MEEG671 Final Project (Sujal Amrit Topno)

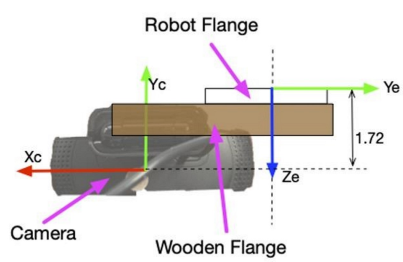
Monday, May 9, 2022   10:14 PM

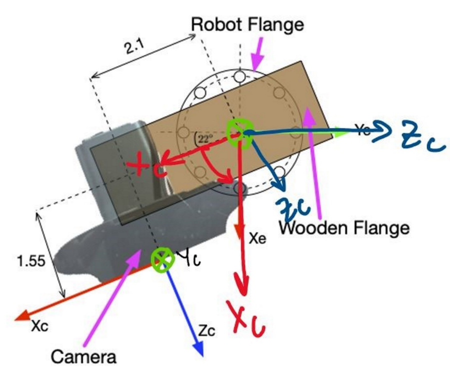
The Introduction

A 7 degree-of-freedom robot arm (*LBR iiwa 7 R800, KUKA*) is mounted on a base. A wooden flange is mounted on the robot flange. The goal of this project is to program a trajectory for the robot to position the wooden object on top of a rectangular target located in the workspace of the robot arm. The exact location and orientation of the target can be computed using an Aruco1 marker, that is printed close to the target. The robot is programmed to start from configuration q_1 (given) and needs to position the wooden object as close as possible to the target, matching its corners A, B, C, D with those of the target. During the motion, the robot should not hit in any way the surface/table the target is put on. The main idea is to design the trajectory in joint space so that the robot arm moves from its starting configuration and finishes when the wooden object is on top of the target.

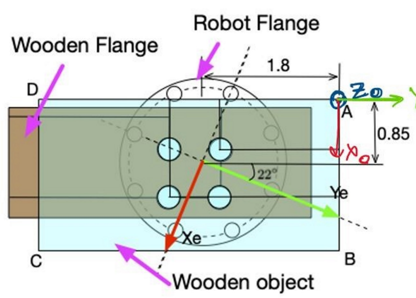
Experimental Setup

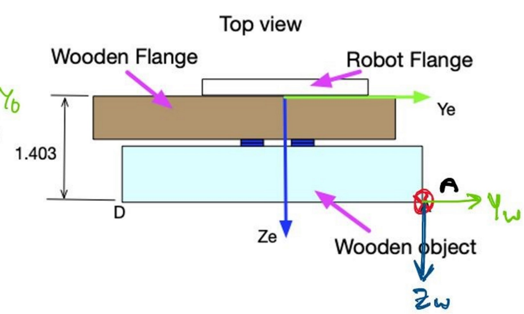
We first compute DH parameters for the robot. As per the problem statement, the camera takes a picture of the Aruco marker along with the target at a given configuration q_c. We use it compute the transformation matrix base to end-effector (T0e). Next, we compute the transformation matrix from end-effector to camera (TeC). In order for that, we first compute the inverse of TeC. This computation uses transformation as per the given dimensions and rotations about Y and X axis of camera.



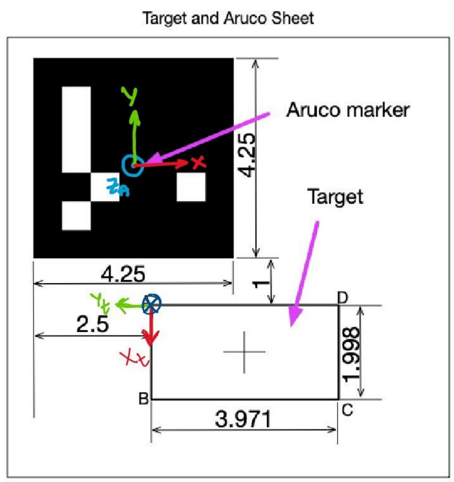


Now we compute the transformation matrix from camera to aruco marker (TCA). We use the given pose of aruco marker with respect to camera.





Next, create a frame on the point A on the wooden object and the target. First, we compute the transformation matrix of the point A with respect to the aruco marker (TAt). In the final configuration (q_f), the frame at A of the wooden object and the frame at point A of the target will be coinciding. Their transformation matrix with respect to the end effector will be the same.



Hence, we use this property of homogeneous transformation matrices to determine the transformation matrix from robot base to target (T0t).

T0t = T0e * TeC * TCA * Tat

Now, we consider the state when the robot ahs reached the target and compute the transformation matrix from end-effector to target (Tet). By the property of homogeneous transformations, we know that,

T0t = T0t * T0eF

where, T0eF denotes the transformation matrix of from the base to end-effector at the final configuration (q_f), when the wooden block has reached the target.

Next, we use inverse kinematics to get the final configuration (q_f) of the robot from the obtained transformation matrix. We maintain the joint angles within [-π, π] by adding or subtracting π from the obtained joint angles when needed.

Trajectory Planning

After obtaining the final configuration of the robot i.e. q_f we use the given initial configuration q_I to solve the following cubic polynomial with t = 5 secs.

$$q(t) = a_3t^3 + a_2t^2 + a_1t + a_0$$

Since the initial and final velocities and accelerations are zero, we use a cubic polynomial to compute a smooth trajectory between the initial and final joint angles. We use the following formulas to create a function for computing the constant values (a_3 and a_2).

$$\begin{aligned}
a_0 &= q_i \\
a_1 &= \dot{q}_i \\
a_2 &= \frac{-3(q_i - q_f) - (2\dot{q}_i + \dot{q}_f)t_f}{t_f^2} \\
a_3 &= \frac{2(q_i - q_f) + (\dot{q}_i + \dot{q}_f)t_f}{t_f^3}
\end{aligned}$$

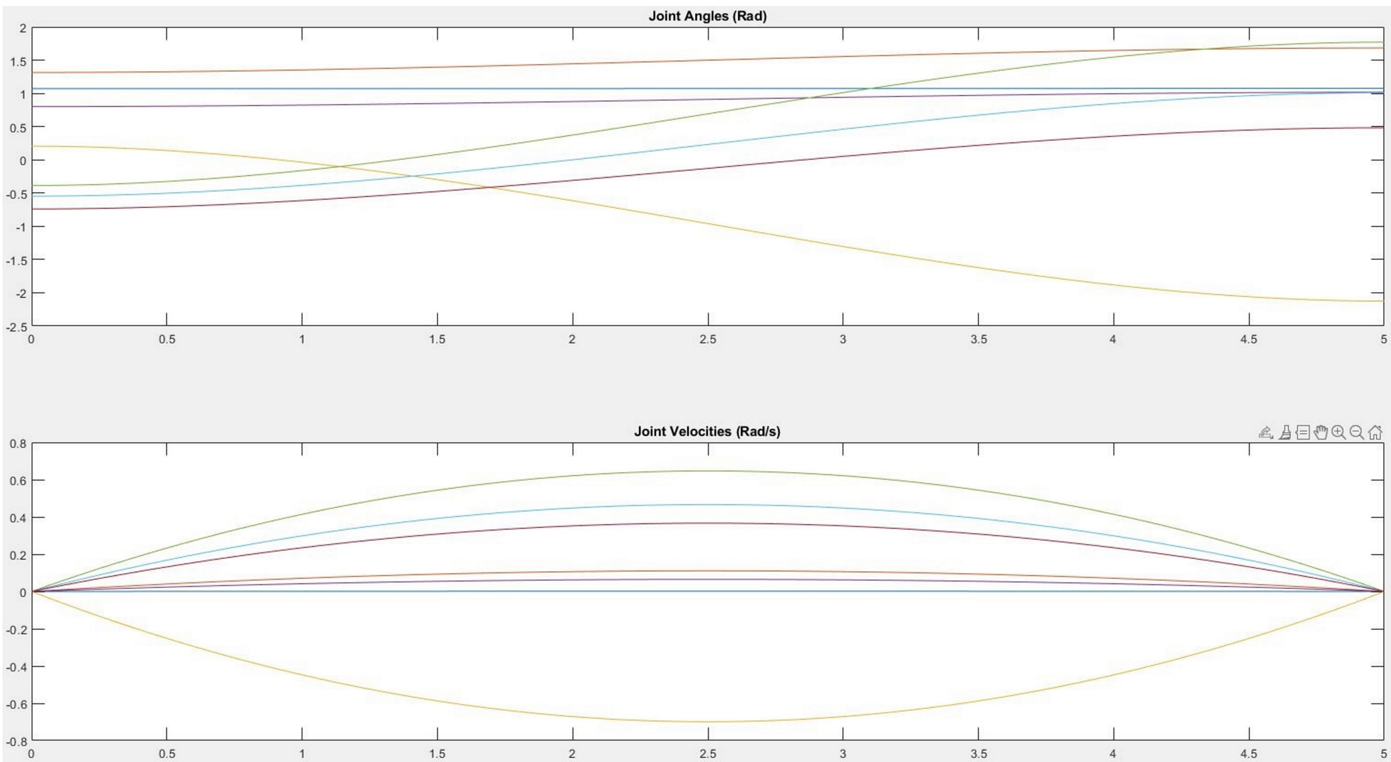
The computed a_3 and a_2 values are as follows for joints 1 to 7.

a_3 a_2

-0.0001	0.0004
-0.0059	0.0443
0.0373	-0.2797
-0.0035	0.0259
-0.0345	0.2588
-0.0249	0.1865
-0.0196	0.1466

Results

We generate the joint angle trajectory and joint velocities (by differentiating the cubic polynomial) and plot them against time. We also check if the joints have exceeded and their angular or velocity limits. Below are the plots for joints (in radians) and velocities (in radians/5ms).



Conclusion

The given problem statement can have infinite amount of solutions. There are infinite trajectories that one could create. For each of these trajectories we can have infinite configurations at each point to reach the same point. The allocated time of 30 seconds is quite a lot for the desired task. Since the robot is a redundant one, we had to come up with many trajectories to finalize the one which does not hit the table. Though traversing from point to point seems easy, it was challenging to find the right trajectory for the robot so that the links would not hit against the table. In conclusion, it is always important to check the joint trajectories to make sure that not only the end-effector but also all of the links are not hitting the obstacles.

Thank You!