Abhijeet M. Kulkarni
UD ID: 702525838

Robot

Figure

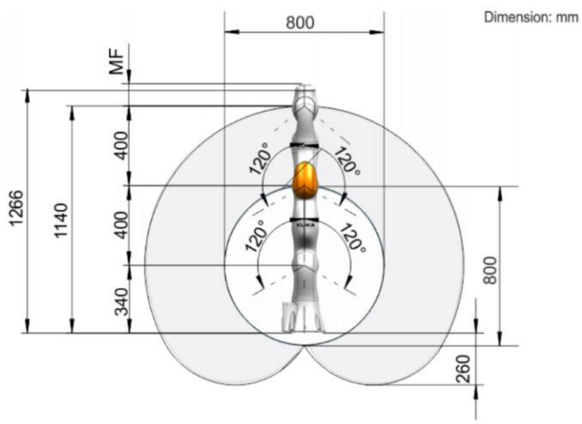


Figure 1 LBR iiwa 7 R800 Side View

Joint	Limits
A1	±170
A2	±120
A3	±170
A4	±120
A5	±170
A6	±120
A7	±175

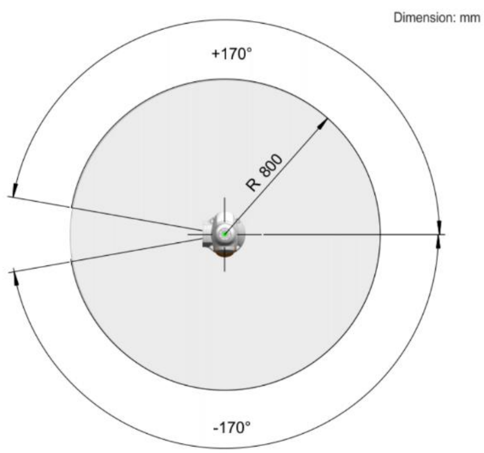
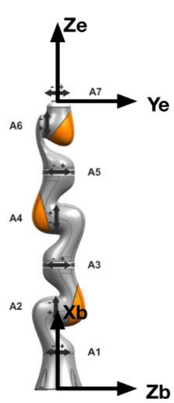


Figure 2 LBR iiwa 7 R800 Top View

Joint	Max Speed
A1	98 °/s
A2	98 °/s
A3	100 °/s
A4	130 °/s
A5	140 °/s
A6	180 °/s
A7	180 °/s

DH Parameters



Transformation	***d***(m)	**θ**	***a***	**α**
0 → 1	0.34	q1	0	-90°
1 → 2	0	q2	0	90°
2 → 3	0.4	q3	0	90°
3 → 4	0	q4	0	-90°
4 → 5	0.4	q5	0	-90°
5 → 6	0	q6	0	90°
6 → e	0.126	q7	0	0°

Forward Kinematics

The homogenous transformation matrix is given by,

$$T_e^b = T_0^b T_1^0 T_2^1 T_3^2 T_4^3 T_5^4 T_6^5 T_e^6$$

Where,

$$T_0^b = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Problem statement

The starting point and ending points are given by p_A and p_B and are given by,

$$p_A = \begin{bmatrix} 0.9099 \\ 0.3576 \\ 0.4660 \end{bmatrix} \text{(m)}, \quad p_B = \begin{bmatrix} 0.9099 \\ -0.4189 \\ 0.4660 \end{bmatrix} \text{(m)}.$$

Initial angular configuration is at point A is given by,

$$q_A = [58.2686 \quad 75.3224 \quad 11.7986 \quad 45.9029 \quad -22.1081 \quad -31.2831 \quad -42.3716]^T \text{(deg)}$$

The objective is to calculate the joint configurations at 5ms such that end effector is moved to point B with its initial position at A. The robot should move while satisfying following constraints,

- Final position error should be less than 1mm and its orientation should not vary by 2 deg during the motion.
- The robot must not touch the obstacle at any time.
- Joint angles must be kept within the limits.
- Joint angular speeds should not exceed maximum.

The obstacle's corner positions are,

a(0.65, 0.20, 0.92); b(0.65, -0.20, 0.92); c(0.65, -0.20, 0.42); d(0.65, 0.20, 0.42); e(-0.50, 0.20, 0.92)

f(-0.50, -0.20, 0.92); g(-0.50, -0.20, 0.42)& h(-0.50, 0.20, 0.42)

Trajectory Planning and Inverse Kinematics

In this project, the problem of trajectory planning and inverse kinematics is done simultaneously. The robot trajectory is given by the trajectory of the solution in the inverse kinematics iterations. Inverse kinematics gives us the joint configuration of the desired pose and it is done while optimizing the null space of the joint space, which is based on the secondary objective functions. Secondary objective functions try to maximize – minimum distance from obstacle and robot, joints from their limits, joints and the sum of distances of joints from the obstacle's center.

For inverse kinematics, a modified jacobian inverse Algorithm is used, which calculates the joint configuration by pushing the solution to the local minimum. The modification is done to limit the maximum Δq under the maximum joint speed, as, for this project, trajectory planning and inverse kinematics is done simultaneously. If the any of joint crosses the maximum speed limit, the Δq is then normalized to get a unit norm and then scaled to maximum joint velocity. This allows us to obtain trajectory which reaches its final position while satisfying constraints.

The update formula is given by,

$$q_{k+1} = \begin{cases} q_k + \Delta q, & \max \Delta q < |\dot{q}_{max}| \\ q_k + |\dot{q}_{max}| \dfrac{\Delta q}{\|\Delta q\|_2}, & \max \Delta q \geq |\dot{q}_{max}| \end{cases}$$

Here, $\dot{q}_{max}$ is the maximum joint speed(scalar) and Δq(vector) is given by,

$$\Delta q = K J_A^{+}\left(x_d - x_{e_k}\right) + (I_n - J_A^{+} J_A)\Delta q_0$$

Here, $J_A^{+} = J_A^T (J_A J_A^T)^{-1}$ is the pseudoinverse of the analytical jacobian, and Δq_0 (vector) is given by ,

$$\Delta q_0 = \sum k_i \frac{\partial w_i(q)}{\partial q}$$

Secondary objective functions

Joint limits

$$w(q) = -\frac{1}{2(n=7)} \sum_{i=1}^{n=7} \left(\frac{q_{i_k} - \bar{q_i}}{q_{iM} - q_{im}} \right)^2$$

q_{im} is lower joint limit; q_{iM} is upper joint limit; $\bar{q_i}$ is mid of limits & q_{i_k} is the angle at kth step

Obstacle Avoidance

This objective function tries to maximize the distance between the closest modelled points of the robot and the obstacle by optimizing the null space of the joint space. The robot is modelled as points at the joints. The obstacle is modelled as points on its surface.

$$w(q) = \begin{cases} 0, & d_{closest} > D_{influence} \\ \dfrac{\left\| P_{closest\ obs} - P_{closest\ joint}\ (q) \right\|_2}{|D_{min} - d_{closest}|}, & d_{closest} \leq D_{influence} \end{cases}$$

Here, $D_{influence}$ is the distance where the influence of the obstacle starts, $d_{closest}$ is the closest distance between the robot and the obstacle, D_{min} is the minimum allowable distance from the center of the joint, $P_{closest\ obs}$ position of the closest obstacle and $P_{closest\ joint}\ (q)$ is the position of the closest joint position as a function of q.

Distance maintaining

This objective function pushes the joint position away from the centroid of the obstacle. This is useful to reorient the robot in the initial position. This function's effect on the orientation is not as effective as the obstacle avoidance function but is useful when the robot is not in influence distance of the obstacle.

$$w(q) = \sum_{i=1}^{7} \left\| P_{centroid} - P_{i^{th}\,joint} \right\|$$

Where, $P_{centroid}$ is the centroid of the upper part of the obstacle.

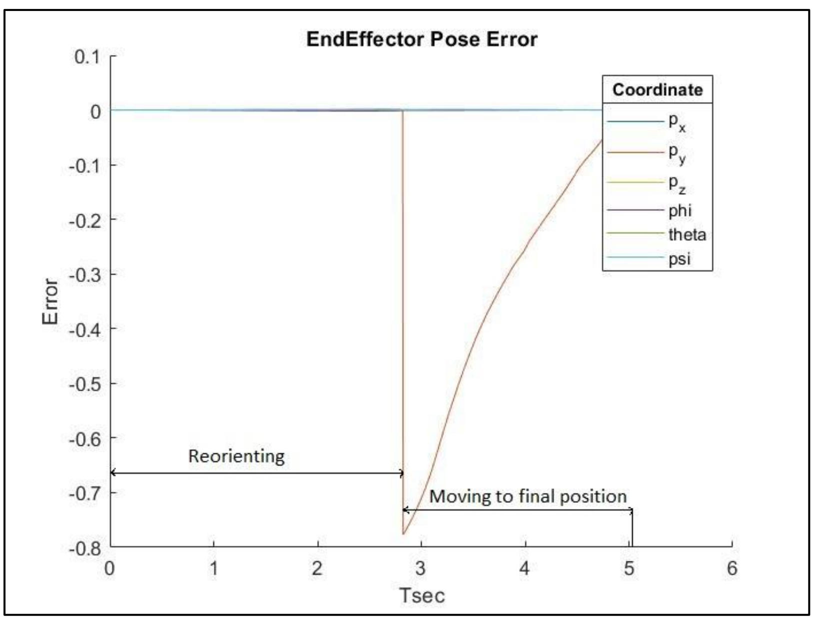
Procedure

Trajectory generation is done in two steps,

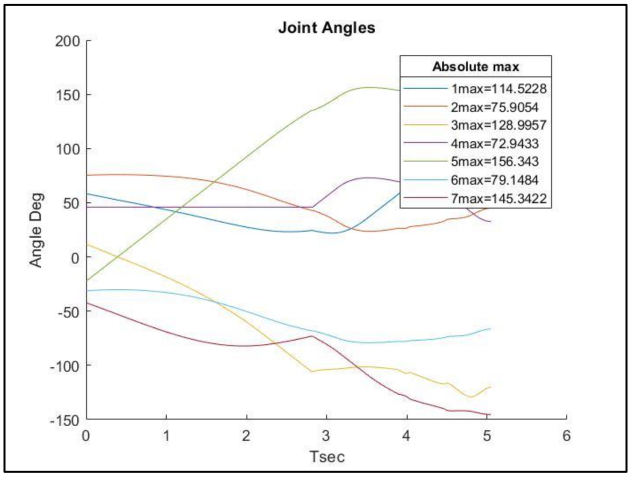
- Let the robot reorient at the initial position, A, by optimizing the null space of the joint space until the change in joint configuration at each iteration is less than a threshold, while maintaining the end effector pose.
- Now give the desired point as B and iterate till the error in the pose of end effector is less than a threshold.

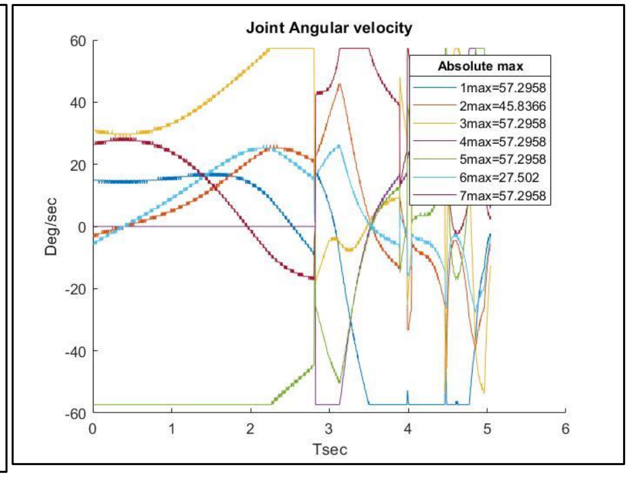
Results

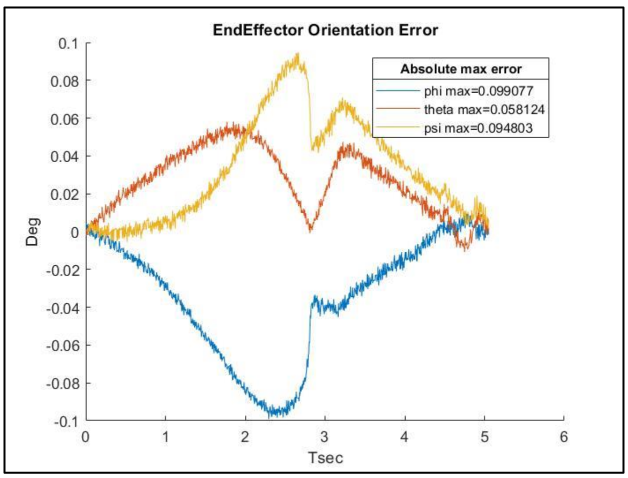
- Time to final position = 5.05sec
- Maximum Joint angle = 156.343°
- Maximum Joint speed = 57.2958°/sec
- Maximum End effector orientation error = 0.099097°

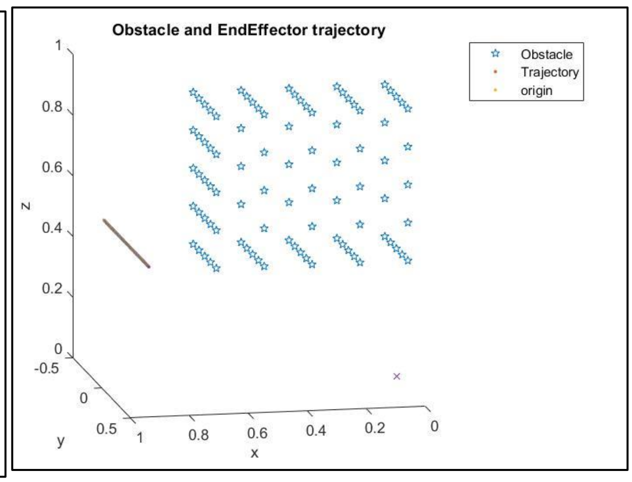


The first reorienting step last till the Δq is above some threshold. The distance maintaining objective function is the main influencer at this point. For the second step, the desired position is set to B and the iterations lasts the pose error is reduced to some threshold, i.e., the maximum allowed errors.









In Obstacle and EndEffector trajectory figure, the blue stars denote the modelled obstacle (only half of it is modelled as lower is below the base of robot). The blackish line denotes the path followed by the end effector. And x is the base of the robot in the plot.

END