

Sawyer project

Part 1 A.

Poses of the targets were created as SE3 objects in Matlab using their poses and Roll-Pitch Yaw-angles from Gazebo environment. These were target1 [0.75, -0.35, 1.014, 0, 0, -0.785], target2 [1, 0, 1.014, 1, 0, 1.014] and target3 [0.85, 0.25, 1.014, 0, 0, 0.586931].

To calculate their poses with respect to robot I multiplied inverse transformation matrix of robot pose to each target pose.

sawyerTtarget1 =

1	0	0	0.75
0	1	0	-0.35
0	0	1	0.084
0	0	0	1

sawyerTtarget2 =

0.7074	0.7068	0	1
-0.7068	0.7074	0	0
0	0	1	0.084
0	0	0	1

sawyerTtarget3 =

0.8326	-0.5538	0	0.85
0.5538	0.8326	0	0.25
0	0	1	0.084
0	0	0	1

Part 1. B. To find inverse kinematics I used Serial Link class from the robotic tool box and robot FH table from the article.

I rotated each target pose around their y and z axes by 90 degrees to get gripper position.

Function ikine yielded joint angles for each position.

in_radians =

	q1	q2	q3	q4	q5	q6	q7
pos1dwn	-0.36642	-0.55917	-0.74016	0.99382	-0.78784	-1.0055	-0.28228
pos2dwn	0.016076	-0.098902	-0.84517	-0.053642	-0.60174	0.84959	-0.17939
pos3dwn	0.36526	-0.43363	-0.83871	0.8286	-0.66369	-0.82339	-0.22415
pos4dwn	1.0891	-0.22987	-1.1957	1.015	-1.262	0.23996	0.87751

in_degrees =

	q1	q2	q3	q4	q5	q6	q7
pos1dwn	-20.994	-32.038	-42.408	56.942	-45.14	-57.609	-16.173
pos2dwn	0.92109	-5.6667	-48.424	-3.0735	-34.477	48.678	-10.278
pos3dwn	20.928	-24.845	-48.055	47.475	-38.026	-47.177	-12.843
pos4dwn	62.4	-13.17	-68.511	58.157	-72.305	13.749	50.277

Part 2.

Attached run.cpp, planning.cpp and planning.h.

I was able to pass angles for each position and call the position for each target from the command line. Unfortunately, robot is dropping from the final pose and I cannot fix it (something wrong with URDF or simulator). Also, when moving from one pose to another, arm sometimes dragging on the table and wiping targets from their poses.