RBE 502 — Robot Control

Instructor: Siavash Farzan

Spring 2023

Programming Assignment 5:

Adaptive Control of the RRBot Robotic Arm

By Chinmayee Prabhakar

a) The cubic polynomial trajectories for the given time span and the desired initial and final joint angles and velocities :

t0 = 0, tf = 10 sec

θ1(t0) = 180◦, θ1(tf) = 0, θ2(t0) = 90◦, θ2(tf) = 0

θ1_dot(t0) = θ1_dot(tf) = θ2_dot(t0) = θ2_dot(tf) = 0

The desired trajectories:

Theta1 = (pi*t^3)/500 - (3*pi*t^2)/100 + pi

Theta2 = (pi*t^3)/1000 - (3*pi*t^2)/200 + pi/2

Theta1_dot = (3*pi*t^2)/500 - (3*pi*t)/50

Theta2_dot = (3*pi*t^2)/1000 - (3*pi*t)/100]

Theta1_ddot = (3*pi*t)/250 - (3*pi)/50

Theta2_ddot = (3*pi*t)/500 - (3*pi)/100

Programming Assignment 5

We have the Manipulator Eq. form of the system as:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau$$

where,

$$M = \begin{bmatrix} m_1 r_1^2 + m_2 r_2^2 + 2 m_2 \cos\theta_2 r_2 l_1 + m_2 l_1^2 + I_1 + I_2 & m_2 r_2^2 + l_1 m_2 \cos(\text{theta2}) r_2 + I_2 \\ m_2 r_2^2 + l_1 m_2 \cos\theta_2 r_2 + I_2 & m_2 r_2^2 + I_2 \end{bmatrix}$$

$$C = \begin{bmatrix} -2 r_2 \dot{\theta}_2 l_1 m_2 \sin(\theta_2) & -r_2 \dot{\theta}_2 l_1 m_2 \sin\theta_2 \\ r_2 l_1 m_2 \sin(\theta_2) \dot{\theta}_1 & 0 \end{bmatrix}$$

$$g = \begin{bmatrix} -\sin\theta_1 (r_1 g m_1 + g l_1 m_2) - r_2 g m_2 \sin(\theta_1 + \theta_2) \\ -r_2 g m_2 \sin(\theta_1 + \theta_2) \end{bmatrix}$$

$$q = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} \qquad \dot{q} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \qquad \ddot{q} = \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix}$$

The Manipulator Eq. form can be written in the linear parametric form as :

$$Y(q, \dot{q}, \ddot{q})\, \alpha = \tau$$

where ~~ea~~

$$Y = \begin{bmatrix} \ddot{\theta}_1, & \cos\theta_2 (2 \times \ddot{\theta}_1 + \ddot{\theta}_2) - 2\sin\theta_2 \times \dot{\theta}_1 \times \dot{\theta}_2 - \sin\theta_2 \times \dot{\theta}_2^2, & \ddot{\theta}_2, & -\sin\theta_1 g, & -\sin(\theta_1+\theta_2) g \\ 0, & \sin\theta_2 \times \dot{\theta}_1^2 + \cos\theta_2 \cdot \ddot{\theta}_1, & \ddot{\theta}_1 + \ddot{\theta}_2, & 0, & -\sin(\theta_1+\theta_2) \times g \end{bmatrix}$$

$$\alpha = \begin{bmatrix} m_2 \times l_1^2 + m_1 \times r_1^2 + m_2 \times r_2^2 + I_1 + I_2 \\ m_2 \times l_1 \times r_2 \\ m_2 \times r_2^2 + I_2 \\ m_1 \times r_1 + m_2 \times l_1 \\ m_2 \times r_2 \end{bmatrix}$$

$$Y \times \alpha = \Big[\ddot{\theta}_1[m_1 r_1^2 + m_2 r_2^2 + m_2 l_1^2 + I_1 + I_2] + \cdots$$
$$m_2 l_1 r_2 [\cos\theta_2 (2\ddot{\theta}_1 + \ddot{\theta}_2) - 2\dot{\theta}_1 \dot{\theta}_2 \sin\theta_2 - \dot{\theta}_2^2 \cdot \sin\theta_2] + \cdots$$
$$\ddot{\theta}_2 [m_2 r_2^2 + I_2] + \cdots$$
$$-\sin\theta_1 \cdot g [m_1 r_1 + m_2 l_1] + \cdots$$
$$m_2 r_2 g (-\sin(\theta_1 + \theta_2));$$

$$0 + \cdots$$
$$m_2 l_1 r_2 [\dot{\theta}_1^2 \sin\theta_2 + \ddot{\theta}_1 \cdot \cos\theta_2] + \cdots$$
$$[m_2 r_2^2 + I_2][\ddot{\theta}_1 + \ddot{\theta}_2] + \cdots$$
$$0 + \cdots$$
$$m_2 r_2 g (-\sin(\theta_1 + \theta_2))\Big]$$

Taking $\ddot{q}$, $\dot{q}$, q terms as common.

$$= \Big[[m_1 r_1^2 + m_2 r_2^2 + m_2 l_1^2 + I_1 + I_2 + 2 m_2 l_1 r_2 \cos\theta_2]\ddot{\theta}_1 + \cdots$$
$$[m_2 r_2^2 + I_2 + m_2 l_1 r_2 \cos\theta_2]\ddot{\theta}_2 + \cdots$$
$$[-2\dot{\theta}_2 m_2 l_1 r_2 \sin\theta_2]\dot{\theta}_1 + [-\dot{\theta}_2 \sin\theta_2 m_2 l_1 r_2]\dot{\theta}_2 + \cdots$$
$$[-\sin\theta_1 g [m_1 r_1 + m_2 l_2] - \sin(\theta_1 + \theta_2)[m_2 r_2 g]]\Big];$$

$$[m_2 l_1 r_2 \cos\theta_2 + m_2 r_2^2 + I_2]\ddot{\theta}_1 + \cdots$$
$$[m_2 r_2^2 + I_2]\ddot{\theta}_2 + \cdots$$
$$[m_2 l_1 r_2 \dot{\theta}_1 \sin\theta_2]\dot{\theta}_1 + (0)\dot{\theta}_2 + \cdots$$
$$[-\sin(\theta_1 + \theta_2)[m_2 r_2 g]]$$

This can be further simplified as:

$$\begin{bmatrix} m_1 r_1^2 + m_2 r_2^2 + 2 m_2 \cos\theta_2 r_2 l_1 + m_2 l_1^2 + I + I_2 & m_2 r_2^2 + l_1 m_2 \cos\theta_2 r_2 + I_2 \\ m_2 r_2^2 + l_1 m_2 \cos\theta_2 r_2 + I_2 & m_2 r_2^2 + I_2 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} +$$

$$\begin{bmatrix} -2 r_2 \dot{\theta}_2 l_1 m_2 \sin\theta_2 & -r_2 \dot{\theta}_2 l_1 m_2 \sin\theta_2 \\ r_2 l_1 m_2 \sin(\theta_2)\dot{\theta}_1 & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} +$$

$$\begin{bmatrix} -\sin\theta_1 (r_1 g m_1 + g l_1 m_2) - r_2 g m_2 \sin(\theta_1 + \theta_2) \\ -r_2 g m_2 \sin(\theta_1 + \theta_2) \end{bmatrix} \rightarrow$$

This is further Manipulator Eq. Form

b)

b) We have the system:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau$$

which can be written in linear parametric form as

$$Y(q,\dot{q},\ddot{q})\,\alpha = \tau$$

Constructing 'τ' using nominal values,

we get, $\tau = Y(q,\dot{q},\ddot{q})\hat{\alpha}$

Using virtual control input 'v' as $\ddot{q}$, we can write our Feedback linearization control law as:

$$\tau = Y(q,\dot{q},v)\hat{\alpha}$$

~~For trajectory tracking, we design the virtual control input as~~

The state space form of our system:

$$X = \begin{bmatrix} e \\ \dot{e} \end{bmatrix}, \quad e = q - q_d$$

$$\dot{e} = \dot{q} - \dot{q}_d$$

We get,

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \& \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

→ set Eigenvalues to : $\{-2, -2, -3, -3\}$

Using 'place' fn. in MATLAB,

we get k = place (A, B, lambda)

$$k = \begin{bmatrix} 6 & 0 & 5 & 0 \\ 0 & 6 & 0 & 5 \end{bmatrix}$$

Therefore, for trajectory tracking,

the virtual control input :

$$v = \ddot{q}_d - K_p e - K_d \dot{e}$$

We define the state-space model:

$$\dot{x} = \begin{bmatrix} \dot{e} \\ \ddot{e} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & I \\ -K_p & -K_d \end{bmatrix}}_{A_{cl}} \begin{bmatrix} e \\ \dot{e} \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ I \end{bmatrix}}_{B} \hat{M}^{-1} Y(q, \dot{q}, \ddot{q}) \tilde{\alpha}$$

where, $A_{cl} = A - BK$, $\hat{M} \rightarrow$ nominal value

$$\tilde{\alpha} = \hat{\alpha} - \alpha$$

$$\phi = \hat{M}^{-1} Y(q, \dot{q}, \ddot{q})$$

∴ We obtain P matrix using MATLAB to find the solution to the Lyapunov Equation:

$$A^T P + PA = -Q.$$

where $A = A_{cl}$, $Q = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 10 \end{bmatrix}$

$\Rightarrow$ P = lyap (A', Q)

$$P = \begin{bmatrix} 11.1667 & 0 & 0.8333 & 0 \\ 0 & 11.1667 & 0 & 0.8333 \\ 0.833 & 0 & 1.1667 & 0 \\ 0 & 0.8333 & 0 & 1.1667 \end{bmatrix}$$

We initially ~~assume~~ set Γ to be $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$

$\therefore$ The control law:

$$\boxed{\begin{aligned} \tau &= \hat{M} v + \hat{C}\dot{\theta} + \hat{g} \\ \text{where } v &= \ddot{q}_d - K_p e - K_d \dot{e} \end{aligned}}$$

The adaptation update law:

$$\boxed{\dot{\hat{\alpha}} = -\Gamma^{-1} \phi^T B^T P x}$$

where $\Gamma = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$, $\phi = \hat{M}^{-1} Y(q, \dot{q}, \ddot{q})$, $P = \begin{bmatrix} 11.1667 & 0 & 0.83 & 0 \\ 0 & 11.16 & 0 & 0.83 \\ 0.83 & 0 & 1.16 & 0 \\ 0 & 0.83 & 0 & 1.16 \end{bmatrix}$

$B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$,

$x = \begin{bmatrix} e \\ \dot{e} \end{bmatrix}$

$$\hat{M} = [a + (b \times 2 \times \cos\theta_2), \quad d + (b \times \cos\theta_2); \quad d + (b \times \cos\theta_2), \quad d]$$

where
$$a = I_1 + I_2 + m_1 r_1^2 + m_2 l_1^2 + m_2 r_2^2 = \hat{\alpha}_1$$
$$b = m_2 l_1 r_2 = \hat{\alpha}_2$$
$$d = I_2 + m_2 r_2^2 = \hat{\alpha}_3$$

c) We then update the ode function .
We define the trajectories and the virtual control law.
Using the Linear parametric form with alpha_hat we compute the state feedback linearized control law.

We finally compute the state derivatives and alpha_hat derivative as these are to be integrated over time to simulate the system performance.

```
dX(1) = theta1_dot;

dX(2) = theta2_dot;

dX(3) = (I2*Tau1 - I2*Tau2 + Tau1*r2^2*m2 - Tau2*r2^2*m2 +
r2^3*theta1_dot^2*l1*m2^2*sin(theta2) + r2^3*theta2_dot^2*l1*m2^2*sin(theta2) +
r2^2*g*l1*m2^2*sin(theta1) + I2*r1*g*m1*sin(theta1) - Tau2*r2*l1*m2*cos(theta2) +
I2*g*l1*m2*sin(theta1) + 2*r2^3*theta1_dot*theta2_dot*l1*m2^2*sin(theta2) +
r2^2*theta1_dot^2*l1^2*m2^2*cos(theta2)*sin(theta2) - r2^2*g*l1*m2^2*sin(theta1 +
theta2)*cos(theta2) + I2*r2*theta1_dot^2*l1*m2*sin(theta2) + I2*r2*theta2_dot^2*l1*m2*sin(theta2)
+ r1*r2^2*g*m1*m2*sin(theta1) + 2*I2*r2*theta1_dot*theta2_dot*l1*m2*sin(theta2))/(I1*I2 +
r2^2*l1^2*m2^2 + I2*r1^2*m1 + I1*r2^2*m2 + I2*l1^2*m2 + r1^2*r2^2*m1*m2 -
r2^2*l1^2*m2^2*cos(theta2)^2);

dX(4) = -(I2*Tau1 - I1*Tau2 - I2*Tau2 - Tau2*r1^2*m1 + Tau1*r2^2*m2 - Tau2*r2^2*m2 -
Tau2*l1^2*m2 + r2*theta1_dot^2*l1^3*m2^2*sin(theta2) + r2^3*theta1_dot^2*l1*m2^2*sin(theta2) +
r2^3*theta2_dot^2*l1*m2^2*sin(theta2) - r2*g*l1^2*m2^2*sin(theta1 + theta2) -
I1*r2*g*m2*sin(theta1 + theta2) + r2^2*g*l1*m2^2*sin(theta1) + I2*r1*g*m1*sin(theta1) +
Tau1*r2*l1*m2*cos(theta2) - 2*Tau2*r2*l1*m2*cos(theta2) + I2*g*l1*m2*sin(theta1) +
2*r2^3*theta1_dot*theta2_dot*l1*m2^2*sin(theta2) +
2*r2^2*theta1_dot^2*l1^2*m2^2*cos(theta2)*sin(theta2) +
r2^2*theta2_dot^2*l1^2*m2^2*cos(theta2)*sin(theta2) - r2^2*g*l1*m2^2*sin(theta1 +
theta2)*cos(theta2) + r2*g*l1^2*m2^2*cos(theta2)*sin(theta1) - r1^2*r2*g*m1*m2*sin(theta1 +
theta2) + I1*r2*theta1_dot^2*l1*m2*sin(theta2) + I2*r2*theta1_dot^2*l1*m2*sin(theta2) +
I2*r2*theta2_dot^2*l1*m2*sin(theta2) + r1*r2^2*g*m1*m2*sin(theta1) +
2*r2^2*theta1_dot*theta2_dot*l1^2*m2^2*cos(theta2)*sin(theta2) +
r1^2*r2*theta1_dot^2*l1*m1*m2*sin(theta2) + 2*I2*r2*theta1_dot*theta2_dot*l1*m2*sin(theta2) +
r1*r2*g*l1*m1*m2*cos(theta2)*sin(theta1))/(I1*I2 + r2^2*l1^2*m2^2 + I2*r1^2*m1 + I1*r2^2*m2 +
I2*l1^2*m2 + r1^2*r2^2*m1*m2 - r2^2*l1^2*m2^2*cos(theta2)^2);

dX(5) = alphat_hat_dot(1);

dX(6) = alphat_hat_dot(2);

dX(7) = alphat_hat_dot(3);

dX(8) = alphat_hat_dot(4);

dX(9) = alphat_hat_dot(5);
```

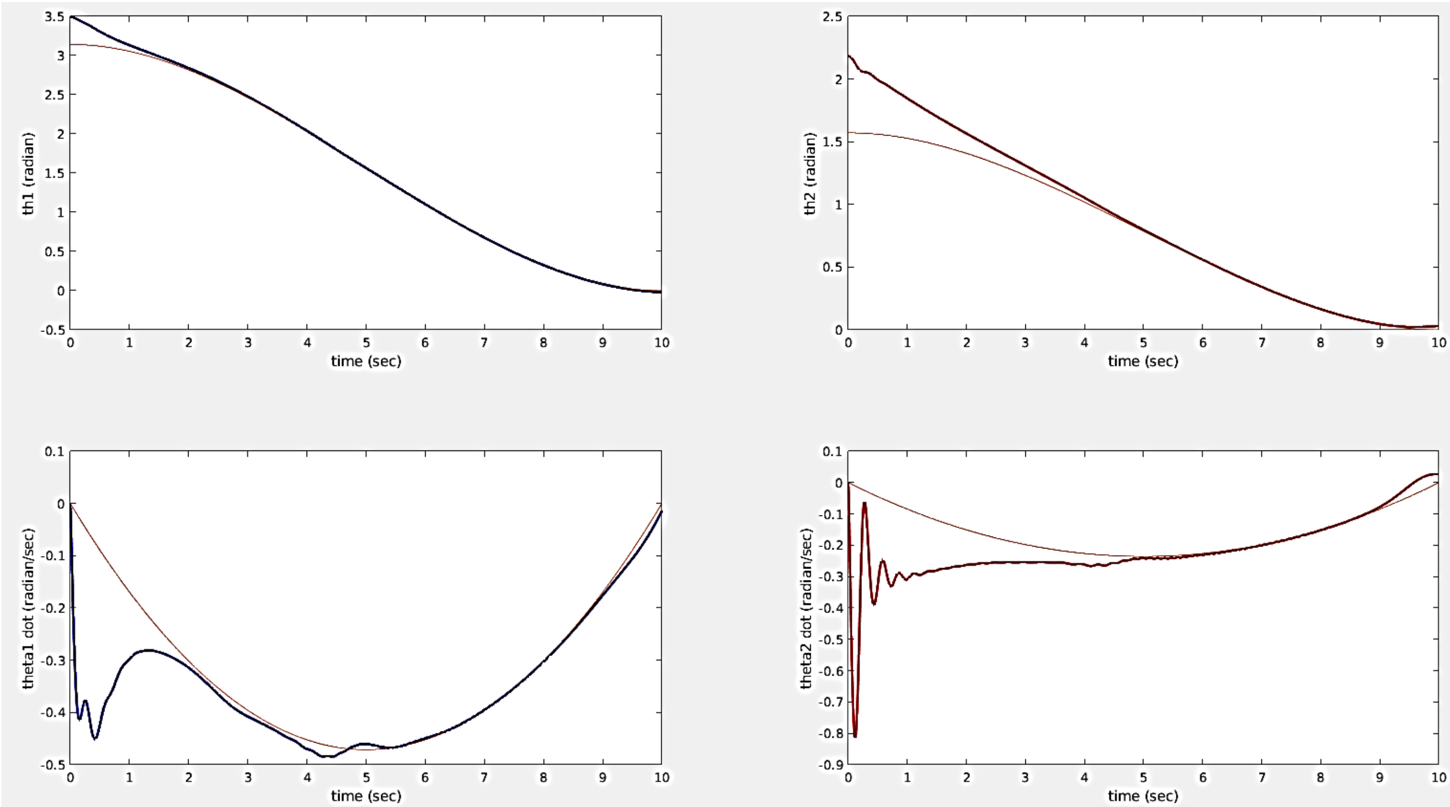
d) A simulation of the system is produced in MATLAB with the time span of [0, 10] sec and initial conditions of:

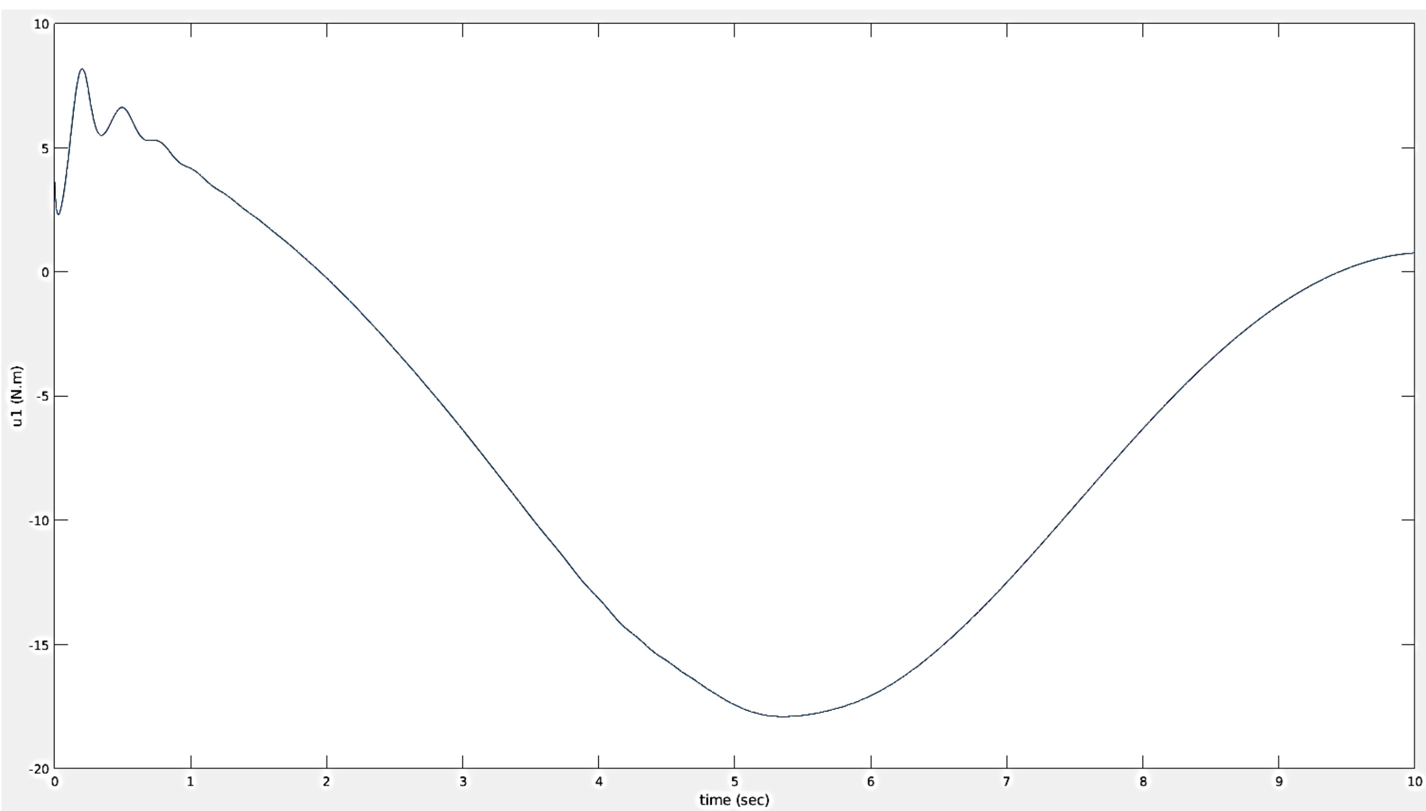
$$\theta_1(0) = 200^\circ,\ \theta_2(0) = 125^\circ,\ \dot{\theta}_1 = 0,\ \dot{\theta}_2 = 0$$

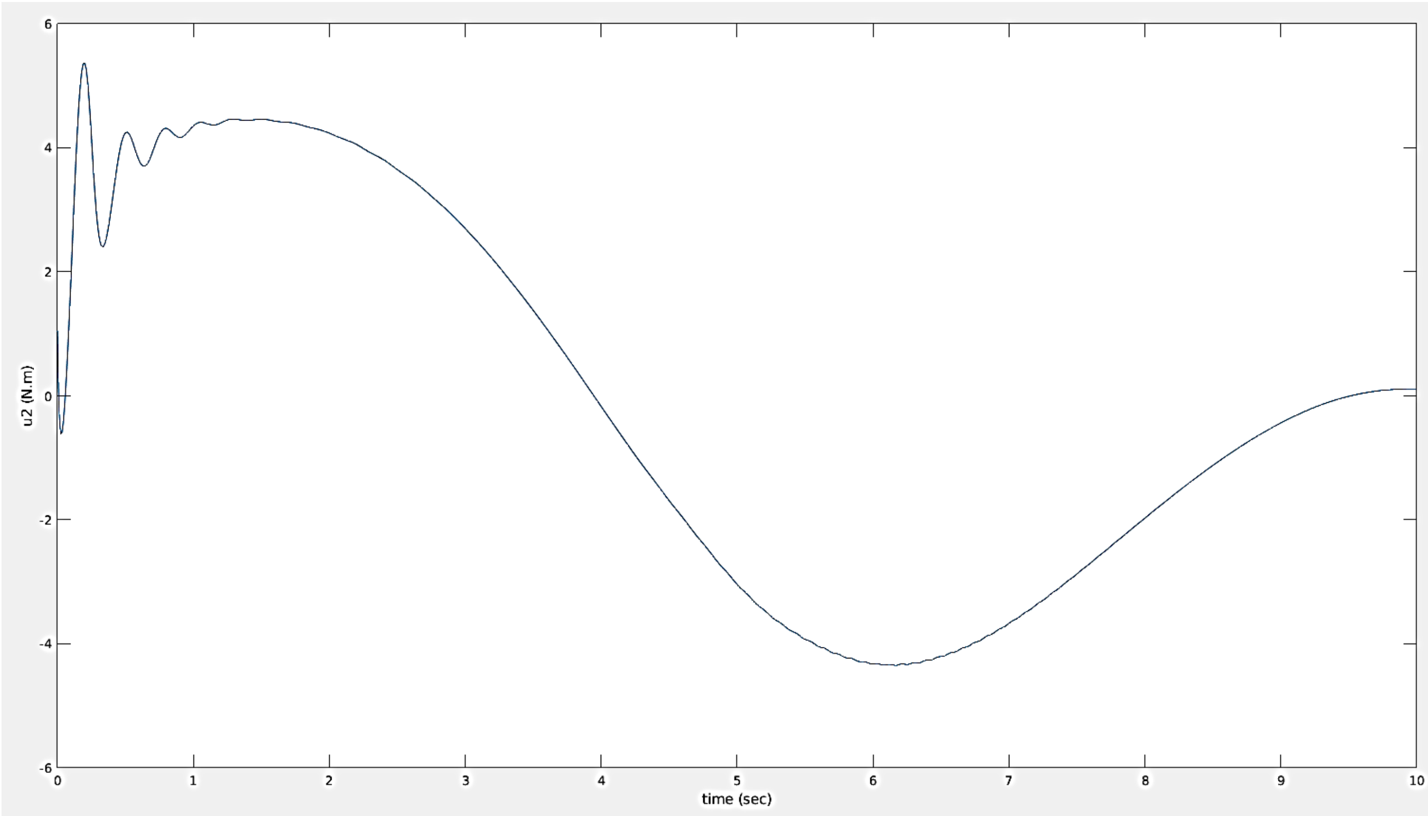
We set the initial values of the unknown parameter vector α to 75% of the actual values:

$$\hat{\alpha}(0) = 0.75\alpha$$

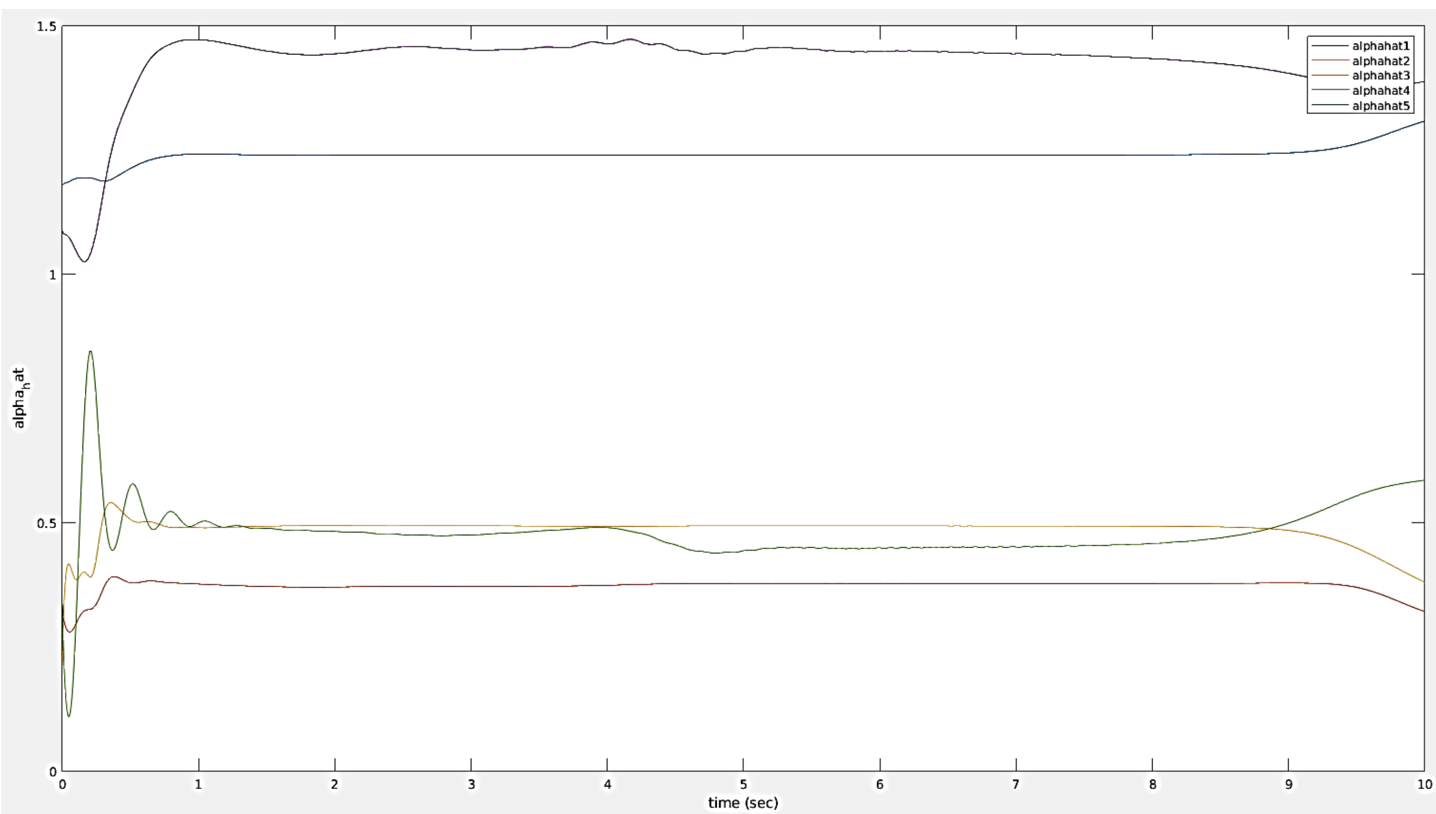
The following trajectories are obtained:







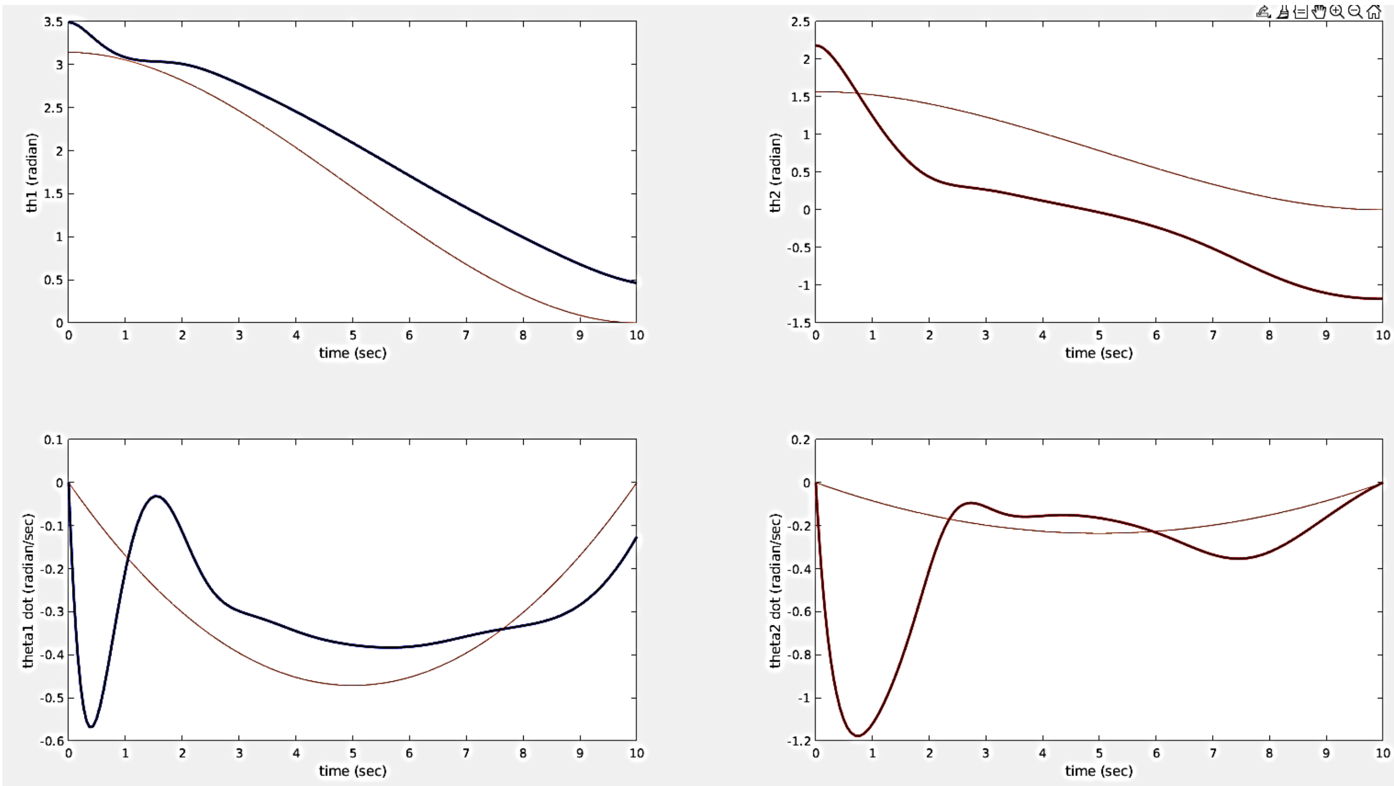
u2 (N.m)
time (sec)

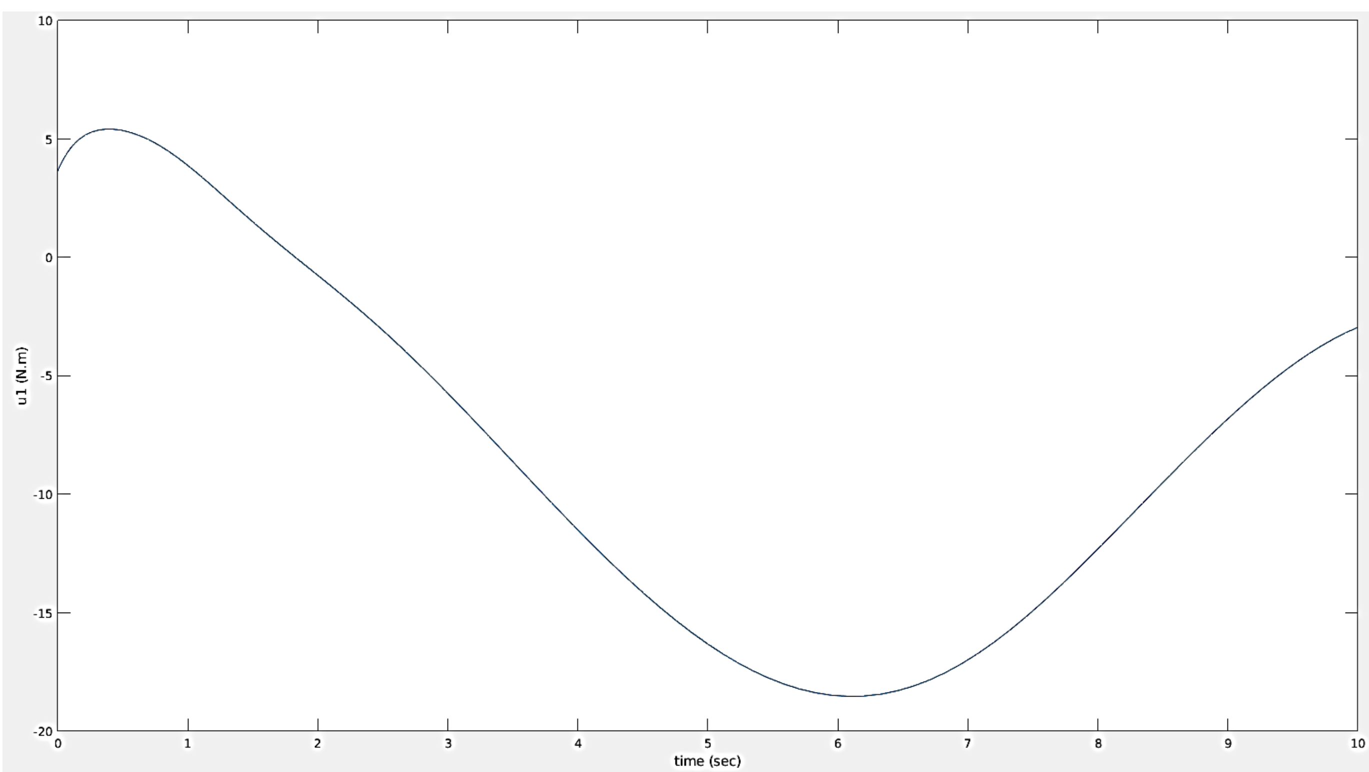


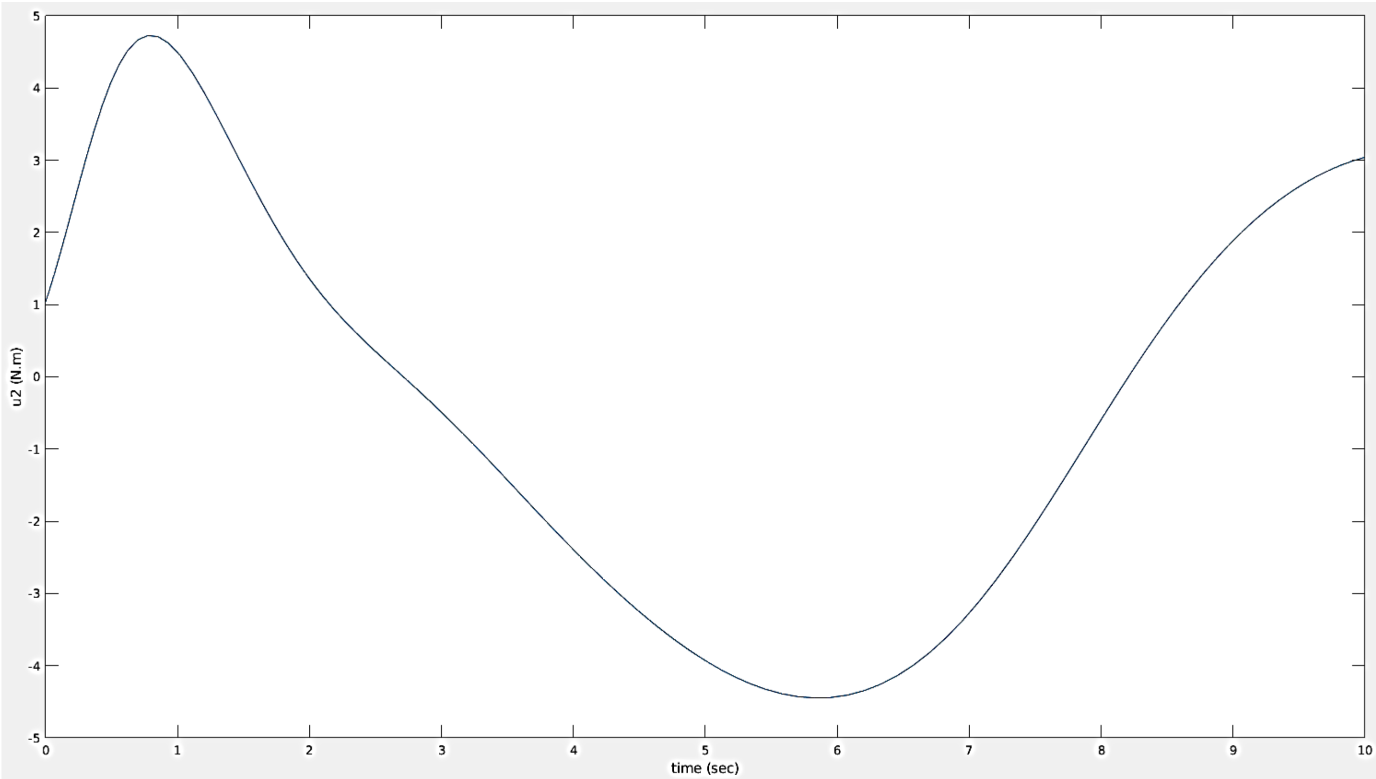
alphahat1
alphahat2
alphahat3
alphahat4
alphahat5
alpha_hat
time (sec)

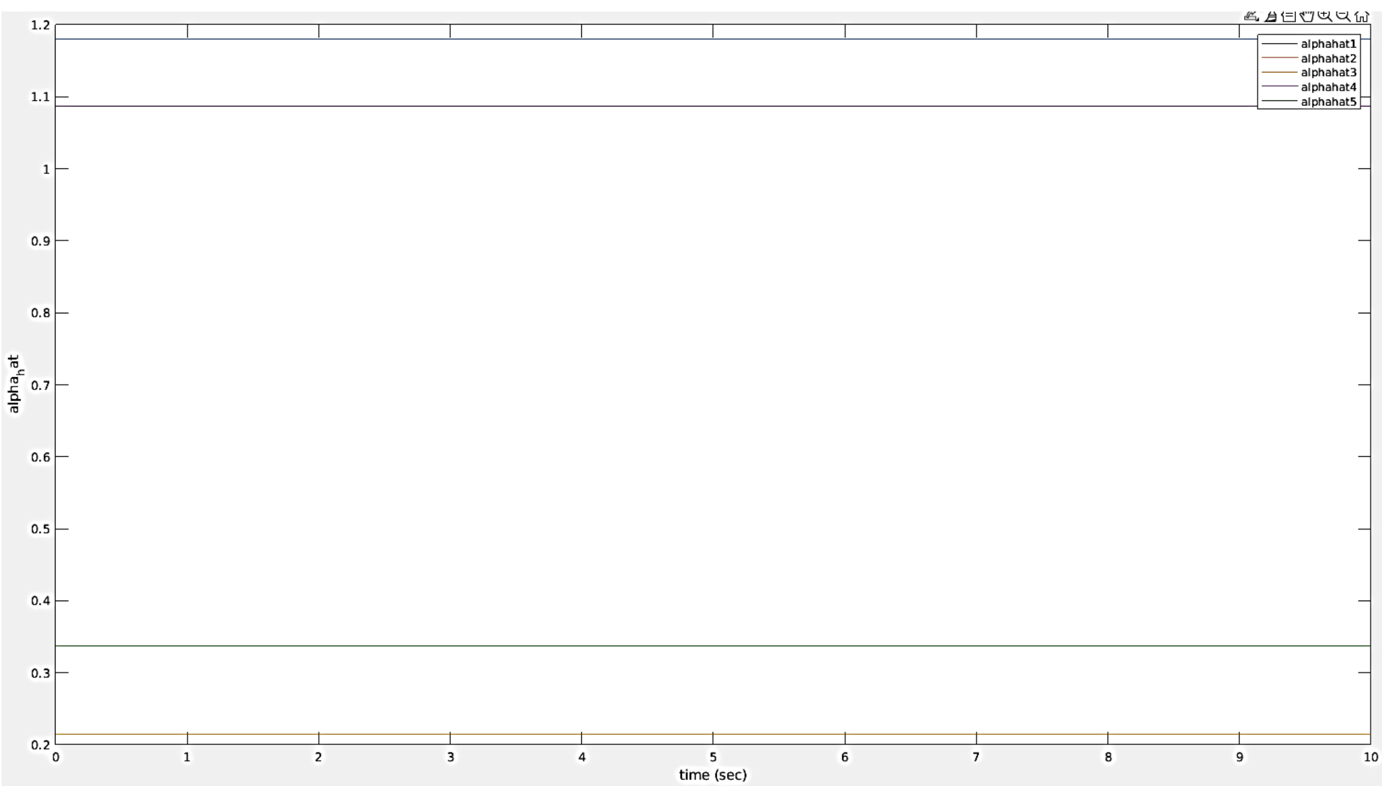
We observe that under the adaptive control law, the state trajectories of the system converge to the desired trajectories within the given time frame. Additionally, the ‘alpha’ parameter values also stabilize to a constant value over time.

e) Without the adaptive inverse dynamics control:









We observe that in the absence of updating the parameters, the system shows no sign of converging to the desired trajectories. This is because of the uncertainty in the system parameters. The controller hence fails to stabilize the system.