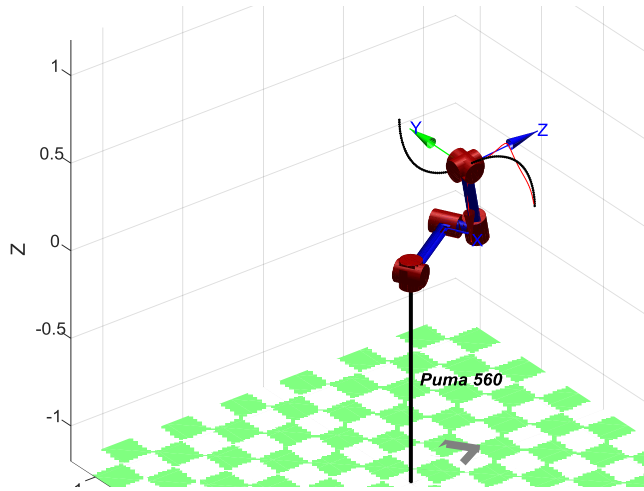
Lab 9 Exercises



1 Resolved Motion Rate Control in 6DOF

(Download **Lab9Question1Skeleton.m** from UTSOnline)

1.1 Set parameters for the simulation

1.2 Allocate array data

1.3 Set up trajectory, initial pose

1.4 Track the trajectory with RMRC

1.5 Plot the results

1.6 Consider and discuss the following questions:

- Does the robot successfully track the trajectory? Why or why not?
- Is the robot hitting singularities?
- Are the joints losing control?
- Is Damped Least Squares applied sufficiently?
- Is the trajectory error too big?
- How could you attenuate the damping coefficient to fix this?
- Does the robot hit joint limits?
- How might the initial guess of joint angles when solving inverse kinematics affect this?

2 Static Torque

(Download **Lab9Question2Skeleton.m** from UTSOnline)

2.1 Load a model of the Puma560 robot. Then find the maximum static load (kg) that the Puma 560 robot can sustain at the joint configuration:

$$\mathbf{q} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^{\mathrm{T}}$$

Note that the maximum joint torques are given by:

$$\boldsymbol{\tau}_{\max} = [97.6 \quad 186.4 \quad 89.4 \quad 24.2 \quad 20.1 \quad 21.3]^{\mathrm{T}}$$

2.2 For the following end-effector pose

$$\mathbf{T} = \begin{bmatrix} 0 & 0 & 1 & 0.7 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

calculate an inverse kinematics solution. Then, determine the maximum static load (kg) that can be supported by the Puma560 in this configuration.

2.3 Assume that we have a mass of 40kg mounted on a frictionless surface at x = 0.8m.

$$\mathbf{T} = \begin{bmatrix} 0 & 0 & 1 & \mathrm{x} \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

What is the smallest distance of x that we can draw the object back to so that the Puma560 can overcome the static torque? (*Hint: There is more than 1 joint configuration that can achieve the same end-effector pose!)*

3 Dynamic Torque

(Download **Lab9Question3Skeleton.m** from UTSOnline)

The Puma 560 robot is required to lift and transport a known mass of 21kg between two points. It is offset from the end-effector frame by 0.1m in the x-direction.

The function `p560.payload(mass, [x,y,z])` will alter the dynamics of the Puma 560 model to incorporate the gravitational forces and inertia.

Your task is to find the fastest time in which the payload can be transported between two given transforms $\mathbf{T}_1$ and $\mathbf{T}_2$.

3.1 Use inverse kinematics to interpolate between the following two end-effector transforms:

$$\mathbf{T}_1 = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0.7 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \mathbf{T}_2 = \begin{bmatrix} 0 & 0 & 1 & 0.5 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0.6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3.2 Solve the necessary joint acceleration at each time-step needed to move the Puma560 to the next set of joint angles

$$\ddot{\mathbf{q}}(\mathrm{t}) = \mathbf{f}(\mathbf{q}(\mathrm{t}), \mathbf{q}(\mathrm{t}+\Delta\mathrm{t}), \dot{\mathbf{q}}(\mathrm{t}), \Delta\mathrm{t})$$

3.3 Use Robot Toolbox to calculate the inertia, Coriolis, and gravitational torques at the current joint configuration. Note that the Coriolis will be given as a matrix, not a vector.

$$\mathbf{M}(\mathbf{q}), \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}), \mathbf{g}(\mathbf{q})$$

3.4 Solve the dynamics equation to get the required torque to move the joints. (You don't need to calculate the static torque here, since the p560.payload() function has incorporated it for us).

$$\boldsymbol{\tau} = \mathbf{f}(\mathbf{M}, \mathbf{C}, \mathbf{g}, \mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$$

3.5 Check that the calculated torques are within torque limits. If not, cap the joint torque.

3.6 Recalculate the resultant joint accelerations based on the capped joint torques. (Reverse the equation you used in 3.3, and solve for the joint accelerations)

3.7 Update the joint angles for the *next* time step, based on the acceleration.

3.8 Update the joint velocities for the *next* time step, based on the acceleration.