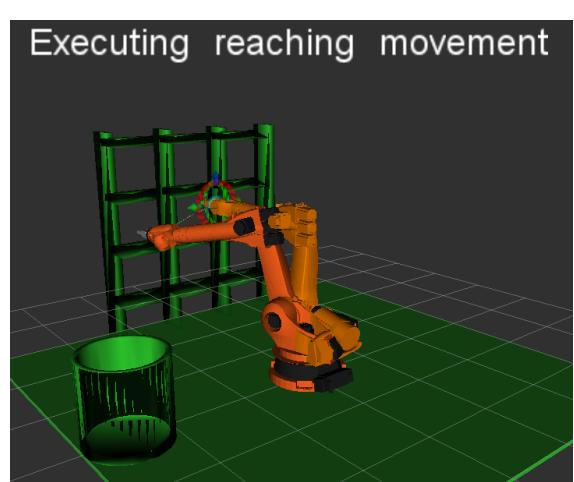


Figure 12: KViz: Executing Reaching Movement

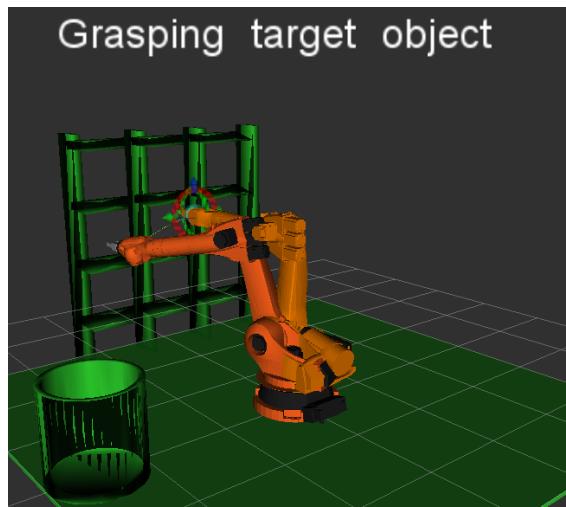


Figure 13: KViz: Grasping Target Object

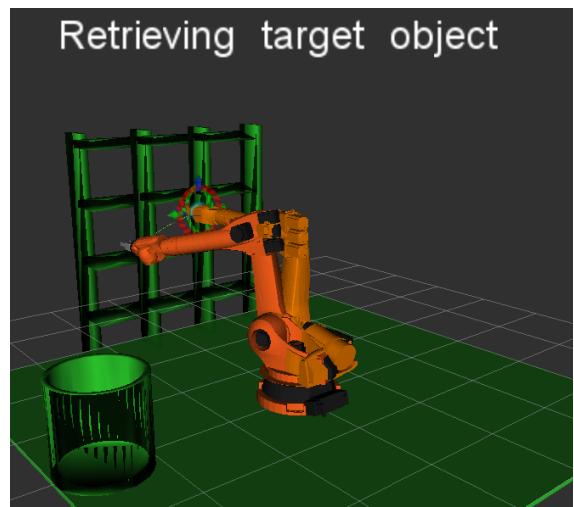


Figure 14: KViz: Retrieving Target Object

Displaying plan to drop-off location

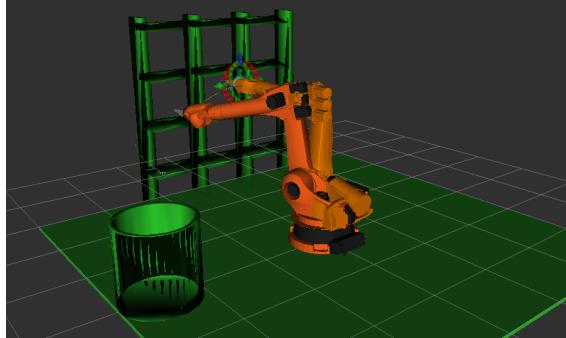


Figure 15: KViz: Display Plan to Drop-off Location

Calculating Inverse Kinematics

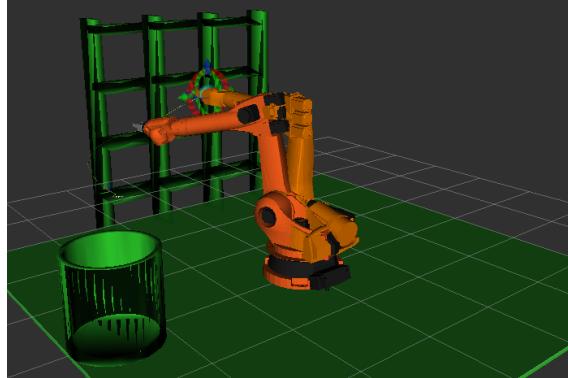


Figure 16: KViz: IK Calculation

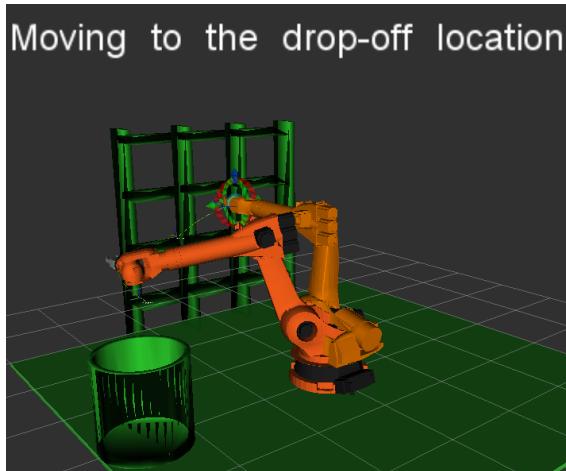


Figure 17: KViz: Moving to Drop-off Location

Reached drop-off location

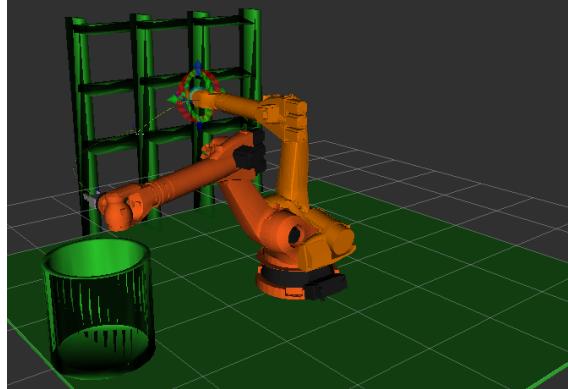


Figure 18: KViz: Reached to Drop-off Location



Figure 19: KViz: Releasing Target Object

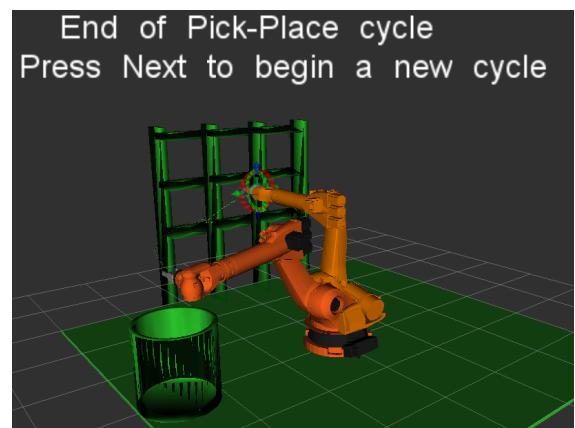


Figure 20: KViz: End

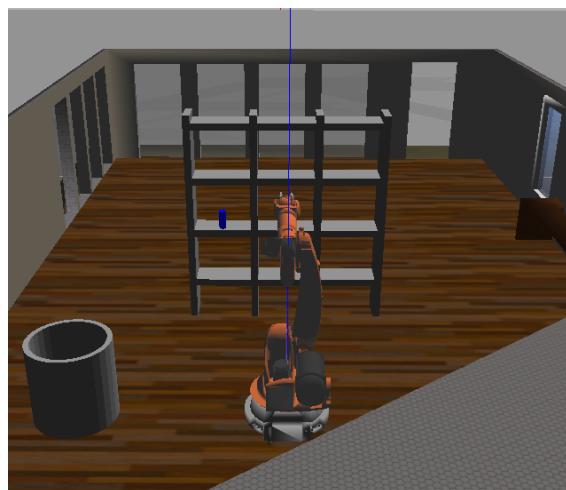


Figure 21: Gazebo: Start



Figure 22: Gazebo: Reach and Grasp



Figure 23: Gazebo: Move to Drop-off Location



Figure 24: Gazebo: Release Target Object to Drop-off Location

```

/home/robond/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/launch/cafe.launch http://localhost:11311
/home/robond/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/launch/cafe.launch http://localhost:11311
request received for MoveGroup action. Forwarding to planning and execution pipe
Line
[ INFO] [1533495360.287514925, 1206.037000000]: Planning attempt 1 of at most 1
[ERROR] [1533495360.287681609, 1206.037000000]: Found empty JointState message
[ERROR] [1533495360.287745481, 1206.037000000]: Found empty JointState message
[ERROR] [1533495360.287747064, 1206.037000000]: Found empty JointState message
[ERROR] [1533495360.288345679, 1206.037000000]: Found empty JointState message
[INFO] [1533495360.288345679, 1206.037000000]: Found empty JointState message
[INFO] [1533495360.289510218, 1206.037000000]: gripper_group[RRTkConfigDefault]
: Starting planning with 1 states already in datastructure
[ INFO] [1533495360.330255354, 1206.052000000]: gripper_group[RRTkConfigDefault]
: Created 32 states
[ INFO] [1533495360.330976682, 1206.052000000]: Solution found in 0.041721 seconds
[ INFO] [1533495360.345360708, 1206.056000000]: SimpleSetup: Path simplification
took 0.013860 seconds and changed from 3 to 17 states
[ WARN] [1533495360.400979646, 1206.067000000]: Dropping first 1 trajectory point(s) out of 28, as they occur before the current time.
First valid point will be reached in 0.042s.
GazeboGraspFix: Detaching target_model::target_link_1::target_link_1_collision from gripper_kuka_arm!!!!!!
[!] robond@udacity: ~/catkin_ws/s...matics-Project/kuka_arm/scripts - + x
[!] robond@udacity: ~/catkin_ws/s...matics-Project/kuka_arm/scripts 80x24
Do you want to source ROS in this workspace (y/n): y
ROS sourced!
robond@udacity:~$ cd ~/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/scripts
robond@udacity:~/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/scripts$ rosrun kuka_arm IK_server.py
Ready to receive an IK request
[INFO] [1533495132.018767, 1126.715000]: Received 10 eef-poses from the plan
[INFO] [1533495139.906620, 1129.544000]: length of Joint Trajectory List: 10
[INFO] [1533495229.542371, 1162.857000]: Received 37 eef-poses from the plan
[INFO] [1533495249.804495, 1167.427000]: length of Joint Trajectory List: 37

```

Figure 25: Run 1

```

/home/robond/catkin_ws/src/Rob...e.launch http://localhost:11311 - + x
/home/robond/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/launch/cafe.launch http://localhost:11311
request received for MoveGroup action. Forwarding to planning and execution pipe
Line
[ INFO] [1533498144.764386315, 1209.902000000]: Planning attempt 1 of at most 1
[ERROR] [1533498144.764587817, 1209.902000000]: Found empty JointState message
[ERROR] [1533498144.764685124, 1209.902000000]: Found empty JointState message
[ERROR] [1533498144.764888519, 1209.902000000]: Found empty JointState message
[ERROR] [1533498144.765221974, 1209.902000000]: Found empty JointState message
[INFO] [1533498144.765221974, 1209.902000000]: Found empty JointState message
[INFO] [1533498144.765393561, 1209.902000000]: Found empty JointState message
[INFO] [1533498144.765405051, 1209.902000000]: Found empty JointState message
[INFO] [1533498144.765405051, 1209.902000000]: Found empty JointState message
[INFO] [1533498144.766009204, 1209.924000000]: gripper_group[RRTkConfigDefault]
: Starting planning with 1 states already in datastructure
[INFO] [1533498144.874657041, 1209.924000000]: gripper_group[RRTkConfigDefault]
: Created 33 states
[INFO] [1533498144.874923568, 1209.924000000]: Solution found in 0.109098 seconds
[INFO] [1533498144.903600124, 1209.925000000]: SimpleSetup: Path simplification
took 0.028431 seconds and changed from 4 to 33 states
[WARN] [1533498144.987635067, 1209.953000000]: Dropping first 1 trajectory point(s) out of 33, as they occur before the current time.
First valid point will be reached in 0.064s.
GazeboGraspFix: Detaching target_model::target_link_1::target_link_1_collision from gripper_kuka_arm!!!!!!
[!] robond@udacity: ~/catkin_ws/s...matics-Project/kuka_arm/scripts - + x
[!] robond@udacity: ~/catkin_ws/s...matics-Project/kuka_arm/scripts 80x24
Ready to receive an IK request
[INFO] [1533497844.400979646, 1124.385000]: Received 10 eef-poses from the plan
[INFO] [1533497853.552183, 1127.014000]: length of Joint Trajectory List: 10
[INFO] [1533497985.629901, 1163.716000]: Received 49 eef-poses from the plan
[INFO] [1533497985.629901, 1163.716000]: length of Joint Trajectory List: 49

```

Figure 26: Run 2

```

[ /home/robond/catkin_ws/src/Rob...e.launch http://localhost:11311 - + x
[ /home/robond/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/launch/cafe.launch http://loca
request received for moveGroup action. Forwarding to planning and execution pipe
line
[ INFO] [1533500865.524254914, 1505.014000000]: Planning attempt 1 of at most 1
[ERROR] [1533500865.524242181, 1505.014000000]: Found empty JointState message
[ERROR] [1533500865.524489470, 1505.014000000]: Found empty JointState message
[ERROR] [1533500865.524658398, 1505.014000000]: Found empty JointState message
[ERROR] [1533500865.525003532, 1505.014000000]: Found empty JointState message
[ INFO] [1533500865.52533730, 1505.014000000]: Planner configuration 'gripper_g
roup(RRTkConfigDefault)' will use planning geometric:RRT. Additional configura
tion parameters will be set when the planning is constructed.
[ INFO] [1533500865.525683379, 1505.016000000]: gripper_group(RRTkConfigDefault)
: Starting planning with 1 states already in datastructure
[ INFO] [1533500865.575638312, 1505.023000000]: gripper_group(RRTkConfigDefault)
: Created 131 states
[ INFO] [1533500865.576013667, 1505.014000000]: Solution found in 0.050470 secon
ds
[ INFO] [1533500865.580869483, 1505.077000000]: SimpleSetup: Path simplification
took 0.032373 seconds and changed from 4 to 33 states
[ WARN] [1533500865.63002451, 1505.036000000]: Dropping first 1 trajectory point
(s) out of 33, as they occur before the current time.
First valid point will be reached in 0.021s.
GazeboGraspFix: Detaching target_model::target_link_1::target_link_1_collision f
rom gripper kuka arm!!!!!!!
[robond@udacity:~/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/scripts$ rosru
Ready to receive an IK request
[ INFO] [1533500865.524881, 1302.840000000]: Received 10 eef-poses from the plan
[ INFO] [1533500865.525003532, 1302.840000000]: length of Joint Trajectory List: 10
[ INFO] [1533500421.078935, 1356.0340000]: Received 45 eef-poses from the plan
[ INFO] [1533500444.508028, 1362.2390000]: length of Joint Trajectory List: 45

```

Figure 27: Run 3

```

[ robond@udacity: ~/catkin_ws/s...matics-Project/kuka_arm/scripts - + x
[ robond@udacity: ~/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/scripts 80x24
Do you want to source ROS in this workspace (y/n): y
ROS sourced!
[robond@udacity:~/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/scripts$ rosru
n kuka_arm IK server.py
Ready to receive an IK request
[INFO] [1533509348.773819, 1122.841000]: Received 25 eef-poses from the plan
[INFO] [1533509360.854273, 1127.441000]: length of Joint Trajectory List: 25
[INFO] [1533509352.365076, 1205.891000]: Received 52 eef-poses from the plan
[INFO] [1533509345.594025, 1209.479000]: length of Joint Trajectory List: 32
[ /home/robond/catkin_ws/src/Rob...e.launch http://localhost:11311 - + x
[ /home/robond/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/launch/cafe.launch http://loca
request received for moveGroup action. Forwarding to planning and execution pipe
line
[ INFO] [1533509779.606706950, 1294.399000000]: Planning attempt 1 of at most 1
[ERROR] [1533509779.606956380, 1294.399000000]: Found empty JointState message
[ERROR] [1533509779.607025220, 1294.399000000]: Found empty JointState message
[ERROR] [1533509779.607203063, 1294.399000000]: Found empty JointState message
[ERROR] [1533509779.607532186, 1294.399000000]: Found empty JointState message
[ INFO] [1533509779.607857439, 1294.399000000]: Planner configuration 'gripper_g
roup(RRTkConfigDefault)' will use planning geometric:RRT. Additional configura
tion parameters will be set when the planning is constructed.
[ INFO] [1533509779.608594357, 1294.399000000]: gripper_group(RRTkConfigDefault)
: Starting planning with 1 states already in datastructure
[ INFO] [1533509779.609219930, 1294.402000000]: gripper_group(RRTkConfigDefault)
: Created 22 states
[ INFO] [1533509779.613270491, 1294.402000000]: SimpleSetup: Path simplification
took 0.000147 seconds and changed from 5 to 41 states
[ WARN] [1533509779.675296722, 1294.422000000]: Dropping first 1 trajectory point
(s) out of 41, as they occur before the current time.
First valid point will be reached in 0.063s.
GazeboGraspFix: Detaching target_model::target_link_1::target_link_1_collision f
rom gripper kuka arm!!!!!!!

```

Figure 28: Run 4

```

[ robond@udacity: ~/catkin_ws/s...matics-Project/kuka_arm/scripts - + x
[ robond@udacity: ~/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/scripts 80x24
Do you want to source ROS in this workspace (y/n): y
ROS sourced!
[robond@udacity:~/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/scripts$ rosru
n kuka_arm IK server.py
Ready to receive an IK request
[INFO] [1533510006.561681, 1111.016000]: Received 73 eef-poses from the plan
[INFO] [1533510036.338827, 1123.438000]: length of Joint Trajectory List: 73
[INFO] [1533510041.533199422, 1274.713000000]: Received 45 eef-poses from the plan
[INFO] [1533510239.519806, 1209.228000]: length of Joint Trajectory List: 60
[ INFO] [1533510416.442147754, 1274.698000000]: Solution found in 0.018454 secon
ds
[ INFO] [1533510416.471768407, 1274.713000000]: SimpleSetup: Path simplification
took 0.000147 seconds and changed from 5 to 41 states
[ WARN] [1533510416.533199422, 1274.713000000]: Dropping first 1 trajectory point
(s) out of 28, as they occur before the current time.
First valid point will be reached in 0.041s.
GazeboGraspFix: Detaching target_model::target_link_1::target_link_1_collision f
rom gripper kuka arm!!!!!!!

```

Figure 29: Run 5

```

[ robond@udacity: ~/catkin_ws/s...matics-Project/kuka_arm/scripts - + x
[ robond@udacity: ~/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/scripts 80x24
Do you want to source ROS in this workspace (y/n): y
ROS sourced!
[robond@udacity:~/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/scripts$ rosru
n kuka_arm IK server.py
Ready to receive an IK request
[INFO] [1533525224.878148, 1137.157000]: Received 29 eef-poses from the plan
[INFO] [1533525235.365817, 1140.799000]: length of Joint Trajectory List: 29
[INFO] [1533525356.653771, 1177.632000]: Received 39 eef-poses from the plan
[INFO] [1533525371.223259, 1180.859000]: length of Joint Trajectory List: 39

```

Figure 30: Run 6

```
[robond@udacity: ~/catkin_ws/s...matics-Project/kuka_arm/scripts - + x
robond@udacity: ~/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/scripts 80x24
Do you want to source ROS in this workspace (y/n): y
ROS sourced!
robond@udacity:~/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/scripts
robond@udacity:~/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/scripts$ ls
IK server.py IK solver.py IK solver.pyc safe spawnr.sh target spawnv.py
robond@udacity:~/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/scripts$ rosrun
n kuka arm IK_server.py
Ready to receive an IK request
[INFO] [1533675492.659000, 1181.263000]: Received 10 eef-poses from the plan
[INFO] [1533675493.085057, 1120.659000]: length of Joint Trajectory List: 10
[INFO] [1533675492.548732, 1177.078000]: Received 41 eef-poses from the plan
[INFO] [1533675114.095638, 1181.838000]: length of Joint Trajectory List: 41
[ WARN] [1533675240.168695296, 1220.144000000]: Dropping first 1 trajectory point(s) out of 45, as they occur before the current time.
First valid point will be reached in 0.031s.
GazeboGraspFix: Detaching target_model::target_link_1::target_link_1_collision from gripper kuka_arm!!!!!!
[
```

Figure 31: Run 7

```
[robot@udacity: ~] catkin_ws...matics-Project/kuka_arm/scripts - + x
[robot@udacity: ~] catkin_ws...matics-Project/kuka_arm/scripts 80x24
Do you want to source ROS in this workspace (y/n): y
ROS sourced
[robot@udacity: ~] cd ~/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/scripts
[robot@udacity: ~/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/scripts] rosrun kuka_arm IK_server.py
Ready to receive an IK request
[INFO] [1533675619.396546, 1157.450000]: Received 10 eef-poses from the plan
[INFO] [1533675501.996626, 1123.592000]: Length of Joint Trajectory List: 10
[INFO] [1533675609.753268, 1155.387000]: Received 10 eef-poses from the plan
[INFO] [1533675619.396546, 1157.450000]: length of Joint Trajectory List: 10
[robot@udacity: ~] 

[robot@udacity: ~] /home/robond/catkin_ws/src/Robo...e.launch http://localhost:11311 - + x
[robot@udacity: ~] /home/robond/catkin_ws/src/RoboND-Kinematics-Project/kuka_arm/launch/camera.launch http://loca
request received for moveGroup action. Forwarding to planning and execution pipe
line.
[ INFO] [1533675689.266304935, 1174.410000000]: Planning attempt 1 of at most 1
[ INFO] [1533675689.266304935, 1174.410000000]: Found empty JointState message
[ERROR] [1533675689.266304935, 1174.410000000]: Found empty jointstate message
[ERROR] [1533675689.266304935, 1174.410000000]: Found empty jointstate message
[ERROR] [1533675689.267232026, 1174.410000000]: Found empty JointState message
[ INFO] [1533675689.267669472, 1174.410000000]: Planner configuration 'gripper_g
roup[RRTConfigDefault]' will use planer 'geometric_irt'. Additional configura
tion parameters will be set when the planner is constructed
[ INFO] [1533675689.267669472, 1174.410000000]: Using RRTConfigDefault
: Starting planning with 1 states already in datastructure
[ INFO] [1533675689.339106772, 1174.424000000]: gripper_group[RRTConfigDefault]
: Created 45 states
[ INFO] [1533675689.339299212, 1174.424000000]: Solution found in 0.071353 secon
ds
[ INFO] [1533675689.363473088, 1174.427800000]: SimpleSetup: Path simplification
took 0.023910 seconds and changed from 35 to 33 states
[ WARN] [1533675689.404883029, 1174.430800000]: Dropping first 1 trajectory poin
t(s) out of 33, as they occur before the current time.
First valid point will be reached in 0.057s.
Gazebopraspxif: Detaching target_model::target_link_1::target_link_1_collision f
rom gripper kuka_arm!!!!!!
```

Figure 32: Run 8