Human Neuromechanical Control and Learning
Tutorial 3: Dynamics and Control of Planar Arm Movements

Completed by: Jenna Kelly Luchak
CID: 01429938
February 15, 2018

Question 1

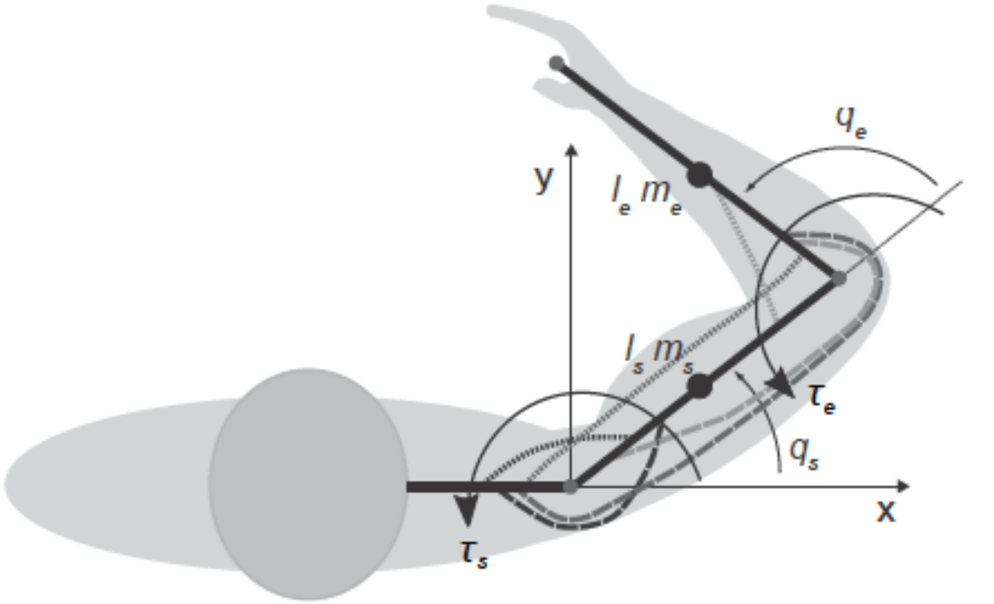


Figure 1: A 2 joint system of planar arm movement. The angle convention where q_e q_s are the elbow and shoulder angles respectively, and l_e l_s are the distances of the forearm and the upper arm respectively. The torques applied to the elbow joint and shoulder joint are identified as τ_e τ_s respectively. The distance of the elbow joint and shoulder joint to its corresponding center of mass is denoted by l_{me} l_{ms} respectively.

Part A.1

For the first part of this question, the shoulder joint is blocked; therefore I assume that the angular velocity and angular acceleration of the shoulder joint is negligible.

$$\dot{q}_s = 0 \quad \text{and} \quad \ddot{q}_s = 0$$

The general dynamic equation of a two joint model is derived as follows.

$$\tau = H(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \psi(q,\dot{q},\ddot{q}) * p$$

The terms of the above equation are defined as follows.

$H(q)$ is the mass or inertia matrix and equals $H(q) = \begin{bmatrix} H_{ss} & H_{se} \\ H_{es} & H_{ee} \end{bmatrix}$

$C(q,\dot{q})\dot{q}$ is the velocity dependent forces.

$$C(q,\dot{q})\dot{q} = \begin{bmatrix} -M_e l_s l_{me} \dot{q}_e (2\dot{q}_s + \dot{q}_e) sin(q_e) \\ M_e l_s l_{me} \dot{q}_s^2 sin(q_e) \end{bmatrix}$$

$G(q)$ is gravity.

φ is the angular space and is

$$\varphi(q,\dot{q},\ddot{q}) = \begin{bmatrix} \ddot{q}_s + \ddot{q}_e & (2\ddot{q}_s + \ddot{q}_e)cos(q_e) - \dot{q}_e(2\dot{q}_s + \dot{q}_e)sin(q_e) & \ddot{q}_s \\ \ddot{q}_s + \ddot{q}_e & (\ddot{q}_s)cos(q_e) + \dot{q}_s^2 sin(q_e) & 0 \end{bmatrix}$$

p is the moment of inertia at each joint and is $p = \begin{bmatrix} J_e + m_e l_{me}^2 \\ m_e l_s l_{me} \\ J_s + m_s l_{ms}^2 + m_e l_s^2 \end{bmatrix} = \begin{bmatrix} 0.1004 kgm^2 \\ 0.12 kgm^2 \\ 0.263 kgm^2 \end{bmatrix}$

τ is the torque in the shoulder and elbow joint and equals $\tau = \begin{bmatrix} \tau_s \\ \tau_e \end{bmatrix}$

Therefore, using matrix multiplication and the assumption about our blocked shoulder joint, the dynamic equation of the elbow single-joint movement can be written as follows.

$$\tau = \begin{bmatrix} (\dot{q}_s + \dot{q}_e)(0.1004) + \left((2\ddot{q}_s + \ddot{q}_e)cos(q_e) - \dot{q}_e(2\dot{q}_s + \dot{q}_e)sin(q_e)\right)(0.12) + (\ddot{q}_s)(0.263) \\ (\dot{q}_s + \dot{q}_e)(0.1004) + \left((\ddot{q}_s)cos(q_e) + \dot{q}_s^2 sin(q_e)\right)(0.12) \end{bmatrix}$$

Since $\dot{q}_s = 0$ and $\ddot{q}_s = 0$ it can be simplified to as follows.

$$\tau = \begin{bmatrix} \tau_s \\ \tau_e \end{bmatrix} = \begin{bmatrix} \dot{q}_e(0.1004) + 0.12\ddot{q}_e cos(q_e) - 0.12\dot{q}_e^2 sin(q_e) \\ \dot{q}_e(0.1004) \end{bmatrix}$$

Based on the torque equation directly above, the resulting dynamics equation of the elbow single-joint movement is both linear and non-linear. The torque in the shoulder joint contains a squared element therefore the dynamics of the shoulder joint is non-linear; however, the torque in the elbow joints dynamics is linear because the equation is first-order.

Part A.2

The angular trajectory of an elbow joint that has a blocked shoulder joint was determined through MATLAB and the specifications of the model are as follows. The initial angles of the shoulder and elbow joints were select to be 0° and 30° respectively. The torque equations for the shoulder, τ_s and elbow, τ_e joint are presented below.

$$\tau_s(t) = \begin{Bmatrix} 0 & 0 \le t \le 2 \\ 0 & 2 \le t \le 20 \end{Bmatrix}$$

$$\tau_e(t) = \begin{Bmatrix} 0.02 - 0.1\dot{q}_e(t) & 0 \le t \le 2 \\ -0.1\dot{q}_e(t) & 2 \le t \le 20 \end{Bmatrix}$$

A plot of the elbows angular trajectory with respect to time is presented below in Figure 2.

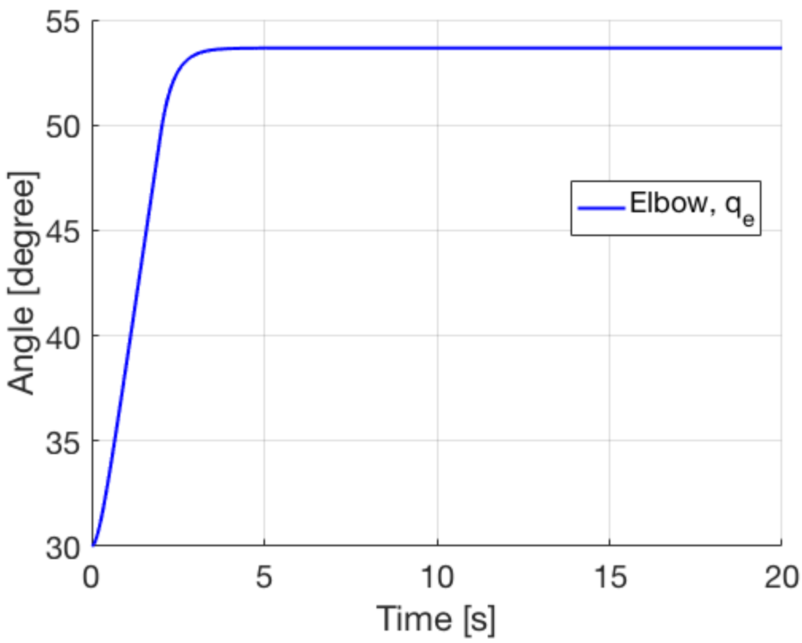


Figure 2: The trajectory of an elbows angular position, q_e given that the shoulder is blocked, with respect to time using a sampling time of 0.01s.

From Figure 2, it can be seen that the angular trajectory of the elbow joint starts at 30° and linearly increases for 3 seconds, until it plateaus at approximately 54° for the remainder of the movement.

Part B

The angular trajectory of an elbow joint that has a freely moving shoulder joint was determined similarly to Part A above; however, there are key differences in the specifications of the model, which are presented as follows. The initial angles of the shoulder and elbow joints remained consistent as 0° and 30° respectively. The torque equations for the shoulder, τ_s are no longer zero, and the elbow, τ_e joint remained unchanged, as presented below.

$$\tau_s(t) = \{-0.1\dot{q}_s(t) \quad 0 \leq t \leq 20 \}$$

$$\tau_e(t) = \begin{Bmatrix} 0.02 - 0.1\dot{q}_e(t) & 0 \leq t < 2 \\ -0.1\dot{q}_e(t) & 2 \leq t \leq 20 \end{Bmatrix}$$

A plot of various angular trajectories for the shoulder and elbow joint with respect to time is presented below in Figure 3.

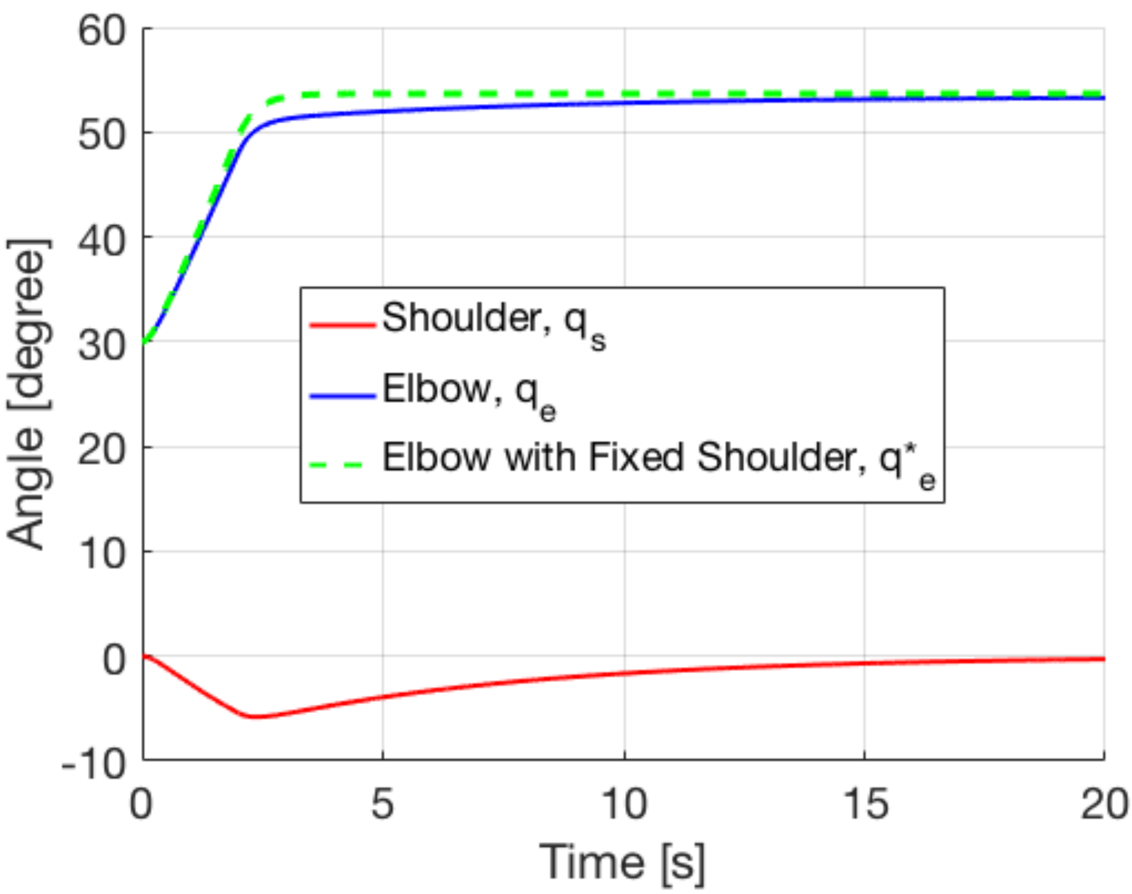


Figure 3: The trajectory of a shoulders angular position, q_s and elbows angular position, q_e when the shoulder is free to move with respect to time, compared against the same elbows angular position with respect to time when the shoulder joint is fixed, q_e^*.

As seen in Figure 3 above, the joint trajectory of an elbow follows a similar curve whether the shoulder joint is blocked or not blocked. There is minimal difference between the two elbow trajectories in Figure 3; however, one noticeable difference is that although both curves plateau at approximately 53˚, the trajectory of the elbow with a fixed shoulder reaches that plateaus sooner than the elbow with a freely moving shoulder. A possible reason for these similar curves is due to the redundancy of the arm. The arm has 7 degrees of freedom, which means that you can move the arm or complete a task via a variety of different movements. When you move your arm to reach for an object directly in front of you, your wrist and elbows move to get into position, but your shoulder remains almost motion less, therefore having a fixed shoulder does not drastically effect your movements.

Question 2

Part A

To investigate the non-linear dynamics and linear feedback control of an arm, a PD controller was used at each joint to represent the actual movement of the arm and those results were compared against a planned hand trajectory. The planned hand trajectory equations are presented as follows.

$$x^*(t_n) = -0.2605 + 0.11g\left(\frac{t}{T}\right)$$

$$y^*(t_n) = 0.0915 + 0.5g\left(\frac{t}{T}\right)$$

Where, $g(t_n) = t_n^3(6t_n^2 - 15t_n + 10)$

The model for the actual trajectory had the following specifications. The initial angles of shoulder and elbow joints were 90˚ and 130˚ respectively. The control gains were $K_p = 100Nm/rad$ and $K_d = 100N\ ms/rad$ and the torque equation, is presented below.

$$\tau = K_p(q_{ref} - q) + K_d(q_{ref} - q)$$

Where, q represents the actual angles and q_{ref} represents the desired angles.

Figure 4 and 5 below present the graphical results of this investigation.

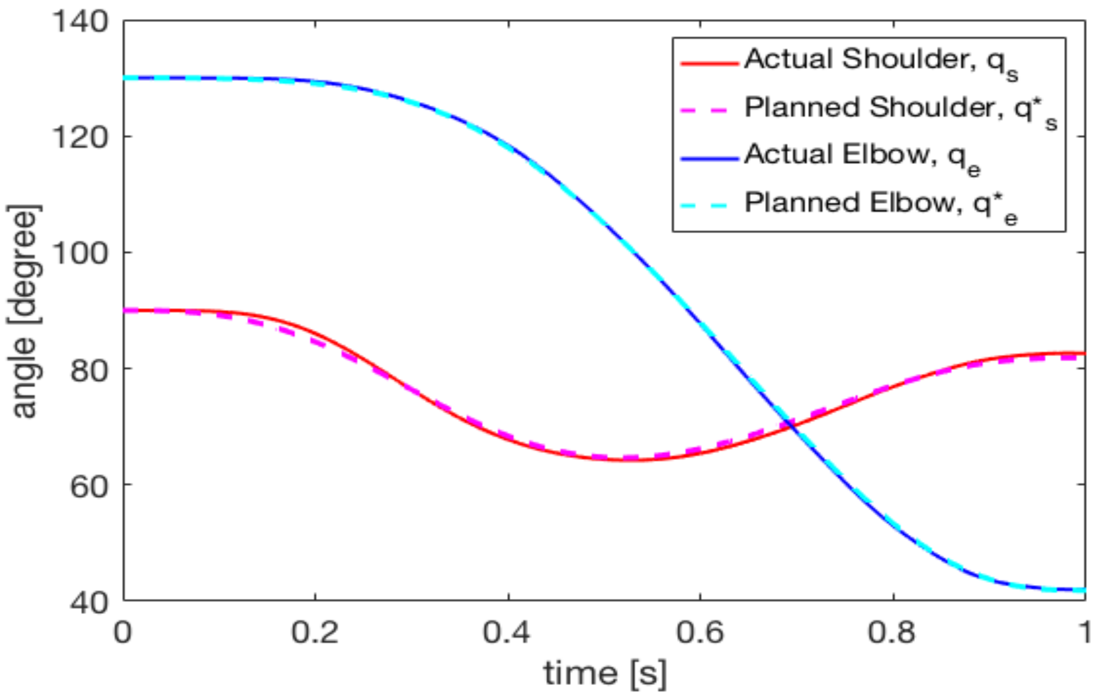


Figure 4: Actual Trajectories of the shoulder joint, q_s and elbow joint, q_e with respect to time, compared against the planned trajectories of the shoulder and elbow joints, q_s^* and q_e^* respectively. Results found using Kp = 100Nm/rad, Kd = 10Nms/rad and T=1s.

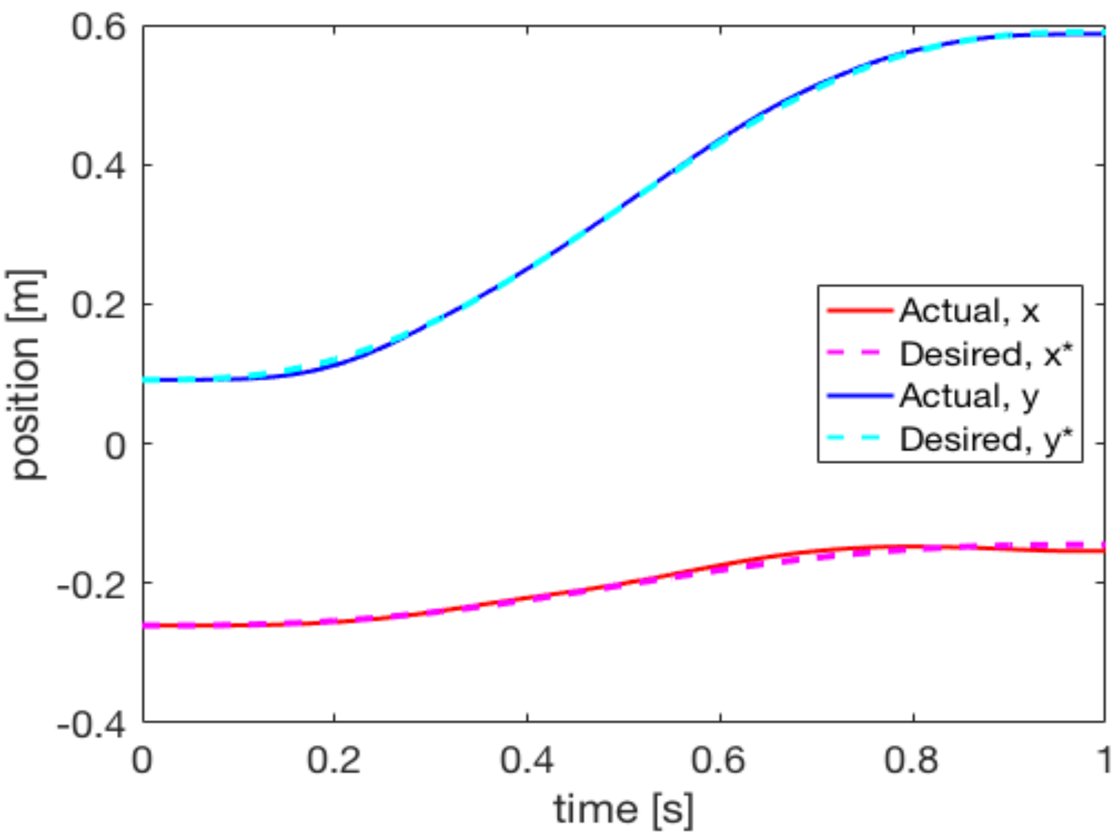
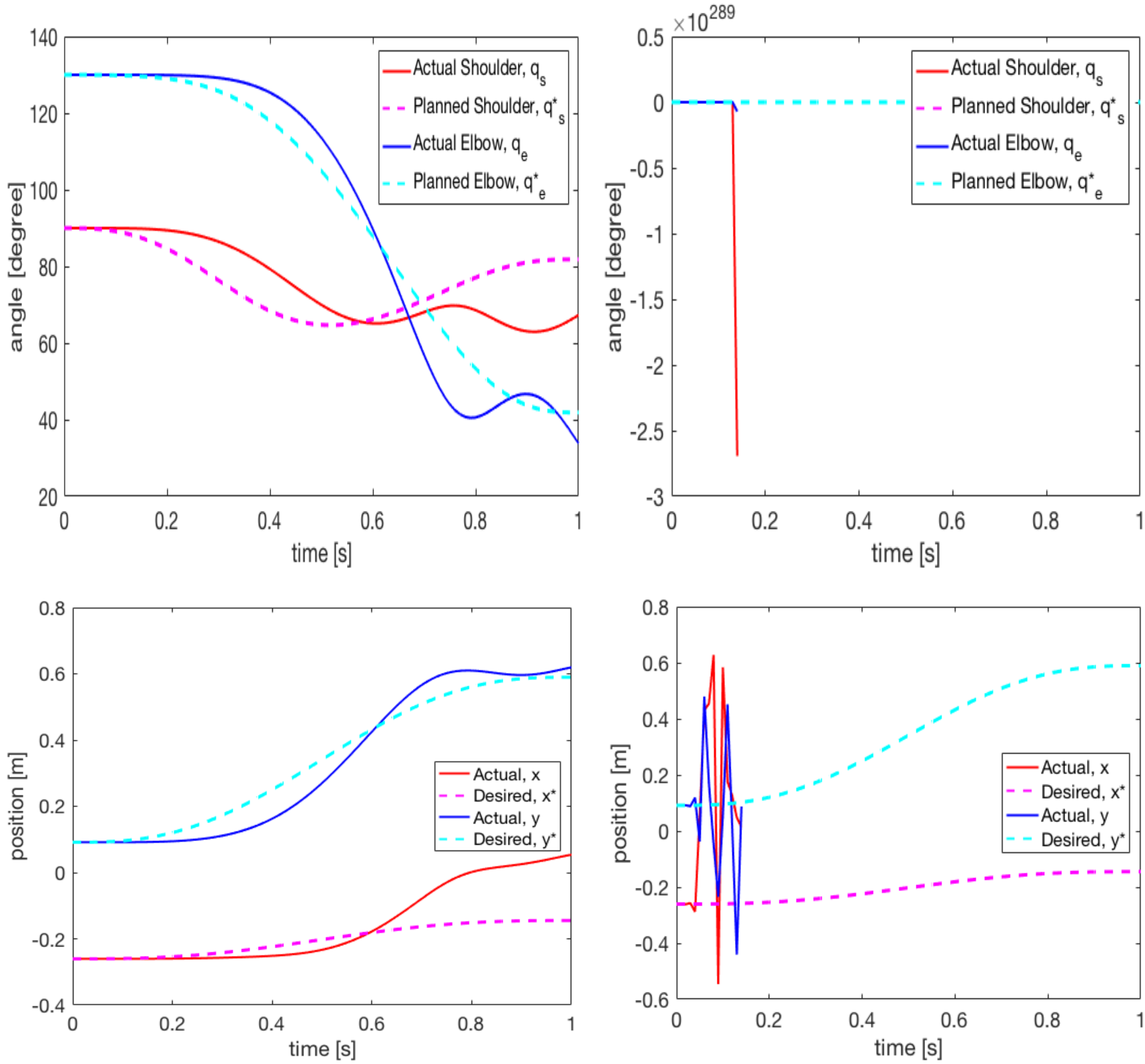


Figure 5: Resultant Hand Trajectory in the x and y directions with respect to time compared against the desired hand trajectory of the same arm. Results found using Kp = 100Nm/rad, Kd = 10Nms/rad and T=1s.

Based on the results from Figures 4 and 5, the curves from the actual angular and positional trajectories of the arm agree well with the desired/planned trajectories. From this result, we can infer that using a PD controller with linear feedback control is an accurate method to model nonlinear dynamics.

Part B

When the derivative and peripheral gains of the PD controller are altered, the resulting control of the arm is compromised. First I kept Kd gain constant at 10Nms/rad and changed the Kp gain to see how the trajectories were affected. The graphs from this investigation are presented below in Figure 6.



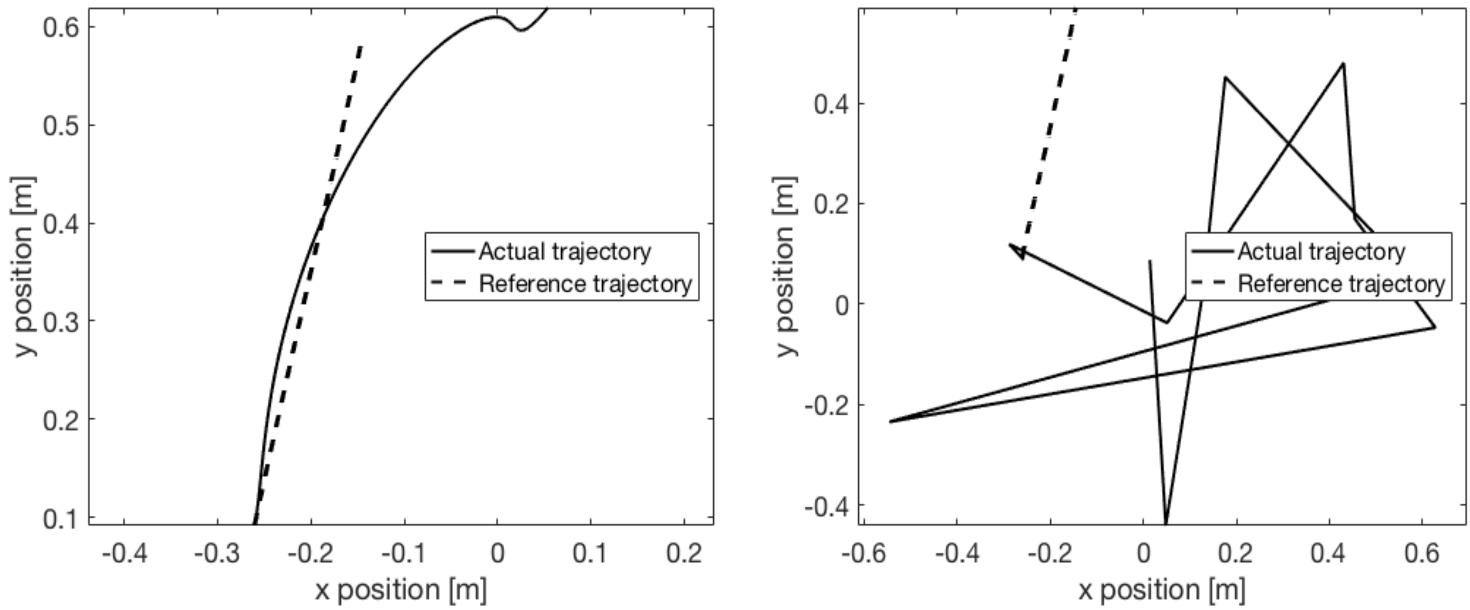


Figure 6: Trajectory graphs of an arm under PD control. The first row shows angular position with respect to time, the second row presents positional trajectory with respect to time and the last row is the hands trajectory. [Left Column] Properties of the PD controller are Kp = 0 Nm/rad and Kd = 10 Nms/rad. [Right Column] PD controller properties are Kp = 10,000 Nm/rad and Kd = 10 Nms/rad.

From the results of Figure 6 it was found that as you decreased Kp, the accuracy of the actual trajectory curves decreases, as is evident by the oscillations in the left column of Figure 6. The angular and positional trajectories of the actual curves no longer agree with the desired curves or the trajectories as seen in Figure 4 and 5 above when Kp = 100Nm/rad. Alternatively, as you increase the Kp gain while keeping Kd constant at 10Nms/rad, the curves accuracy increases and agrees almost perfectly with the desired curves; however, if you increase Kp to a very large value such as 10,000 Nm/rad, then the angular trajectory reach's a singularity and as a result, the solutions become inaccurate, as can be seen by sharp spikes and abnormal graphs in the right column of Figure 6 below.

Secondly, I kept Kd gain constant at 10Nms/rad and changed the Kp gain to see how the trajectories were affected. The graphs from this investigation can be seen below in Figure 7.

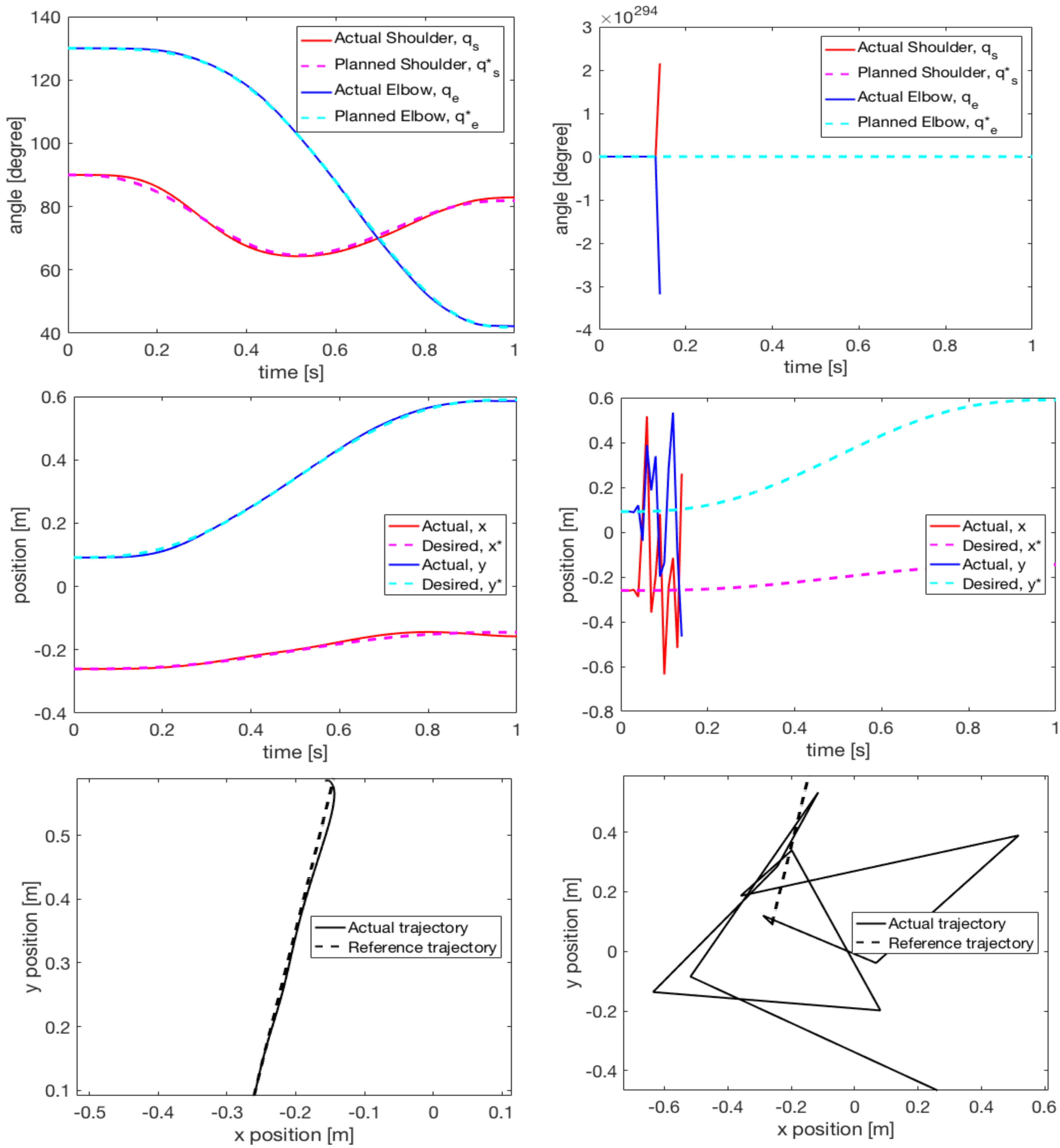
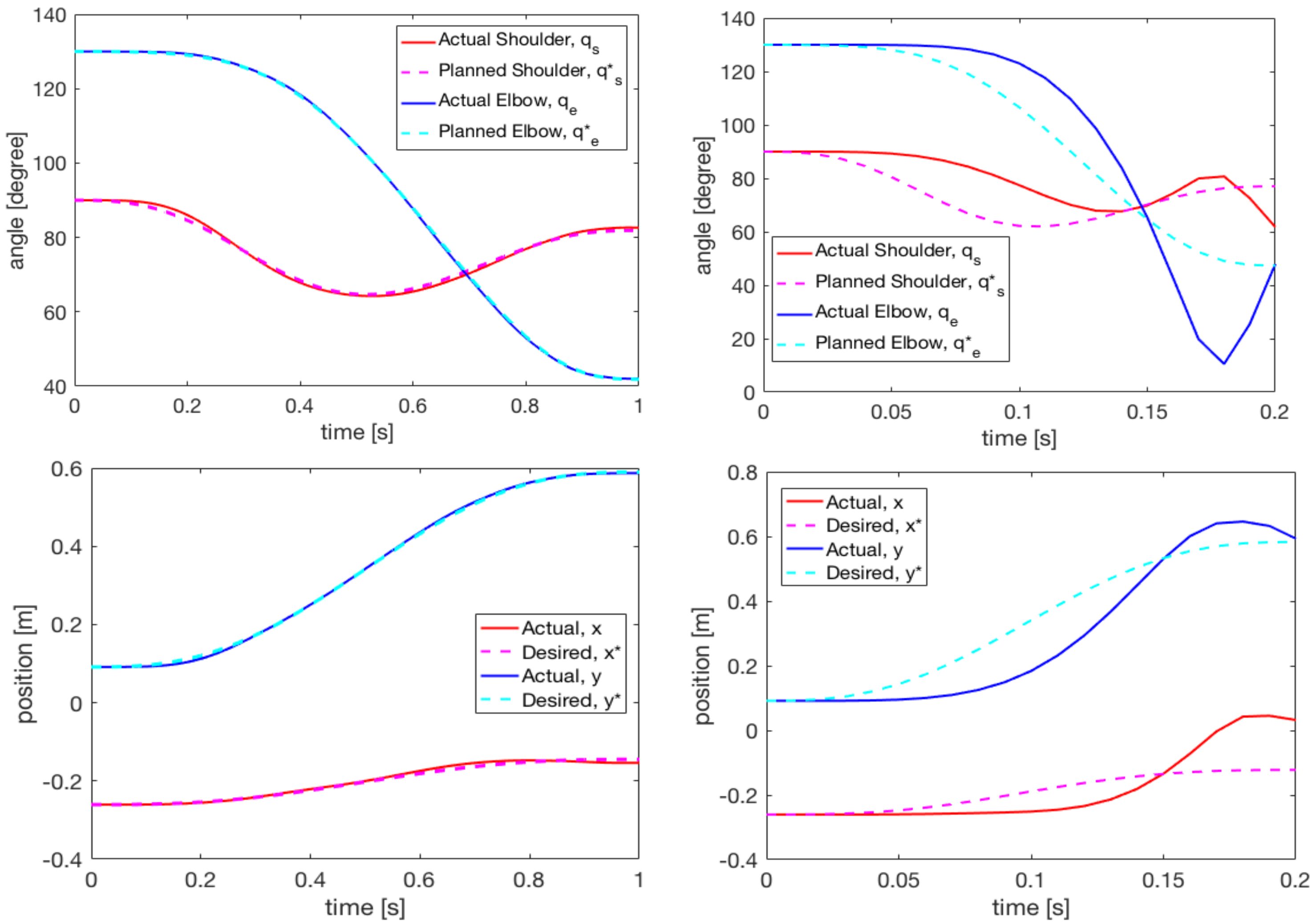


Figure 7: Trajectory graphs of an arm under PD control. The first row shows angular position with respect to time, the second row presents positional trajectory with respect to time and the last row is the hands trajectory. [Left Column] Properties of the PD controller are Kp = 100 Nm/rad and Kd = 0 Nms/rad. [Right Column] PD controller properties are Kp = 100 Nm/rad and Kd = 10,000 Nms/rad.

From Figure 7 it was found that as you decreased Kd while keeping Kp constant at 100Nm/rad, the accuracy of the actual trajectory curves remains consistent, relative to the trajectories from Part A, seen in Figure 4 and 5 above. Therefore the desired trajectories agree well with the actual trajectories when Kd is low or non-existent, as long as Kp gain is present. Alternatively, as you increase the Kd gain while keeping Kp constant at 100Nm/rad, the curves accuracy increases and agrees almost perfectly with the desired curves; however, when the effect of Kd is quite large, such as 10,000 Nms/rad, the angular trajectory reach a singularity and as a result, the solutions become inaccurate. This effect is evident in Figure 7, as the actual trajectories displayed are jagged or straight lines and not smooth curves like the desired pathways.

Part C

The last investigation to be analyzed is how movement duration time affects accuracy of the arms movement trajectories. The arms movement was modelled for both a duration of 1s, similarly to the test done in Question 2, Part A above, and 0.2s. The resulting graphs from this investigation can be found below in Figure 8.



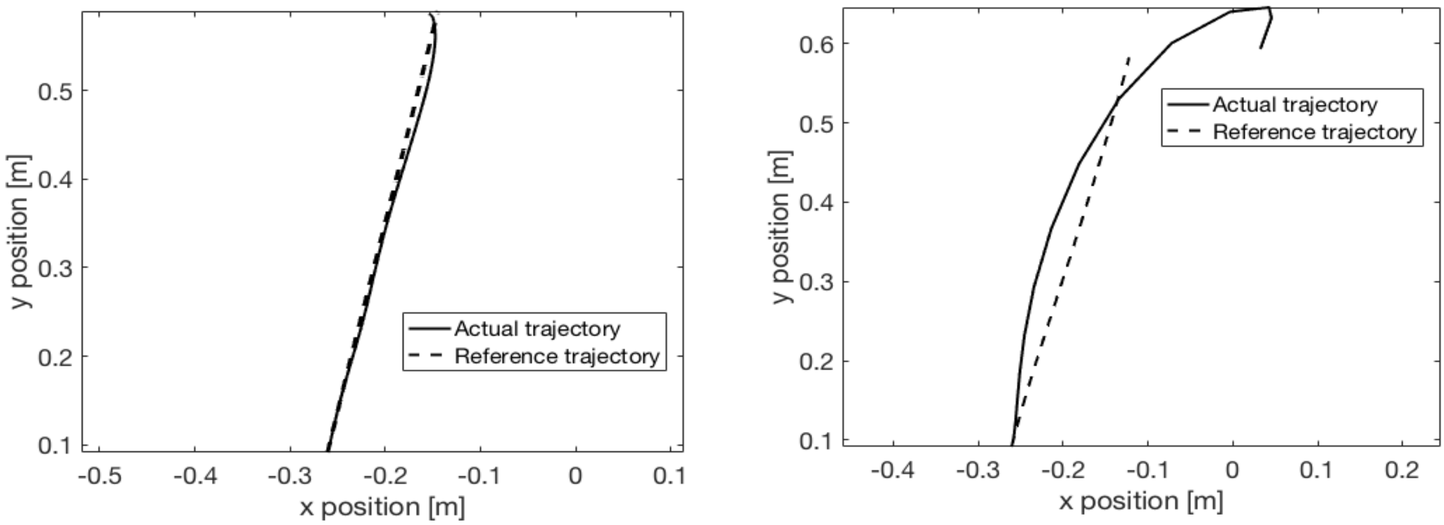


Figure 8: Trajectory graphs of an arm under PD control with Kp = 100Ms/rad and Kd = 10Nms/rad. The first row shows angular position with respect to time, the second row presents positional trajectory with respect to time and the last row is the hands trajectory. [Left Column] Movement duration of 1 s. [Right Column] Movement duration of 0.2s.

From Figure 8, is can be seen that as time duration of a movement decreases, the accuracy of the trajectory curves decreases. For example, in the middle row of Figure 8, it can be seen that the actual trajectory of the hand oscillates around the planned trajectory when movement duration is T=0.2s (right column), but the curves agree with one another when the time duration is T=1s (left column). A possible reason for this result may be due to the added torque applied to the joints during acceleration. When the velocity of joints increases, the force applied to those joints increases, therefore, it is likely that the additional torque is not present in the planned trajectory equations, but is present in reality, hence the difference in the resulting trajectories.

Appendix

The results for this assignment were determined using the MATLAB code given to us. I kept the script relatively unchanged, but the line of code that I did change/add was for the torque controller. The equations are presented as follows.

For question 1A:

```
% WRITE HERE THE CONTROLLER EQUATION
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if i > 200
        Torque = [0 -0.1*qdot(i,2)]';
else
        Torque = [0 0.02-0.1*qdot(i,2)]';
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

For question 1B:

```
% WRITE HERE THE CONTROLLER EQUATION
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if i > 200
        Torque = [-0.1*qdot(i,1) -0.1*qdot(i,2)]';
else
        Torque = [-0.1*qdot(i,1) 0.02-0.1*qdot(i,2)]';
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

For question 2:

```
% WRITE HERE THE CONTROLLER EQUATION
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

 Torque = Kp*(qr(i,:)-q(i,:))+Kd*(qr(i,:)-q(i,:));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```