MEEG671 – INTODUCTION TO ROBOTICS

Final Project Report – Muthupalaniappan Annamalai

Abstract:

A robotic arm from Kuka - LBR iiwa 7 R800 with 7 degree of freedom is mounted on a base as shown. It consists of an adapter mounted on its flange which holds a camera on one side and a rectangular shape on other side as shown. The goal of the project is to program a trajectory for the robot to position its end – effector on top of a rectangular target located in the workspace of the robot. The precise location of the target is computed by the camera using the Aruco marker placed close to the target. The trajectory design should account for any potential singularities, self-collision, and obstacles in the workspace environment.

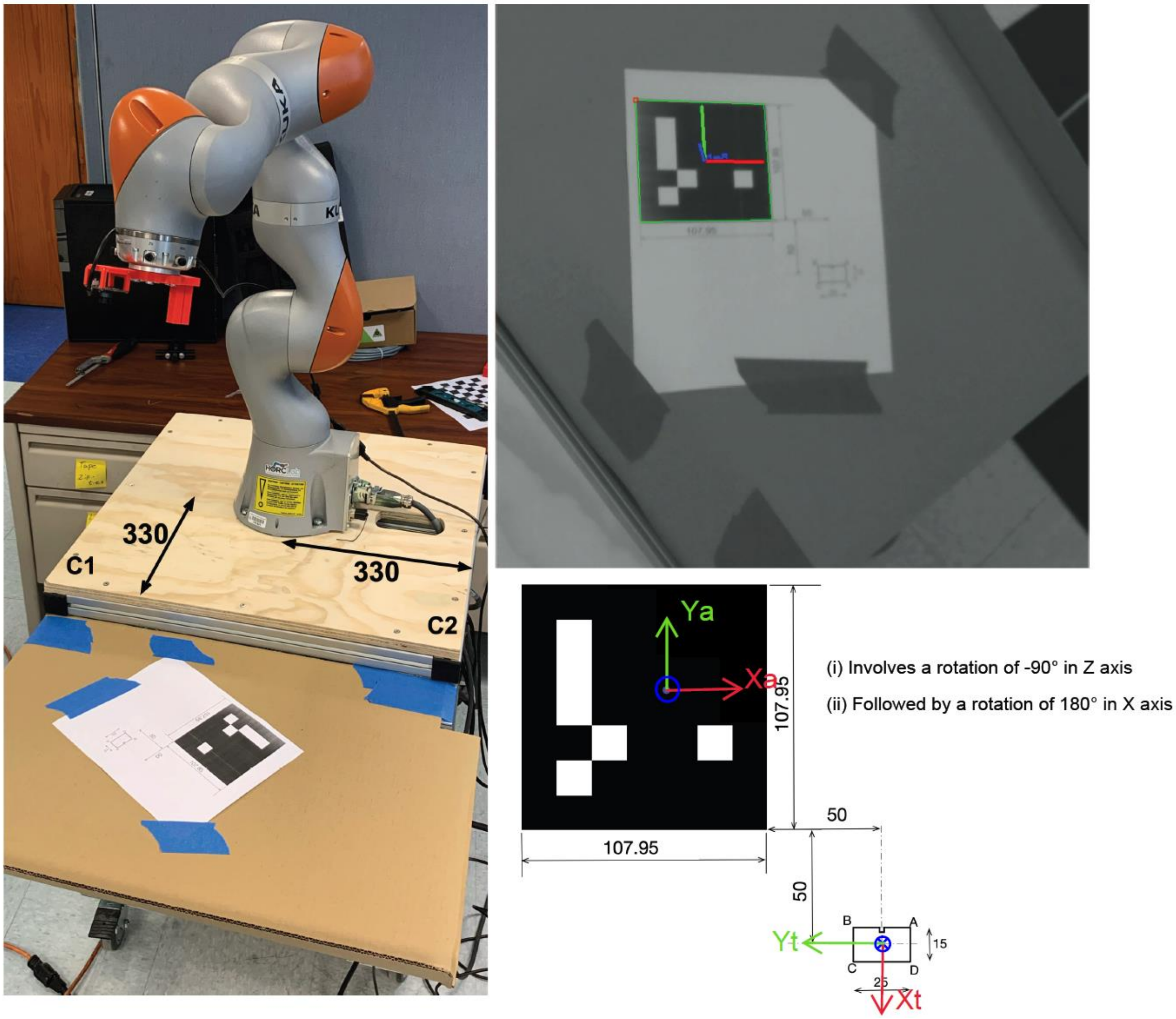
Objectives:

There are some additional constraints and objectives that the robot arm must follow during its operation in this project, they are.

- The robot must start from the given initial configuration *q1* and should position itself as close as possible with mm level precision. The trajectory design should be in joint space starting from *q1*.
- The trajectory design should ensure that the joint limits and velocity limits are not exceeded for all robot joints.
- The entire motion of the robot should last for exactly *10 seconds* in which the robot must remain stationary by holding the same configuration during its last *2 seconds* of operation.
- The final trajectory deliverable file must be in a .txt format. It must consist of the robot configuration in radians from all joints (1 to 7). The joint angles should contain up to 8 decimals with a single space between the values of each joint. Each configuration is commanded to the robot every 5 ms. Therefore, the text file should contain exactly 2000 rows and 7 columns of configuration based on the provided description above.

Setup and Methods:

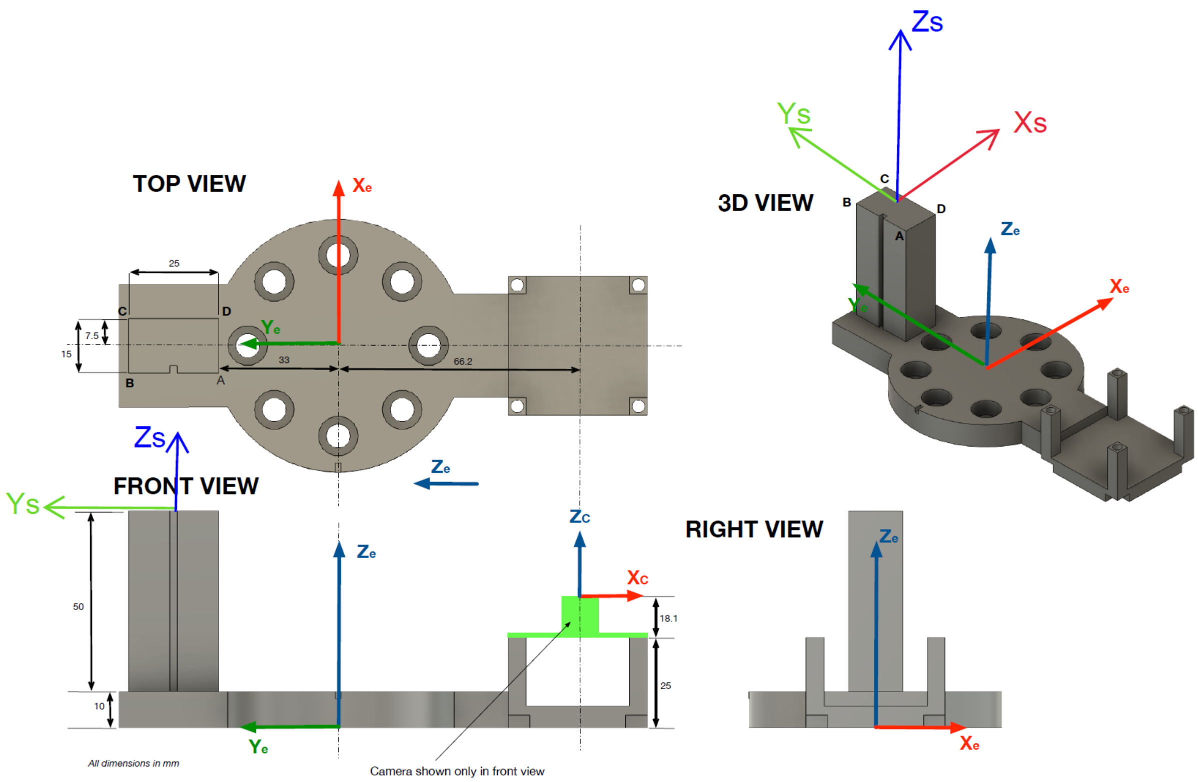
The robotic setup involves several computations of rotation and transformation matrices along with optimization algorithms. Methods like Denavit-Hartenberg, homogeneous transformation and inverse kinematics are used for the positioning of the Robotic arm on the target. Initially, we formulate the DH parameters of the robot and use the given qc configuration to compute the transformation matrix (T0e) from robot's base to end-effector. Then we calculate the transformation matrix (TeC) from the end-effector to the camera by considering the orientation and dimension of the camera. Similarly, we compute the respective transformation matrices from camera to Aruco marker (TCA), Aruco marker to target (TAt) using the provided information (distances, dimensions, position, and orientation along with respective axis frames). The reference frames ([Xt, Yt, Zt], [Xs, Ys, Zs]) are set as shown below to compute the above transformation matrices and orient the stamp on target as required.



After computing the above matrices, we now derive the homogeneous transformation matrix (T0t) from the robot's base to the target by

*T0t = T0e * TeC * TCA * TAt*

Now we compute the transformation matrix (TeS) from the end-effector to the stamp using the given dimensions, distances, and frames.



Now considering the robot has reached the target using the property of Homogeneous Transformation, we know that.

$T0e\_F(qf) * TeS = T0t$

Where T0e_F is the desired final transformation matrix from robot's base to end-effector at the final configuration (*qf*) when the stamp has reached the target as per required orientation.

Now to determine the desired final transformation matrix we use inverse kinematics and optimization algorithm to get the final configuration (*qf*) of the robot by extracting the information and requirements from the desired matrix. The joint angles are maintained in the range of [-π, π] during the process of optimization.

Trajectory Design:

After obtaining the final configuration (*qf*) of the robot. Using both initial and final config we use a 3rd order cubic polynomial to design a smooth trajectory between them. We solve them every 5ms as per the required norms. The cubic polynomial equation is.

$$q(t) = a_3 t^3 + a_2 t^2 + a_1 t + a_0$$

The below formulas are used for computing the constants (a0, a1, a2, a3) for the cubic polynomial equation. In our case for smooth trajectory, we take initial and final velocities as 0.

$$a_0 = q_i$$

$$a_1 = \dot{q}_i$$

$$a_2 = \frac{-3(q_i - q_f) - (2\dot{q}_i + \dot{q}_f)t_f}{t_f^2}$$

$$a_3 = \frac{2(q_i - q_f) + (\dot{q}_i + \dot{q}_f)t_f}{t_f^3}$$

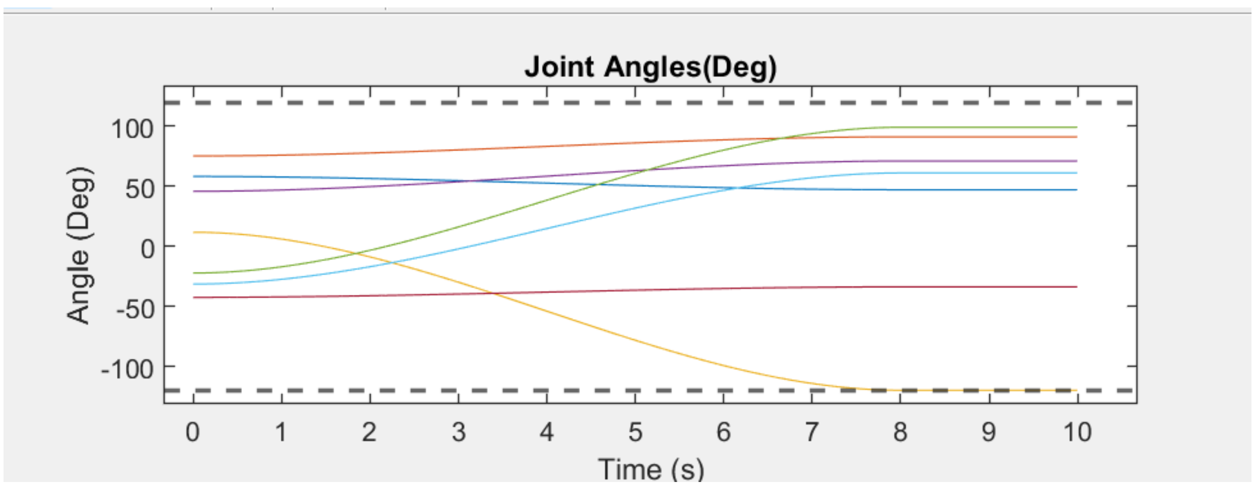
Results:

So, the final configuration of the robot is obtained use the above methods and the following trajectory is designed and verified. The final configuration is.

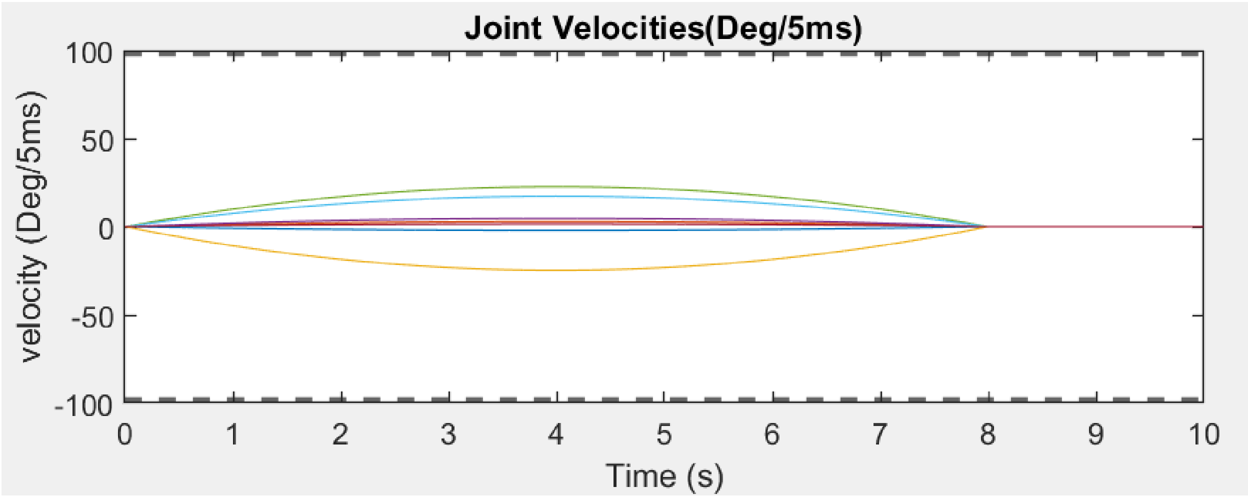
```
Computed Configuration (q values in deg)
   47.1897   91.2336 -119.7025   71.1001   99.1841   61.1818  -33.6306
```

Verification of Results:

The obtained trajectory is written in a text file named Muthupalaniappan Annamalai containing 2000 lines of the configuration of the robot. The trajectory and other objectives of the project are verified by checking whether it exceeds any joint/velocity limits and by software simulation in a similar environment. This is checked for all the joints. We can also verify it by plotting the values and checking whether they are within the limits specified by the Robot manufacturer to prevent any accidents or damage of the robots. The below image is obtained by plotting our velocity and configuration.

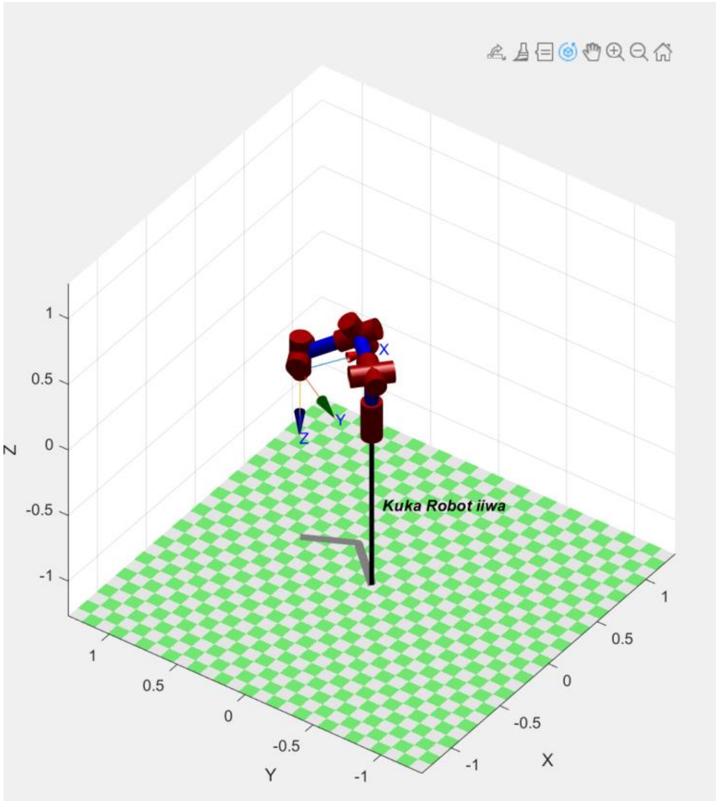


I plotted the **lowest limit** of *all joints* (+120, -120) and then checked whether any joints exceeded the limit. We could see that *joint 3* is very near to the lower limit (-120). But the actual joint limit is (+170, -170). So, our configuration and trajectory are safe.



Here I did the same, took the lowest velocity limits of the joints and plotted it to check whether it exceeds the limits. We can see a smooth trajectory which satisfies all the requirements.

I also wrote a script using Robot toolbox and checked the simulation of my trajectory and verified that there is not self-collision or obstacle collision. The name of the file is "*simulation.m*". I have attached it along with the other files.



Conclusion:

Therefore, the trajectory design is completed as per the requirements and satisfies all the required objectives.