Fall 2022

Final Project

Control of Robotic Systems

December 20, 2022

Student:
Aditi Bhoir, Mudit Singal

Instructors:
W. Malik

Course code:
ENPM667

Contents

1 Problem Statement

Consider a crane that moves along an one-dimensional track. It behaves as a frictionless cart with mass$M$ actuated by an external force F that constitutes the input of the system. There are two loads suspended from cables attached to the crane. The loads have mass $m1$ and $m2$, and the lengths of the cables are $l1$ and $l2$, respectively. The following figure depicts the crane and associated variables used throughout this project.

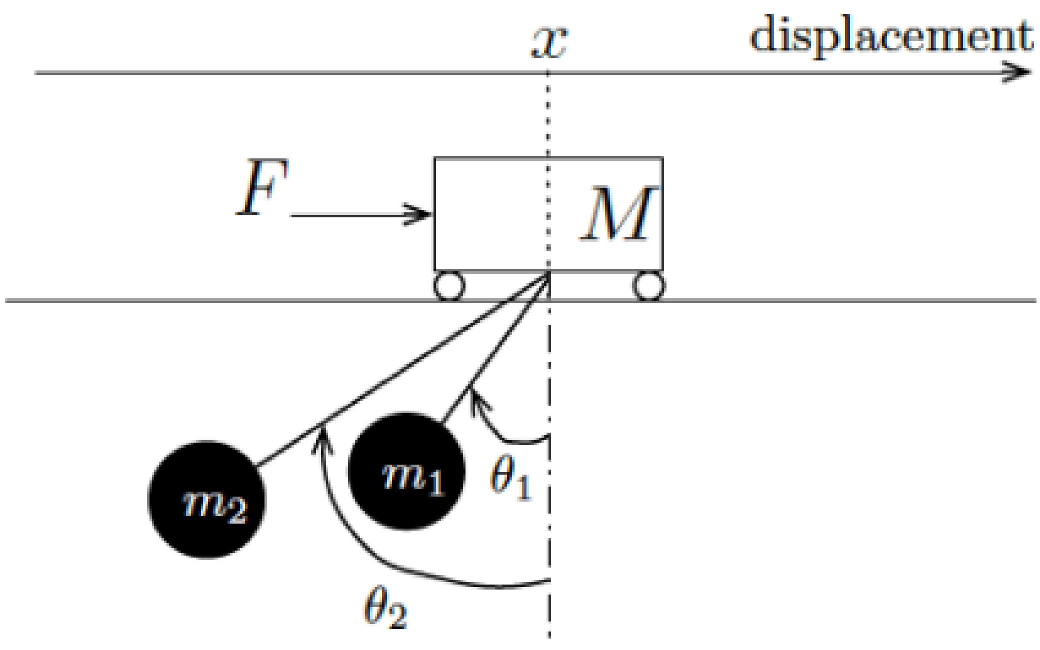

Figure 1: Given System

2 Equation of motion and non-linear state space representation

The position of mass m_1 in x and y direction is given as follows:

$$x_{m1} = x - l_1 sin(\theta_1)$$
$$y_{m1} = -l_1 cos(\theta_1)$$

The velocity of mass m_1 is given as follows:

$$\dot{x}_{m1} = \dot{x} - l_1 cos(\theta_1)\dot{\theta}_1$$
$$\dot{y}_{m1} = l_1 sin(\theta_1)\dot{\theta}_1$$

similarly the position of mass $m2$ in x and y direction is given as follows:

$$x_{m2} = x - l_2 sin(\theta_2)$$
$$y_{m2} = -l_2 cos(\theta_2)$$

The velocity of mass m_2 is given as follows:

$$\dot{x}_{m2} = \dot{x} - l_2 cos(\theta_2)\dot{\theta}_2$$
$$\dot{y}_{m2} = l_2 sin(\theta_2)\dot{\theta}_2$$

The kinematic energy of the system is given by:

Total Kinetic Energy = Kinetic energy of mass M + Kinetic energy of mass m_1 + Kinetic energy of mass m_2

$$\mathcal{K} = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m_1(\dot{x}_{m1}^2 + \dot{y}_{m1}^2) + \frac{1}{2}m_2(\dot{x}_{m2}^2 + \dot{y}_{m2}^2)$$

by substituting the values of $\dot{x}_{m1}$, $\dot{x}_{m2}$, $\dot{y}_{m1}$ and $\dot{y}_{m2}$ we get following equation for the kinetic energy of the system,

$$\mathcal{K} = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m_1\dot{x}^2 + \frac{1}{2}m_2\dot{x}^2 + \frac{1}{2}m_1 l_1\dot{\theta}_1^2 + \frac{1}{2}m_2 l_2\dot{\theta}_2^2 - m_1 l_1\dot{\theta}_1\dot{x}cos(\theta_1) - m_2 l_2\dot{\theta}_2\dot{x}cos(\theta_2)$$

Potential energy of the system is given as follows:

$$\mathcal{P} = -m_1 g l_1 cos(\theta_1) - m_2 g l_2 cos(\theta_2)$$

The Euler Lagrange equation is given by:

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{y}}\right) - \frac{\partial \mathcal{L}}{\partial y} = F(t) \tag{1}$$

The Lagrangian function is given by:

$$\mathcal{L} = \mathcal{K} - \mathcal{P} \tag{2}$$

By substituting $\mathcal{K}$ and $\mathcal{P}$ in equation 2 we get,

$$\mathcal{L} = \frac{1}{2}(M+m_1+m_2)\dot{x}^2 + \frac{1}{2}m_1 l_1^2\dot{\theta}_1^2 + \frac{1}{2}m_2 l_2^2\dot{\theta}_2^2 - m_1 l_1\dot{\theta}_1 cos(\theta_1) - m_2 l_2\dot{\theta}_2 cos(\theta_2) + m_1 g l_1 cos(\theta_1)$$
$$+m_1 g l_1 cos(\theta_1) \tag{3}$$

We know that,

$$\frac{\partial \mathcal{L}}{\partial x} = 0$$

Differentiating $\mathcal{L}$ wrt $\dot{x}$ we get,

$$\frac{\partial \mathcal{L}}{\partial \dot{x}} = (M + m_1 + m_2)\dot{x} - m_1 l_1\dot{\theta}_1 cos(\theta_1) - m_2 l_2\dot{\theta}_2 cos(\theta_2)$$

Differentiating above equation wrt time,

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{x}}\right) = (M+m_1+m_2)\ddot{x} - m_1 l_1 \ddot{\theta}_1 cos(\theta 1) + m_1 l_1 \dot{\theta}_1^2 sin(\theta 1) - m_2 l_2 \ddot{\theta}_2 cos(\theta_2) + m_2 l_2 \dot{\theta}_2^2 sin(\theta_2)$$

Similarly, Differentiating $\mathcal{L}$ wrt θ_1 and θ_2 we get,

$$\frac{\partial \mathcal{L}}{\partial \theta_1} = m_1 l_1 \dot{\theta}_1 \dot{x} sin(\theta_1) - m_1 g l_1 sin(\theta_1)$$
$$\frac{\partial \mathcal{L}}{\partial \theta_2} = m_2 l_2 \dot{\theta}_2 \dot{x} sin(\theta_2) - m_2 g l_2 sin(\theta_2)$$

Differentiating $\mathcal{L}$ wrt $\dot{\theta}_1$ and $\dot{\theta}_2$ we get,

$$\frac{\partial \mathcal{L}}{\partial \dot{\theta}_1} = m_1 l_1^2 \dot{\theta}_1 - m_1 l_1 \dot{x} cos(\theta_1)$$
$$\frac{\partial \mathcal{L}}{\partial \dot{\theta}_2} = m_2 l_2^2 \dot{\theta}_2 - m_2 l_2 \dot{x} cos(\theta_2)$$

Differentiating above two equations wrt time,

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\theta}_1}\right) = m_1 l_1^2 \ddot{\theta}_1 - m_1 l_1 cos(\theta_1)\ddot{x} + m_1 l_1 \dot{\theta}_1 sin(\theta_1)\dot{x}$$
$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\theta}_2}\right) = m_2 l_2^2 \ddot{\theta}_2 - m_2 l_2 cos(\theta_2)\ddot{x} + m_2 l_2 \dot{\theta}_2 sin(\theta_2)\dot{x}$$

From equation 1 we get,

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{x}}\right) - \frac{\partial \mathcal{L}}{\partial \dot{x}} = F(t) \tag{4}$$

Since the external torques are zero, from equation 1 we will get,

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\theta}_1}\right) - \frac{\partial \mathcal{L}}{\partial \dot{\theta}_1} = 0 \tag{5}$$

and,

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\theta}_2}\right) - \frac{\partial \mathcal{L}}{\partial \dot{\theta}_2} = 0 \tag{6}$$

2.1 Non-linear state space representation

The state space representation for non linear state is given as follows:

$$\ddot{x} = \frac{-m_1 g sin(\theta_1) cos(\theta_2) - m_2 g sin(\theta_2) cos(\theta_2) - m_1 l_1 \dot{\theta}_2^2 sin(\theta_1) - m_2 l_2 \dot{\theta}_2^2 sin(\theta_2) + F}{M + m_1 + m_2 - m_1 cos^2(\theta_1) - m_2 cos^2(\theta_2)}$$

$$\ddot{\theta}_1 = \frac{-m_1 gsin(\theta_1)cos(\theta_2) - m_2 gsin(\theta_2)cos(\theta_2) - m_1 l_1 \dot{\theta}_1^2 sin(\theta_1) - m_2 l_2 \dot{\theta}_2^2 sin(\theta_2) + F}{l_1(M + m_1 + m_2 - m_1 cos^2(\theta_1) - m_2 cos^2(\theta_2))} - \frac{gsin(\theta_1)}{l_1}$$

$$\ddot{\theta}_2 = \frac{-m_1 gsin(\theta_1)cos(\theta_2) - m_2 gsin(\theta_2)cos(\theta_2) - m_1 l_1 \dot{\theta}_1^2 - m_2 l_2 \dot{\theta}_2^2 sin(\theta_2) + F}{l_2(M + m_1 + m_2 - m_1 cos^2(\theta_1) - m_2 cos^2(\theta_2))} - \frac{gsin(\theta_1)}{l_2}$$

3 Linearized system and state space representation of state space of the system

We have to linearize the system around the equilibrium point specified by $x = 0$ and $\theta_1 = \theta_2 = 0$.

By small angle approximation,

$$\dot{\theta}^2 = 0, sin(\theta) = \theta, cos(\theta) = 1$$

The limiting conditions are as follows:

$$sin(\theta_1) \approx \theta_1, sin(\theta_2) \approx \theta_1, cos(\theta_1) \approx 1, cos(\theta_2) \approx 1, \dot{\theta}_1^2 \approx 0, \dot{\theta}_2^2 \approx 0 \tag{7}$$

and from equation 4 and 7 we get,

$$\therefore F(t) = (M + m_1 + m_2)\ddot{x} - m_1 l_1 \ddot{\theta}_1 - m_2 l_2 \ddot{\theta}_2 \tag{8}$$

From equation 5 and 7 we get,

$$m_1 l_1^2 \ddot{\theta}_1 - m_1 l_1 \ddot{x} + m_1 g l_1 \theta_1 = 0$$

Dividing above equation by $m_1 l_1$ we get,

$$l_1 \ddot{\theta}_1 - \ddot{x} + g\theta_1 = 0$$

From equation 6 and 7 we get,

$$m_2 l_2^2 \ddot{\theta}_2 - m_2 l_2 \ddot{x} + m_2 g l_2 \theta_2 = 0$$

Dividing above equation by $m_2 l_2$ we get,

$$l_2 \ddot{\theta}_2 - \ddot{x} + g\theta_2 = 0$$

Rearranging these equations we get,

$$\ddot{x} = g\theta_1 + l_1 \ddot{\theta}_1 \tag{9}$$

$$\ddot{x} = g\theta_2 + l_2 \ddot{\theta}_2 \tag{10}$$

Substituting equation 9 in 10 we get,

$$\therefore \ddot{\theta}_2 = \frac{1}{l_2}\left(-g\theta_2 + g\theta_1 + l_1 \ddot{\theta}_1\right) \tag{11}$$

Substituting equation 10 in equation 8, we get,

$$F(t) = (M + m_1 + m_2)(g\theta_1 + l_1\ddot{\theta}_1) - m_1 l_1 \ddot{\theta}_1 - m_2(-g\theta_2 + g\theta_1 + l_1\ddot{\theta}_1)$$

Further simplifying,

$$F(t) = (M + m_1 + m_2)(g\theta_1 + l_1\ddot{\theta}_1) - m_1 l_1 \ddot{\theta}_1 + m_2 g\theta_2 - m_2 g\theta_2 - m_2 g\theta_1 - m_2 l_1 \ddot{\theta}_1$$

$$\therefore F(t) = (M + m_1 + m_2)g\theta_1 + M l_1 \ddot{\theta}_1 + m_2 g\theta_2 - m_2 g\theta_1$$

$$\therefore \ddot{\theta}_1 = \frac{1}{M}F(t) - \left(\frac{(M + m_1 + m_2)g}{M l_1}\right)\theta_1 + \frac{m_2 g\theta + 1}{M l_1} - \frac{m_2 g\theta_2}{M l_1}$$

Rearranging the terms;

$$\therefore \ddot{\theta}_1 = -\frac{(M + m_1)}{M l_1} g\theta_1 - \frac{m_2}{M l_1} g\theta_2 + \frac{1}{M l_1}F(t) \tag{12}$$

Substituting equation 12 in equation 11, we get,

$$\therefore \ddot{\theta}_2 = -\frac{(M + m_2)}{M l_2} g\theta_2 - \frac{m_2}{M l_2} g\theta_2 + \frac{1}{M l_2}F(t) \tag{13}$$

Substituting equation 13 in equation 10, we get,

$$\therefore \ddot{x} = -\frac{m_1}{M} g\theta_1 - \frac{m_2}{M} g\theta_1 + \frac{1}{M}F(t) \tag{14}$$

Let the state vector be as follows:

$$X(t) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} = \begin{bmatrix} x \\ \dot{x} \\ \theta_1 \\ \dot{\theta}_1 \\ \theta_2 \\ \dot{\theta}_1 \end{bmatrix} \tag{15}$$

The state space representation is give as from equations 12, 13 and 14,

$$\dot{X}(t) = AX(t) + BF(t)$$

$$\therefore \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_2 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta}_1 \\ \ddot{\theta}_1 \\ \dot{\theta}_2 \\ \ddot{\theta}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{g\,m_1}{M} & 0 & -\frac{g\,m_2}{M} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{g\,(M+m_1)}{M\,l_1} & 0 & -\frac{g\,m_2}{M\,l_1} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{g\,m_1}{M\,l_2} & 0 & -\frac{g\,(M+m_2)}{M\,l_2} & 0 \end{bmatrix} X(t) + \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ \frac{1}{M\,l_1} \\ 0 \\ \frac{1}{M\,l_2} \end{bmatrix} F(t)$$

where,

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{g\,m_1}{M} & 0 & -\frac{g\,m_2}{M} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{g\,(M+m_1)}{M\,l_1} & 0 & -\frac{g\,m_2}{M\,l_1} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{g\,m_1}{M\,l_2} & 0 & -\frac{g\,(M+m_2)}{M\,l_2} & 0 \end{bmatrix} and, B = \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ \frac{1}{Ml_1} \\ \\ \frac{1}{Ml_2} \end{bmatrix}$$

4 Controllability of the system

For Linear Time Invariant system the Grammian of Controllability $W_c(0, t*)$ is is invertible if and only if the n cross nm controllability matrix satisfies the rank condition i.e.

$$rank(\begin{bmatrix} B_k & AB_k & A^2B_k & .... & A^{n-1}B_k \end{bmatrix} = n$$

For our case the condition becomes as follows:

$$rank(\begin{bmatrix} B_k & AB_k & A^2B_k & A^3B_k & A^4B_k & A^5B_k \end{bmatrix} = 6$$

Following code is written to check whether the system is controllable or not:

```
% Declaring variables to derive the state space representation of system
% M is the Mass of cart
% m1 is the mass attached to pendulum 1
% m2 is the mass attached to pendulum 2
% l1 is the length of pendulum 1
% l2 is the length of pendulum 2
% g is the accleration due to gravity
syms M m1 m2 l1 l2 g;
% Declaring the A matrix of the system
A = [0 1 0 0 0 0;
     0 0 -(m1*g)/M 0 -(m2*g)/M 0;
     0 0 0 1 0 0;
     0 0 -((M+m1)*g/(M*l1)) 0 -(m2*g)/(M*l1) 0;
     0 0 0 0 0 1;
     0 0 -(m1*g)/(M*l2) 0 -((M+m2)*g/(M*l2)) 0];

% Declaring the B matrix of the system
B = [0; 1/M; 0; 1/(M*l1); 0; 1/(M*l2)];
% disp(B)

% Defining the controlability matrix in terms of system variables
ctrb_sym = [B A*B (A^2)*B (A^3)*B (A^4)*B (A^5)*B];
% print("The controllability matrix is " )
disp("The controllability matrix is ")
disp(ctrb_sym)
```

```
% Finding the rank of controlability matrix
disp("The rank of controllability matrix is: ")
rank(ctrb_sym)

ctrb_det = simplify(det(ctrb_sym));
disp("The determinant of controllability matrix is: ")
disp(ctrb_det)

% The part for checking rank and determinant when the system is not
% controllable
disp("We can see that if l1 = l2, the rank will be less
than 6 and determinant will be zero, lets verify that")
ctrb_sub = subs(ctrb_sym, l1, l2)
disp("The rank after putting l1 = l2 is: ")
disp(rank(ctrb_sub))
disp("The determinant after putting l1 = l2 is: ")
disp(simplify(det(ctrb_sub)))
```

4.1 Output

```
The rank of controllability matrix is:
ans = 6
```

The determinant of controllability matrix is:

$$\frac{-g_6(l_1 - l_2)^2}{M^6 l_1^6 l_2^6}$$

```
We can see that if l1 = l2, the rank will be less
than 6 and determinant will be zero, lets verify that
The rank after putting l1 = l2 is:
     4
The determinant after putting l1 = l2 is:
0
```

Controllability matrix is given as follows:

$$\begin{pmatrix} 0 & \frac{1}{M} & 0 & \sigma_2 & 0 & \sigma_1 \\ \frac{1}{M} & 0 & \sigma_2 & 0 & \sigma_1 & 0 \\ 0 & \frac{1}{M\,l_1} & 0 & \sigma_6 & 0 & \sigma_4 \\ \frac{1}{M\,l_1} & 0 & \sigma_6 & 0 & \sigma_4 & 0 \\ 0 & \frac{1}{M\,l_2} & 0 & \sigma_5 & 0 & \sigma_3 \\ \frac{1}{M\,l_2} & 0 & \sigma_5 & 0 & \sigma_3 & 0 \end{pmatrix}$$

where

$$\sigma_1 = \frac{\frac{g^2\,m_1\,(M+m_1)}{M^2\,l_1}+\frac{g^2\,m_1\,m_2}{M^2\,l_2}}{M\,l_1} + \frac{\frac{g^2\,m_2\,(M+m_2)}{M^2\,l_2}+\frac{g^2\,m_1\,m_2}{M^2\,l_1}}{M\,l_2}$$

$$\sigma_2 = -\frac{g\,m_1}{M^2\,l_1} - \frac{g\,m_2}{M^2\,l_2}$$

$$\sigma_3 = \frac{\frac{g^2\,m_1\,(M+m_2)}{M^2\,{l_2}^2}+\frac{g^2\,m_1\,(M+m_1)}{\sigma_7}}{M\,l_1} + \frac{\frac{g^2\,(M+m_2)^2}{M^2\,{l_2}^2}+\frac{g^2\,m_1\,m_2}{\sigma_7}}{M\,l_2}$$

$$\sigma_4 = \frac{\frac{g^2\,m_2\,(M+m_1)}{M^2\,{l_1}^2}+\frac{g^2\,m_2\,(M+m_2)}{\sigma_7}}{M\,l_2} + \frac{\frac{g^2\,(M+m_1)^2}{M^2\,{l_1}^2}+\frac{g^2\,m_1\,m_2}{\sigma_7}}{M\,l_1}$$

$$\sigma_5 = -\frac{g\,(M+m_2)}{M^2\,{l_2}^2} - \frac{g\,m_1}{\sigma_7}$$

$$\sigma_6 = -\frac{g\,(M+m_1)}{M^2\,{l_1}^2} - \frac{g\,m_2}{\sigma_7}$$

$$\sigma_7 = M^2\,l_1\,l_2$$

It can be seen that the system is not controllable when $l_1 = l_2$ and $l_1 = l_2 = 0$.

5 LQR Controller

For Linear Quadratic Regular Controller if the pair (A, B_k) is stabilizable then we look for K which minimizes the following cost:

$$J(K, X(0)) = \int_0^\infty X(t)^T Q X(t) + U_k(t)^T R U_k(t)\,dt$$

where Q and R are symmetric positive definite matrices. To simulate the system different values of Q and R are used and output graphs are given to validate the results. The optimal solution is given by:

$$K = R^{-1} B_k^T P$$

where P is the symmetric positive definite solution of the following stationary Ricatti equation

$$A^T P + PA - PB_k R^{-1} B_k P = -Q$$

When the R is minimum stability of the system is achieved faster as less energy is utilized. Q decides the error band within the system should work.

5.1 Checking the controllability

The following MATLAB code is used to find the controllability matrix after the values of system parameters are substituted. It then checks the rank of the controllability matrix and prints if the system is controllable or not. Subsequently, using the command lqr(A,B,Q,R) the optimal feedback gain matrix - 'K' is obtained. Using the optimal K, a closed-loop system is generated and simulated on the initial condition: $\theta_1 = 0.5 radians$ and $\theta_2 = 0.6 radians$.

The optimal Q matrix is:

$$Q = \begin{bmatrix} 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 & 0 \\ 0 & 0 & 100 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

The optimal R matrix is: $R = 0.001$

The code is given below:

```
% Declaring variables to derive the state space representation of system
% M is the Mass of cart
% m1 is the mass attached to pendulum 1
% m2 is the mass attached to pendulum 2
% l1 is the length of pendulum 1
% l2 is the length of pendulum 2
% g is the accleration due to gravity
syms M m1 m2 l1 l2 g;

% Declaring the values of system variables
M_val = 1000;
m1_val = 100;
m2_val = 100;
l1_val = 20;
l2_val = 10;

% Declaring the A matrix of the system
A = [0 1 0 0 0 0;
     0 0 -(m1*g)/M 0 -(m2*g)/M 0;
     0 0 0 1 0 0;
     0 0 -((M+m1)*g/(M*l1)) 0 -(m2*g)/(M*l1) 0;
     0 0 0 0 0 1;
     0 0 -(m1*g)/(M*l2) 0 -((M+m2)*g/(M*l2)) 0];
```

```
% Declaring the B matrix of the system
B = [0; 1/M; 0; 1/(M*l1); 0; 1/(M*l2)];
% disp(B)

% Declaring the C matrix
C = eye(6);
D = [0; 0; 0; 0; 0; 0];

% The part for checking the rank and determinant of controllability matrix after
% evaluating the system as per the values of system properties
ctrb_val = subs(ctrb_sym, {M, m1, m2, l1, l2, g}, {M_val, m1_val, m2_val,
l1_val, l2_val, 9.81});
disp("The controllability matrix obtained is")
disp(simplify(ctrb_val))

% Rank of controlability matrix as per the values of system properties
disp("The rank of controlability matrix after putting the system properties is: ")
disp(rank(ctrb_val))

% Checking and printing if the system,
% with the given parameters of part D, is controllable
if(rank(ctrb_val == 6))
    disp("The controllability matrix is full rank, hence the system
    is controllable")
else
    disp("The controllability matrix is rank deficient, hence the
    system is not controllable")
end
```

The output controllability matrix is given as follows:

$$\begin{bmatrix} 0 & 0.0001 & 0 & -0.000147 & 0 & 0.000141948 \\ 0.0001 & 0 & -0.00014715 & 0 & 0.000141948 & 0 \\ 0 & 0.00005 & 0 & -0.00003188 & 0 & 0.000022735 \\ 0.00005 & 0 & -0.00003188 & 0 & 0.000022735 & 0 \\ 0 & 0.00001 & 0 & -0.0001128 & 0 & 0.000124866 \\ 0.00001 & 0 & -0.000112815 & 0 & 0.00012486 & 0 \end{bmatrix}$$

5.2 Simulating the resulting response to initial conditions when the controller is applied to linearized system

```
% Defining the Q and R matrices for LQR controller
Q = [10  0  0  0  0  0;
     0  10  0  0  0  0;
     0   0 100 0  0  0;
```

```
     0   0  0  1  0  0;
     0   0  0  0 100 0;
     0   0  0  0  0  1];

R = 0.001;

% Taking the A and B matrices as per the specified system parameters in
% part D
A_val = [0 1 0 0 0 0;
     0 0 -(m1_val*9.81)/M_val 0 -(m2_val*9.81)/M_val 0;
     0 0 0 1 0 0;
     0 0 -((M_val+m1_val)*9.81/(M_val*l1_val)) 0 -(m2_val*9.81)/(M_val*l1_val) 0;
     0 0 0 0 0 1;
     0 0 -(m1_val*9.81)/(M_val*l2_val) 0 -((M_val+m2_val)*9.81/(M_val*l2_val)) 0];

% Declaring the B matrix of the system
B_val = [0; 1/M_val; 0; 1/(M_val*l1_val); 0; 1/(M_val*l2_val)];

% Finding the optimal feedback gain K for the LQR controller, and R_soln
% corresponding to the solution of Ricatti equation and the poles of the
% closed loop control system
[K, R_soln, poles] = lqr(A_val,B_val,Q,R);

% Making the closed loop system after finding the optimal 'K' feedback gain
lqr_sys = ss(A_val-(B_val*K), B_val, C, D);

% Initial condition
x_0 = [0; 0; 0.5; 0; 0.6; 0];

% The plot for the linearized system before applying LQR control
% disp("The plot for the linearized system before applying LQR control")
% figure
% initial(orig_sys, x_0)

% The plot for the linearized system after applying LQR control
disp("The plot for the linearized system after applying LQR control")
figure
initial(lqr_sys, x_0)
```

5.3 Lyapunov's indirect method to certify stability

The following summarizes Lyapunov's indirect method:

- Linearize the original system around the equilibrium point of interest.
- Check the eigenvalues of AF . The following describes how to use the test:

- if the eigenvalues of AF have negative real part then the original system is at least locally stable around the equilibrium pint. In this case, a Lyapunov function for the linearized system will be valid at least locally.
- if at least one eigenvalue of AF is positive then the original system is unstable around the equilibrium point.
- if the eigenvalues of AF have non-positive real part, but at least one is on the imaginary axis then the indirect method is inconclusive.

The eigen values of $(A + B_k K)$ are as follows:
$-0.0102 + 0.7277i$
-0.0102 - 0.7277i
-0.0178 + 1.0423i
-0.0178 - 1.0423i
-0.2066 + 0.2023i
-0.2066 - 0.2023i

As all poles are in the left half plane, the closed loop system is stable.
Following code is written for Lyapunov's indirect method.

```
% Using Lyapunov's indirect method to certify stability of the closed-loop
% system
disp("The poles of the closed loop control system are")
disp(poles)

% Initializing a count to keep track of stable poles
count_stable_pole = 0;

% Checking the real part of each pole
for  i = 1 : length(poles)
    if(real(poles(i)) < 0)
        count_stable_pole = count_stable_pole + 1;
    end
end

% Printing the stability of the system based on the location of poles
if(count_stable_pole == 6)
    disp("As all poles are in the left half plane, the closed loop system is stable")
else
    disp("As all the poles are not in the left half plane, the closed loop system is n
end
```

5.4 Output for linearized system

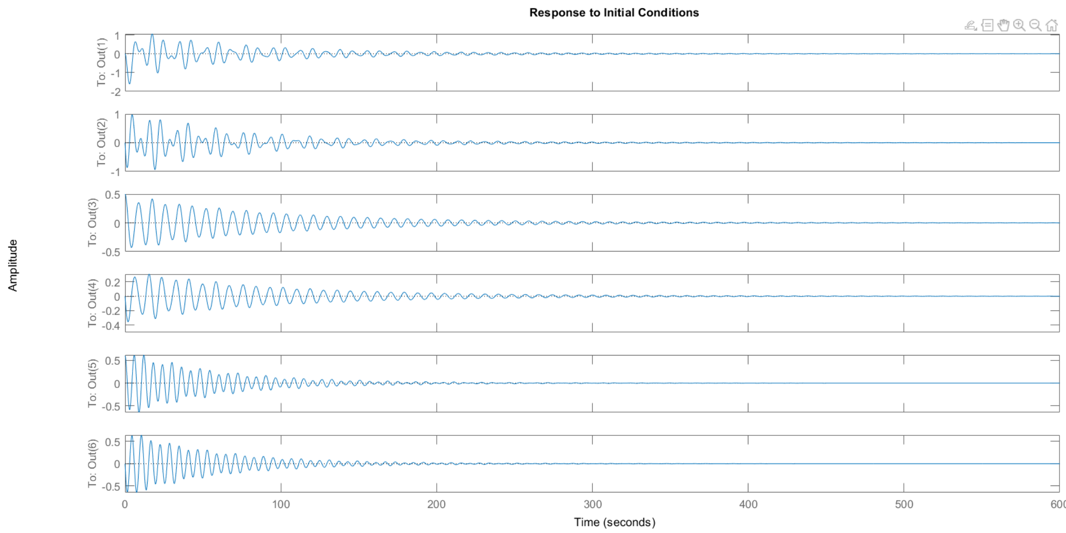

Figure 2: Response to initial conditions

5.5 Simulating the resulting response to initial conditions when the controller is applied to non - linearized system

The code written for non linear system is given as follows:

```
t_span = 0:0.01:600;
x_0 = [0; 0; 0.5; 0; 0.6; 0];
[ts, x_dots] = ode45(@non_lin_sys, t_span, x_0);
figure
plot(ts, x_dots)
xlabel('Time in seconds')
ylabel('Outputs')
legend()
legend({'x','x_d', 'theta1', 'theta1_d', 'theta2', 'theta2_d'},'Location','southeast')

function x_dot = non_lin_sys(t1, x)
    x_dot = zeros(6,1);

    % Declaring the values of system variables
    M_val = 1000;
    m1_val = 100;
    m2_val = 100;
    l1_val = 20;
    l2_val = 10;
    g_val = 9.81;
```

```
    % Declaring the A matrix of the system
    A_val = [0 1 0 0 0 0;
         0 0 -(m1_val*g_val)/M_val 0 -(m2_val*g_val)/M_val 0;
         0 0 0 1 0 0;
         0 0 -((M_val+m1_val)*g_val/(M_val*l1_val)) 0 -(m2_val*g_val)/(M_val*l1_val) 0
         0 0 0 0 0 1;
         0 0 -(m1_val*g_val)/(M_val*l2_val) 0 -((M_val+m2_val)*g_val/(M_val*l2_val)) 0

    % Declaring the B matrix of the system
    B_val = [0; 1/M_val; 0; 1/(M_val*l1_val); 0; 1/(M_val*l2_val)];

    % Defining the Q and R matrices for LQR controller
    Q = [10  0  0   0  0   0;
         0  10  0   0  0   0;
         0   0 100  0  0   0;
         0   0  0   1  0   0;
         0   0  0   0 100  0;
         0   0  0   0  0   1];

    R = 0.001;
    K = lqr(A_val,B_val,Q,R);
    fb = -K*x;

    % State x_dot_1 is x_2
    x_dot(1) = x(2);
    % State x_dot_2 is
    x_dot(2) = (fb - (g_val/2)*(m1_val*sin(2*x(3)) + m2_val*sin(2*x(5))) - (m1_val*l1_
    % State x_dot_3 is theta 1
    x_dot(3) = x(4);
    % State x_dot_4 is theta1 dot
    x_dot(4) = (x_dot(2)*cos(x(3)) - g_val*(sin(x(3))))/l1_val';
    % State x_dot_5 is theta2
    x_dot(5) = x(6);
    % State x_dot_6 is theta2 dot
    x_dot(6) = (x_dot(2)*cos(x(5)) - g_val*(sin(x(5))))/l2_val;

end
```

5.6 Output for non linearized sytem

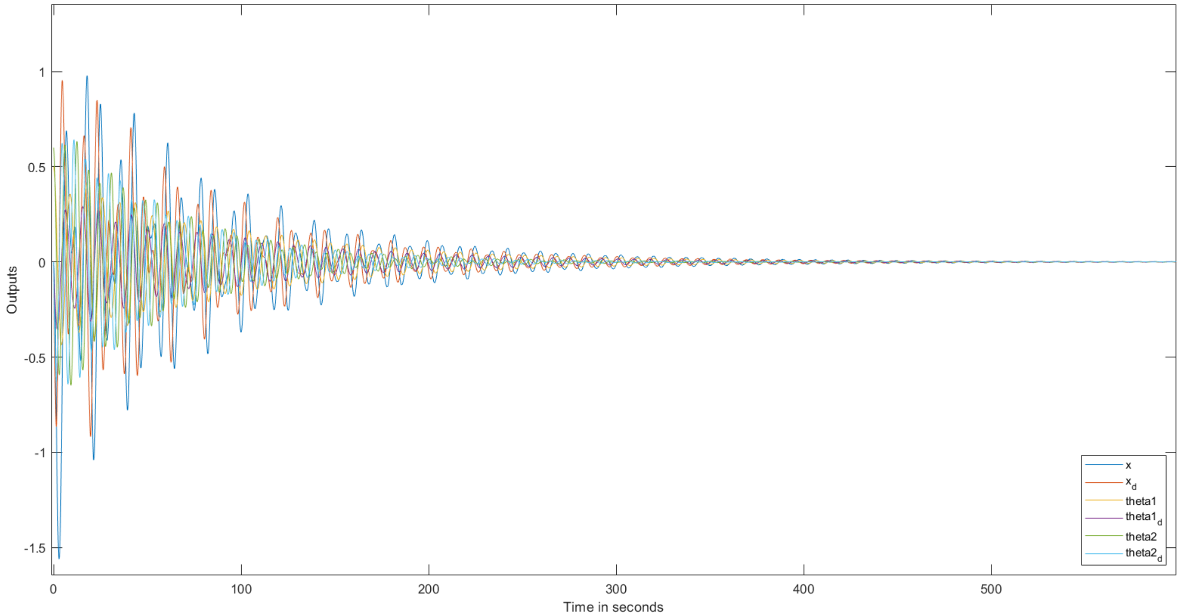

Figure 3: Response to initial conditions

6 Observability

Observability is the ability to measure the internal states of a system by examining its outputs. A system is considered observable if the current state can be estimated by only using specific state information from outputs that are measured using sensors of particular states. For the LTI (Linear Time Invariant) systems, the system is observable if the $np * n$ observability matrix is full rank. That is,

$$rank\left(\begin{bmatrix} C \\ CA \\ CA^2 \\ . \\ . \\ . \\ CA^{n-1} \end{bmatrix}\right) = n$$

For our case the condition becomes as follows:

$$rank\left(\begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \\ CA^4 \\ CA^5 \end{bmatrix}\right) = 6$$

Obtaining the C matrices for the given output vectors:

1. $x(t)$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

2. $(\theta_1(t), \theta_2(t)$

$$C = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

3. $(x(t), \theta_2(t)$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

4. $(x, \theta_1(t), \theta_2(t))$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

The matlab code used to find the observability matrices corresponding to each C matrix is:

```
% The controllability matrix when output vector is x(t)
C1 = [1 0 0 0 0 0];

% The controllability matrix when output vector is (theta1(t), theta2(t))
C2 = [0 0 1 0 0 0;
      0 0 0 0 1 0];

% The controllability matrix when output vector is (x(t), theta2(t))
C3 = [1 0 0 0 0 0;
      0 0 0 0 1 0];

% The controllability matrix when output vector is (x(t), theta1(t), theta2(t))
C4 = [1 0 0 0 0 0;
      0 0 1 0 0 0;
      0 0 0 0 1 0];

% Making the observability matrices for each C matrix
ob1_param = [C1' A'*C1' ((A')^2)*C1' ((A')^3)*C1' ((A')^4)*C1' ((A')^5)*C1'];
ob2_param = [C2' A'*C2' ((A')^2)*C2' ((A')^3)*C2' ((A')^4)*C2' ((A')^5)*C2'];
ob3_param = [C3' A'*C3' ((A')^2)*C3' ((A')^3)*C3' ((A')^4)*C3' ((A')^5)*C3'];
ob4_param = [C4' A'*C4' ((A')^2)*C4' ((A')^3)*C4' ((A')^4)*C4' ((A')^5)*C4'];
```

```
ob1_rank = rank(ob1_param);
ob2_rank = rank(ob2_param);
ob3_rank = rank(ob3_param);
ob4_rank = rank(ob4_param);

% Checking for the case when x(t) is output vector
disp("The rank of observability matrix when x(t) is output vector is:")
disp(ob1_rank)
if(ob1_rank == 6)
    disp("As the observability matrix is full rank, the system is observable")
else
    disp("As the observability matrix is rank deficient, the system is not observable")
end

% Checking for the case when theta1(t) and theta(2) are in output vector
disp("The rank of observability matrix when theta1(t) and theta(2) are in output vector is:")
disp(ob2_rank)
if(ob2_rank == 6)
    disp("As the observability matrix is full rank, the system is observable")
else
    disp("As the observability matrix is rank deficient, the system is not observable")
end

% Checking for the case when x(t) and theta(2) are in output vector
disp("The rank of observability matrix when x(t) and theta2(t) are in output vector is:")
disp(ob3_rank)
if(ob3_rank == 6)
    disp("As the observability matrix is full rank, the system is observable")
else
    disp("As the observability matrix is rank deficient, the system is not observable")
end

% Checking for the case when x(t), theta1(t) and theta(2) are in output vector
disp("The rank of observability matrix when x(t), theta1(t) and theta(2) is output vector is:")
disp(ob4_rank)
if(ob4_rank == 6)
    disp("As the observability matrix is full rank, the system is observable")
else
    disp("As the observability matrix is rank deficient, the system is not observable")
end
```

The output of this code finds the observability matrices, prints the rank of each observability matrix, and depending on the rank of observability matrix for the corresponding output vector, it prints if the system is observable or not. The following output is obtained:

```
The rank of observability matrix when x(t) is output vector is:
     6
As the observability matrix is full rank, the system is observable

The rank of observability matrix when theta1(t) and theta(2) are in output vector is:
```

```
     4
As the observability matrix is rank deficient, the system is not observable

The rank of observability matrix when x(t) and theta2(t) are in output vector is:
     6
As the observability matrix is full rank, the system is observable

The rank of observability matrix when x(t), theta1(t) and theta(2) is output vector is:
     6
As the observability matrix is full rank, the system is observable
```

7 Luenberger Observer

Luenberger Observer is given by following state space representation:

$\dot{\hat{X}}(t) = A\hat{X}(t) + B_k U_k(t) + L(Y(t) - C\hat{X}(t)$

where L is observer gain and $Y(t) - C\hat{X}(t)$ $\hat{X}(t)$ is the estimated state for output $Y(t)$ observer gain L can be found out by placing poles of $(A - LC)$ in negative left half plane. The following code is written for the same:

7.1 Main code of Luenberger Observer

```
M_val = 1000;
m1_val = 100;
m2_val = 100;
l1_val = 20;
l2_val = 10;

% Defining the Q and R matrices for LQR controller
Q = [10  0   0   0   0   0;
     0  10   0   0   0   0;
     0   0 100   0   0   0;
     0   0   0   1   0   0;
     0   0   0   0 100   0;
     0   0   0   0   0   1];

R = 0.001;

% Declaring the A matrix of the system
A_val = [0 1 0 0 0 0;
         0 0 -(m1_val*9.81)/M_val 0 -(m2_val*9.81)/M_val 0;
         0 0 0 1 0 0;
         0 0 -((M_val+m1_val)*9.81/(M_val*l1_val)) 0 -(m2_val*9.81)/(M_val*l1_val) 0;
         0 0 0 0 0 1;
         0 0 -(m1_val*9.81)/(M_val*l2_val) 0 -((M_val+m2_val)*9.81/(M_val*l2_val)) 0];

% Declaring the B matrix of the system
B_val = [0; 1/M_val; 0; 1/(M_val*l1_val); 0; 1/(M_val*l2_val)];


% Finding the optimal feedback gain K for the above LQR controller
[K, R_soln, poles] = lqr(A_val,B_val,Q,R);
```

```
% The controllability matrix when output vector is x(t)
C1 = [1 0 0 0 0 0];

% The controllability matrix when output vector is (theta1(t), theta2(t))
C2 = [0 0 1 0 0 0;
      0 0 0 0 1 0];

% The controllability matrix when output vector is (x(t), theta2(t))
C3 = [1 0 0 0 0 0;
      0 0 0 0 1 0];

% The controllability matrix when output vector is (x(t), theta1(t), theta2(t))
C4 = [1 0 0 0 0 0;
      0 0 1 0 0 0;
      0 0 0 0 1 0];

% For the A matrix after putting parameter values
ob1 = obsv(A_val, C1);
ob2 = obsv(A_val, C2);
ob3 = obsv(A_val, C3);
ob4 = obsv(A_val, C4);

% The poles of the Luenberger observer
L_poles = [-4; -4.5; -5; -5.5; -6; -6.5];
% Specifying the initial condition where estimated states
% are initialized to zero
x0 = [0;0;0.5;0;0.6;0;0;0;0;0;0;0];

% The observer gains for each observer, when the observer poles are placed
% at L_poles
L1 = place(A_val', C1', L_poles);
L3 = place(A_val', C3', L_poles);
L4 = place(A_val', C4', L_poles);
```

7.2 Observers

7.2.1 First observer: $X(t)$ is observed

```
% Luenberger A matrix for first observer
A_c1 = [(A_val - B_val*K) B_val*K ; zeros(size(A_val)) (A_val - (L1'*C1))];
% Luenberger B matrix for first observer
B_c1 = [B_val; B_val];
% Luenberger C and D matrix for first observer
C_c1 = [C1 zeros(size(C1))];
D1 = 0;

% Checking the response of system when x(t) is observed
sys_ob1 = ss(A_c1, B_c1, C_c1, D1);

% The Luenberger observer obtained when x(t) is observed
disp("The Luenberger observer of the system when x(t) is observed ")
```

```
disp(L1);

% The response of linearized system to initial conditions
% theta1 = 0.5 radians and theta2 = 0.6 radians
initial(sys_ob1, x0)
% Response of system when unit step is given as input
step(sys_ob1)
```

7.2.2 Third observer: $X(t)$ and θ_2 are observed

```
 % The Luenberger observer of the system when x(t) and
% theta2(t) are observed
disp("The Luenberger observer of the system when x(t) and theta2(t) are observed ")
disp(L3);

% Luenberger A matrix for third observer
A_c3 = [(A_val - B_val*K) B_val*K ; zeros(size(A_val)) (A_val - (L3'*C3))];
% Luenberger B matrix for third observer
B_c3 = [B_val; B_val];
% Luenberger C and D matrix for third observer
C_c3 = [C3 zeros(size(C3))];
D3 = 0;

% Checking the response of system when x(t) and
% theta2(t) are observed
sys_ob3 = ss(A_c3, B_c3, C_c3, D3);

% The response of linearized system to initial conditions
% theta1 = 0.5 radians and theta2 = 0.6 radians
initial(sys_ob3, x0)
% Response of system when unit step is given as input
step(sys_ob3)
```

7.2.3 Fourth Observer: $\theta_1(t) and \theta_2(t)$ are observed

```
% The Luenberger observer of the system when x(t),
% theta1(t) and theta2(t) are observed
disp("The Luenberger observer of the system when x(t), theta1(t) and theta2(t) are observed ")
disp(L4);

% Luenberger A matrix for fourth observer
A_c4 = [(A_val - B_val*K) B_val*K ; zeros(size(A_val)) (A_val - (L4'*C4))];
% Luenberger B matrix for fourth observer
B_c4 = [B_val; B_val];
% Luenberger C and D matrix for fourth observer
C_c4 = [C4 zeros(size(C4))];
D4 = 0;

% Checking the response of system when x(t), theta1(t) and
% theta2(t) are observed
sys_ob4 = ss(A_c4, B_c4, C_c4, D4);
```

```
% The response of linearized system to initial conditions
% theta1 = 0.5 radians and theta2 = 0.6 radians
initial(sys_ob4, x0)
% Response of system when unit step is given as input
step(sys_ob4)
```

7.3 Outputs of the above observers

The Luenberger observer of the system when $X(t)$ is observed (first observer)

$$e^4 * \begin{bmatrix} 0.0031 & 0.0410 & -4.4234 & -2.9057 & 4.1383 & 1.8485 \end{bmatrix}$$

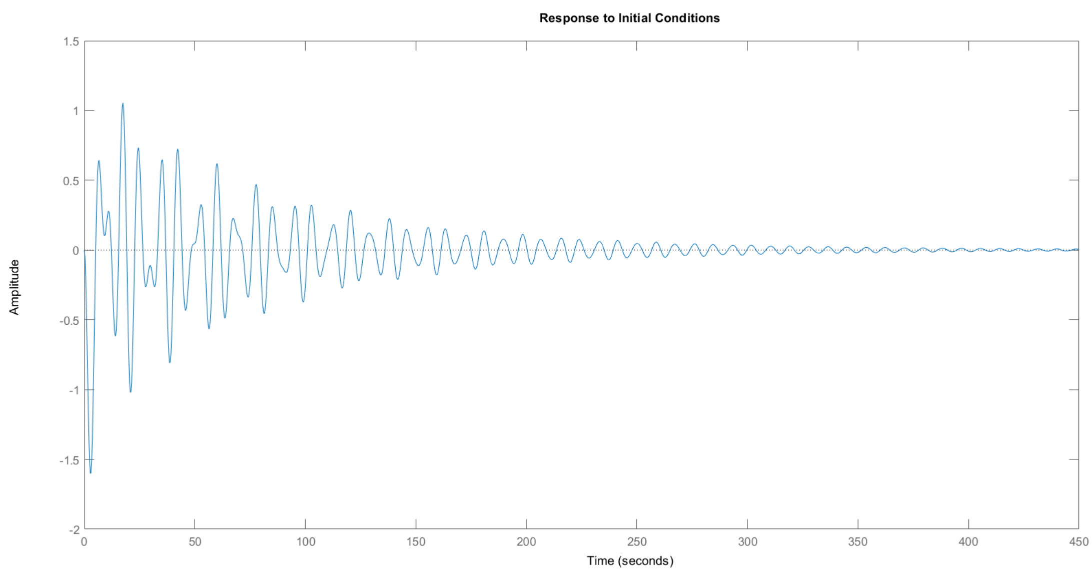

Figure 4: Response to initial condition for the first observer

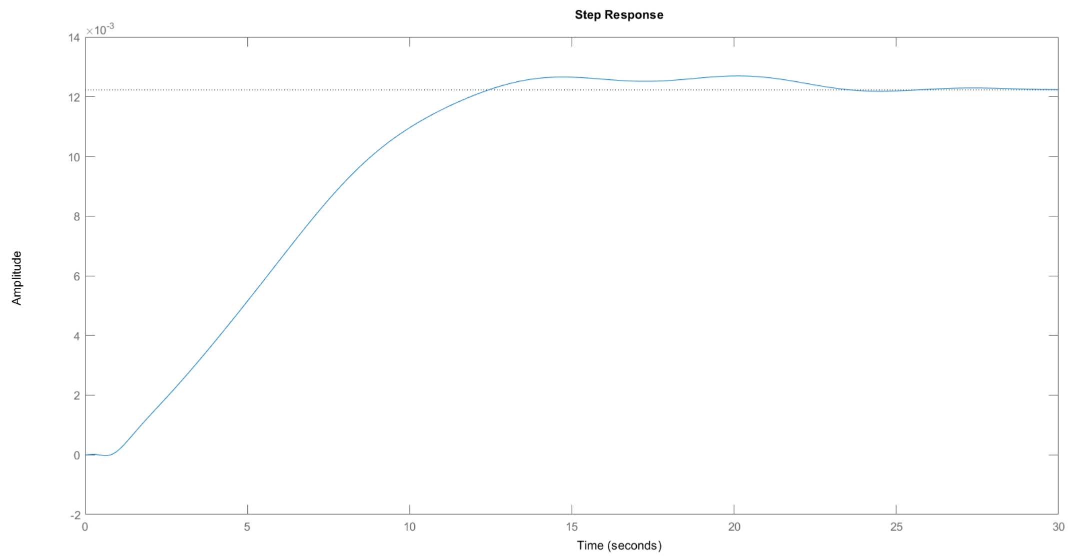

Figure 5: Response to unit step input for the first observer

The Luenberger observer of the system when $X(t)$ and $\theta_2(t)$ are observed (third observer)

$$e^3 \begin{bmatrix} 0.0192 & 0.1352 & -0.4165 & -0.4116 & 0.0009 & 0.0108 \\ 0.0158 & 0.2445 & -1.2641 & -1.9875 & 0.0123 & 0.0519 \end{bmatrix}$$

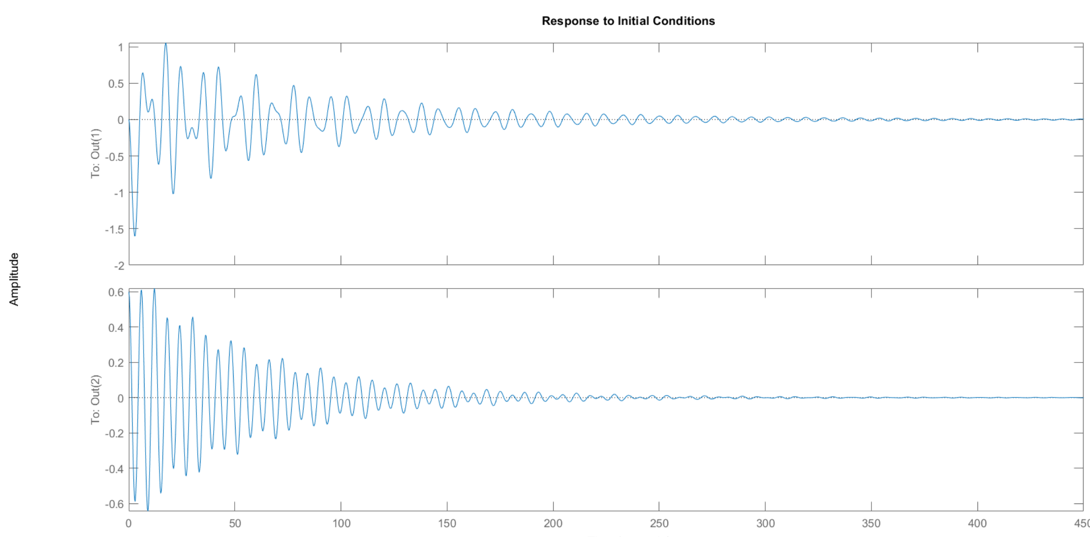

Figure 6: Response to initial condition for the third observer

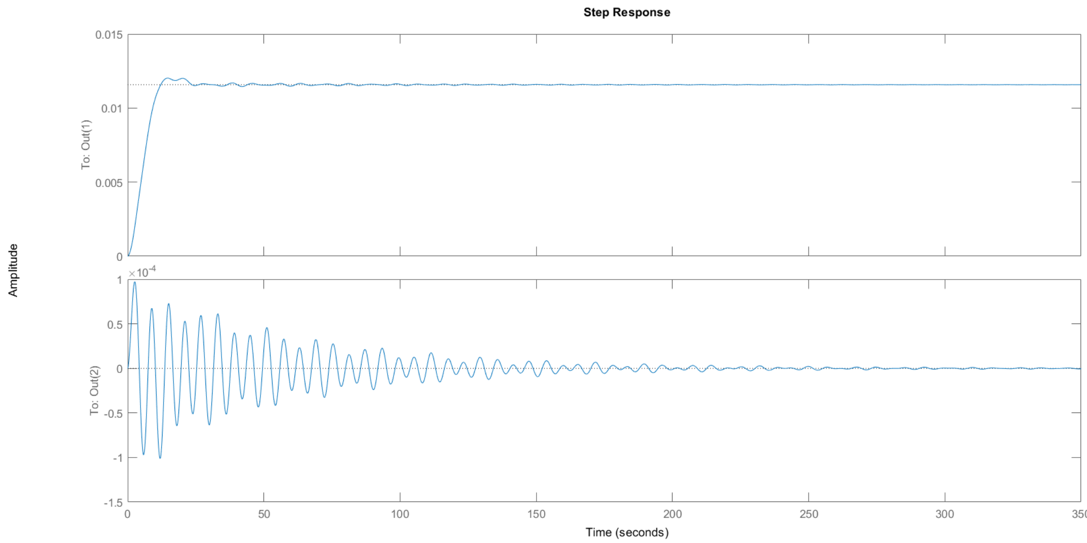

Figure 7: Response to unit step input of the third observer

The Luenberger observer of the system when $X(t)$, $\theta_1(t)$ and $\theta_1(t)$ are observed (fourth observer)

$$\begin{bmatrix} 11.5848 & 33.3556 & 0.5072 & 2.9226 & 0.0000 & 0.0000 \\ 0.4784 & 1.7664 & 11.4152 & 31.8547 & 0.0000 & -0.0981 \\ 0.0000 & -0.9809 & 0.0000 & -0.0489 & 8.5000 & 16.9209 \end{bmatrix}$$

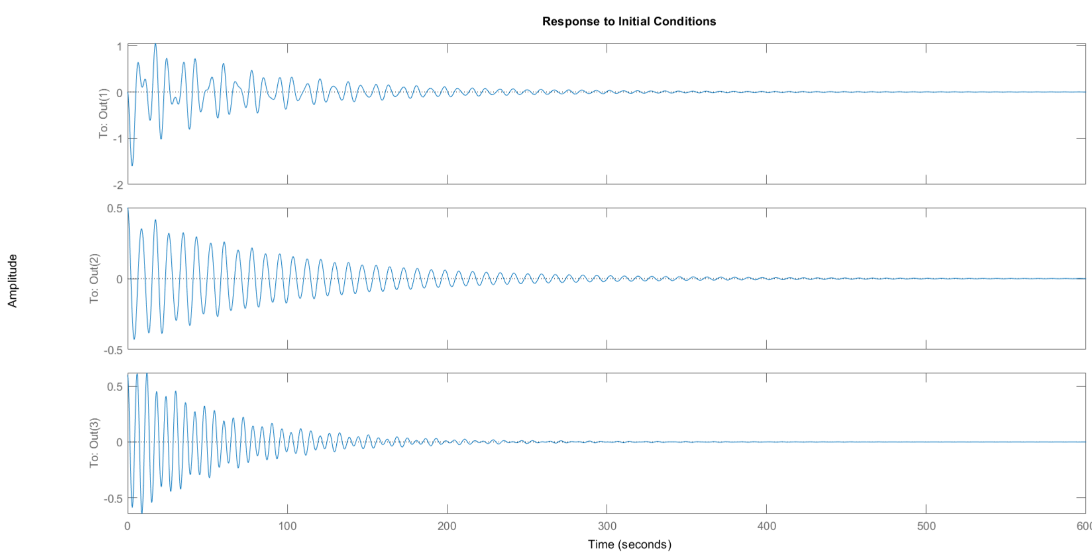

Figure 8: Response to initial condition for fourth observer

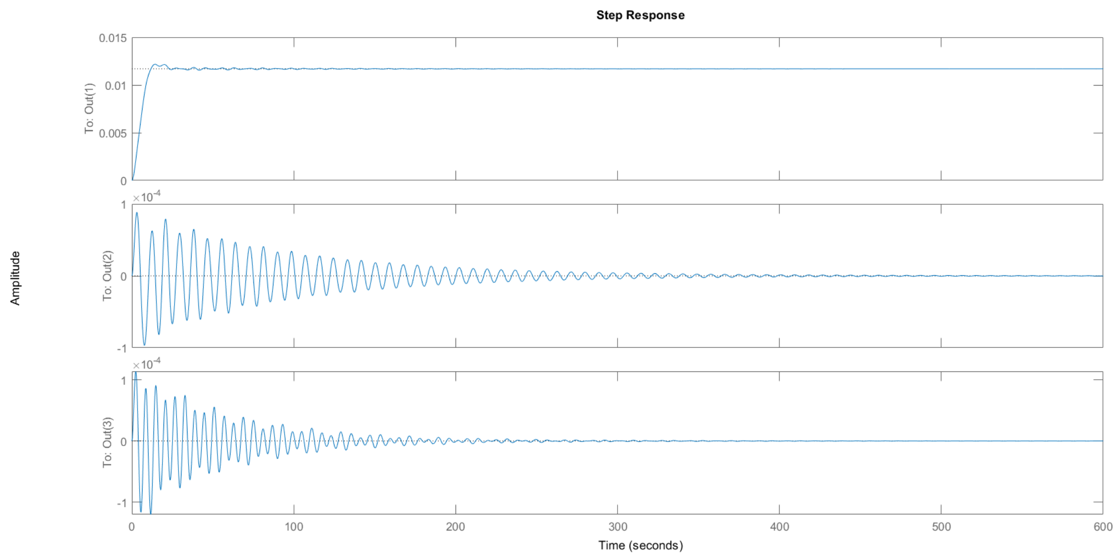

Figure 9: Response to unit step input for fourth observer

7.4 Outputs of the non linear system

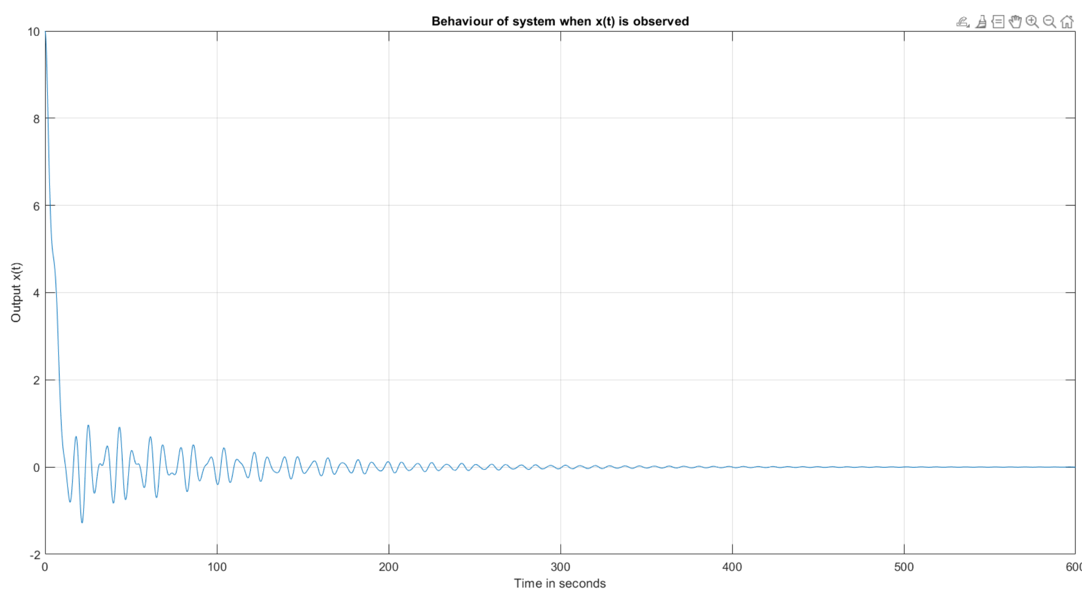

Figure 10: Response of non-linear system for first observer

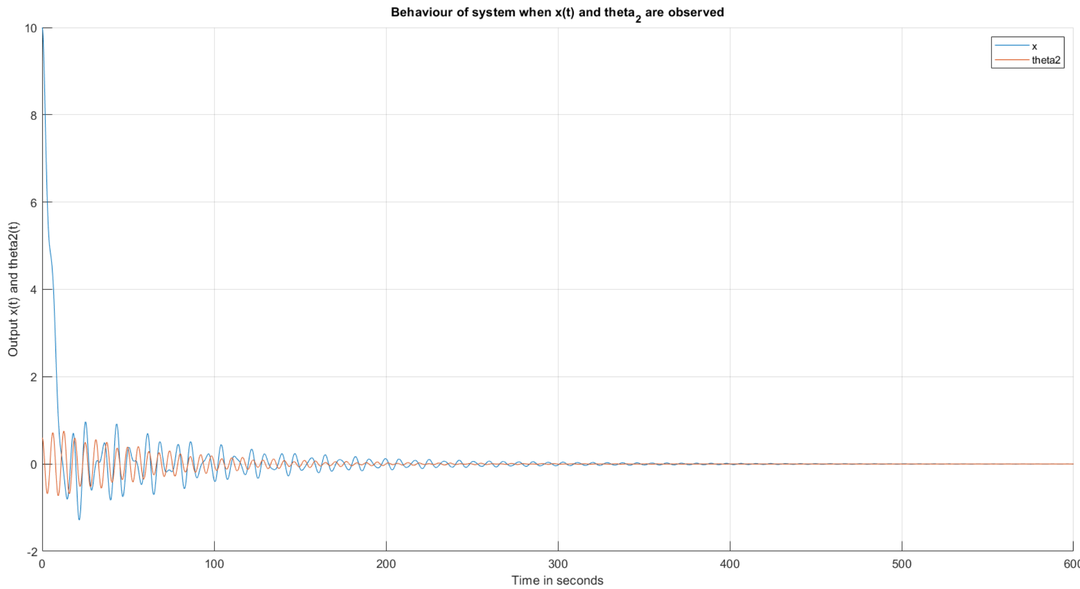

Figure 11: Response of non-linear system for second observer

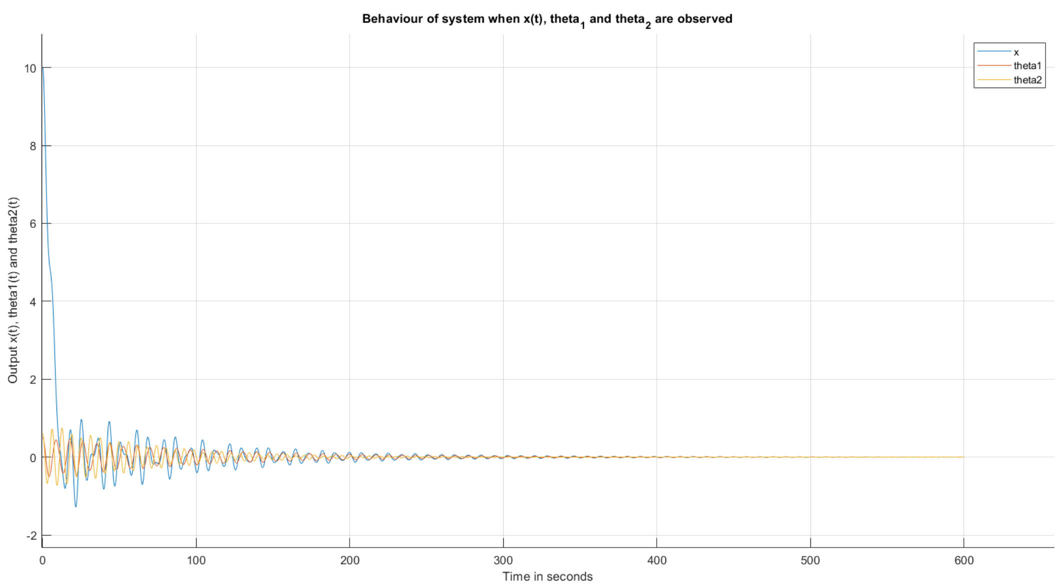

Figure 12: Response of non-linear system for fourth observer

8 LQG Controller

Output feedback: Linear Quadratic Gaussian Method (LQG)

$$X(t) = A\overrightarrow{X(t)} + B_k\overrightarrow{U_k(t)} + B_D\overrightarrow{U_D(t)}$$
$$\overrightarrow{Y(t)} = C\overrightarrow{X(t)} + \overrightarrow{V(t)}$$

where $\overrightarrow{U_D(t)}$ is process noise and $\overrightarrow{V(t)}$ is measurement noise.

8.1 Non linear system

Using the LQG method and applying the resulting output feedback controller to the original nonlinear system.

The code for non linear system is as follows:

```
    % Declaring variables to derive the state space representation of system
% M is the Mass of cart
% m1 is the mass attached to pendulum 1
% m2 is the mass attached to pendulum 2
% l1 is the length of pendulum 1
% l2 is the length of pendulum 2
% g is the accleration due to gravity

% Declaring the values of system variables
M_val = 1000;
m1_val = 100;
m2_val = 100;
l1_val = 20;
l2_val = 10;

A_val = [0 1 0 0 0 0;
         0 0 -(m1_val*9.81)/M_val 0 -(m2_val*9.81)/M_val 0;
         0 0 0 1 0 0;
         0 0 -((M_val+m1_val)*9.81/(M_val*l1_val)) 0 -(m2_val*9.81)/(M_val*l1_val) 0;
         0 0 0 0 0 1;
         0 0 -(m1_val*9.81)/(M_val*l2_val) 0 -((M_val+m2_val)*9.81/(M_val*l2_val)) 0];

% Declaring the B matrix of the system
B_val = [0; 1/M_val; 0; 1/(M_val*l1_val); 0; 1/(M_val*l2_val)];

% Defining the Q and R matrices for LQR controller
Q = [10  0   0   0   0   0;
     0  10   0   0   0   0;
     0   0 100   0   0   0;
     0   0   0   1   0   0;
     0   0   0   0 100   0;
     0   0   0   0   0   1];

R = 0.001;

% Finding the optimal closed-loop feedback gain using LQR
K = lqr(A_val,B_val,Q,R);

D = 0;

% Defining the process and measurement noise
Vd = 0.2*eye(6);
Vn = 1;
```

```
% The controllability matrix when output vector is x(t)
C1 = [1 0 0 0 0 0];

% The controllability matrix when output vector is (x(t), theta2(t))
C3 = [1 0 0 0 0 0;
      0 0 0 0 1 0];

% The controllability matrix when output vector is (x(t), theta1(t), theta2(t))
C4 = [1 0 0 0 0 0;
      0 0 1 0 0 0;
      0 0 0 0 1 0];

% For the observability matrix after putting parameter values
ob1 = obsv(A_val, C1);
ob3 = obsv(A_val, C3);
ob4 = obsv(A_val, C4);

% The poles of the Luenberger observer
L_poles = [-4; -4.5; -5; -5.5; -6; -6.5];
% Specifying the initial condition where estimated states
% are initialized to zero
x0 = [0;0;0.5;0;0.6;0;0;0;0;0;0;0];

% The observer gains for each observer, when the observer poles are placed
% at L_poles
L1 = place(A_val', C1', L_poles);
L3 = place(A_val', C3', L_poles);
L4 = place(A_val', C4', L_poles);

% The Luenberger observer obtained when x(t) is observed
% The initial condition
x0_obs = [10;0;0.5;0;0.6;0];

% Plotting x(t) for non-linear system when x(t) is observed
t_span = 0:0.01:600;
[ts,x_dots_L1] = ode45(@(t,x)non_lin_sys(t,x,-K*x,L1,C1),t_span,x0_obs);
figure
plot(ts,x_dots_L1(:,1))
grid
xlabel('Time in seconds')
ylabel('Output x(t)')
title('Behaviour of system when x(t) is observed')



% Checking the response of system when x(t) and
% theta2(t) are observed
figure
hold on
[ts,x_dots_L3] = ode45(@(t,x)non_lin_sys3(t,x,-K*x,L3',C3),t_span,x0_obs);
```

```
plot(ts,x_dots_L3(:,1))
plot(ts,x_dots_L3(:,5))
grid
xlabel('Time in seconds')
ylabel('Output x(t) and theta2(t)')
title('Behaviour of system when x(t) and theta_2 are observed')
legend({'x', 'theta2'},'Location','northeast')
hold off

% Checking the response of system when x(t), theta1(t) and
% theta2(t) are observed

% The response of non-linear system
% theta1 = 0.5 radians and theta2 = 0.6 radians
figure
hold on
[ts,x_dots_L4] = ode45(@(t,x)non_lin_sys4(t,x,-K*x,L4',C4),t_span,x0_obs);
plot(ts,x_dots_L4(:,1))
plot(ts,x_dots_L4(:,3))
plot(ts,x_dots_L4(:,5))
grid
xlabel('Time in seconds')
ylabel('Output x(t), theta1(t) and theta2(t)')
title('Behaviour of system when x(t), theta_1 and theta_2 are observed')
legend({'x', 'theta1', 'theta2'},'Location','northeast')
hold off


function x_dot = non_lin_sys(t,X,F,L,C)
x_dot = zeros(6,1);
% Declaring the values of system variables
M_val = 1000;
m1_val = 100;
m2_val = 100;
l1_val = 20;
l2_val = 10;
g_val = 9.81;
x = X(1);
x_d = X(2);
theta1 = X(3);
theta1_d = X(4);
theta2 = X(5);
theta2_d = X(6);
obs = L*(x-C*X);
x_dot(1) = x_d + obs(1);
x_dot(2) = (F-((m1_val*sin(theta1)*cos(theta1))+(m2_val*sin(theta2)*cos(theta2)))*g_val - (l1_val*m1_
x_dot(3) = theta1_d+obs(3);
x_dot(4) = ((cos(theta1)*x_dot(2)-g_val*sin(theta1))/l1_val) + obs(4);
x_dot(5) = theta2_d + obs(5);
x_dot(6) = (cos(theta2)*x_dot(2)-g_val*sin(theta2))/l2_val + obs(6);
end
```

```
function x_dot = non_lin_sys3(t,X,F,L,C)
x_dot = zeros(6,1);
% Declaring the values of system variables
M_val = 1000;
m1_val = 100;
m2_val = 100;
l1_val = 20;
l2_val = 10;
g_val = 9.81;

x = X(1);
dx = X(2);
theta1 = X(3);
theta1_d = X(4);
theta2 = X(5);
theta2_d = X(6);
y3 = [x; theta2];
sum = L*(y3-C*X);
x_dot(1) = dx + sum(1);
x_dot(2) = (F-((m1_val*sin(theta1)*cos(theta1))+(m2_val*sin(theta2)*cos(theta2)))*g_val - (l1_val*m1_
x_dot(3) = theta1_d+sum(3);
x_dot(4) = ((cos(theta1)*x_dot(2)-g_val*sin(theta1))/l1_val) + sum(4);
x_dot(5) = theta2_d + sum(5);
x_dot(6) = (cos(theta2)*x_dot(2)-g_val*sin(theta2))/l2_val + sum(6);
end


function x_dot = non_lin_sys4(t,X,F,L,C)
x_dot = zeros(6,1);
% Declaring the values of system variables
M_val = 1000;
m1_val = 100;
m2_val = 100;
l1_val = 20;
l2_val = 10;
g_val = 9.81;

x = X(1);
dx = X(2);
theta1 = X(3);
theta1_d = X(4);
theta2 = X(5);
theta2_d = X(6);
y4 = [x; theta1; theta2];
sum = L*(y4-C*X);
x_dot(1) = dx + sum(1);
x_dot(2) = (F-((m1_val*sin(theta1)*cos(theta1))+(m2_val*sin(theta2)*cos(theta2)))*g_val - (l1_val*m1_
x_dot(3) = theta1_d+sum(3);
x_dot(4) = ((cos(theta1)*x_dot(2)-g_val*sin(theta1))/l1_val) + sum(4);
x_dot(5) = theta2_d + sum(5);
x_dot(6) = (cos(theta2)*x_dot(2)-g_val*sin(theta2))/l2_val + sum(6);
```

```
end
```

8.2 Outputs of the linear system

Best feedback gain:

$$\begin{bmatrix}100.0000 & 501.2631 & 23.4087 & -331.8456 & 38.6244 & -155.4583\end{bmatrix}$$

Poles of feedback controller:

poles =
-0.0102 + 0.7277i
-0.0102 - 0.7277i
-0.0178 + 1.0423i
-0.0178 - 1.0423i
-0.2066 + 0.2023i
-0.2066 - 0.2023i
Best observer gain:

$$L = \begin{bmatrix}2.1429 & 2.1960 & -0.2843 & 0.5589 & -0.3758 & 0.3826\end{bmatrix}$$

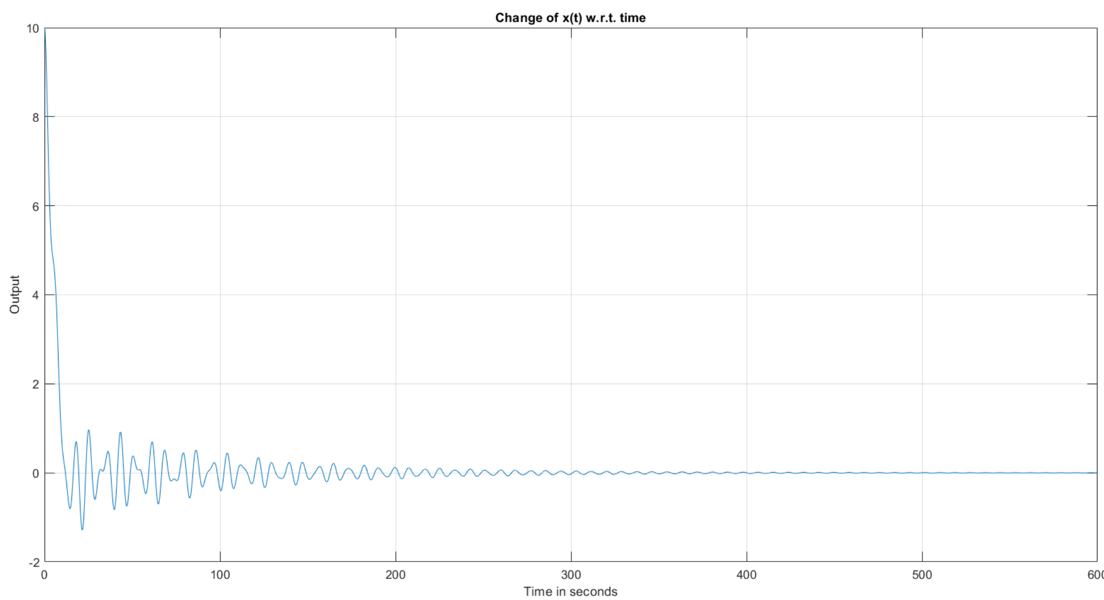

Figure 13: Response of x(t) when LQG controller is applied to non-linear system

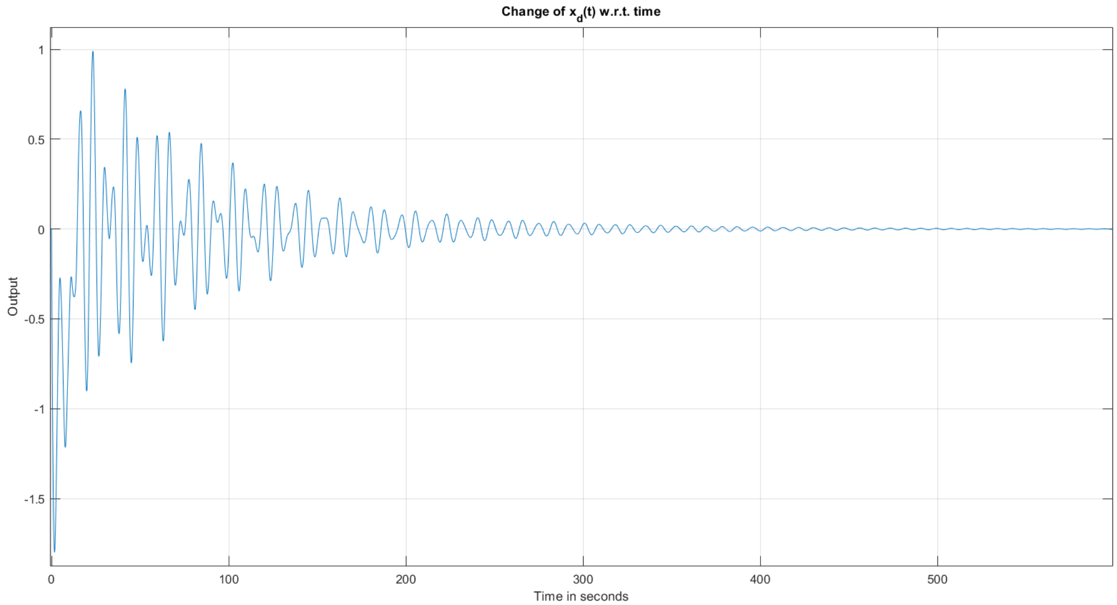

Figure 14: Response of x_d when LQG controller is applied to non-linear system

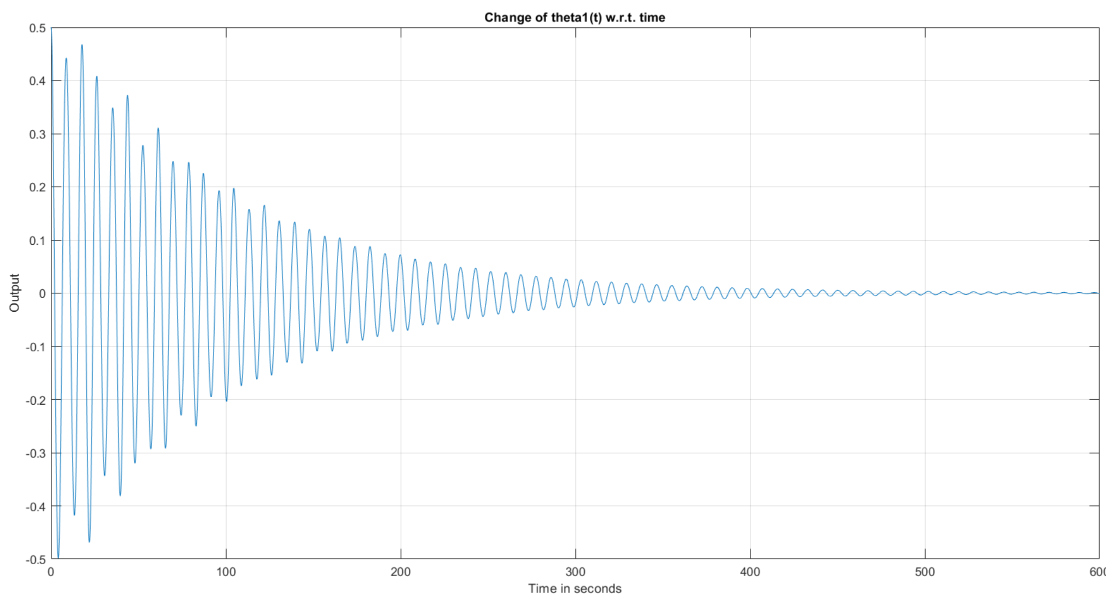

Figure 15: Response of $\theta_1(t)$ when LQG controller is applied to non-linear system

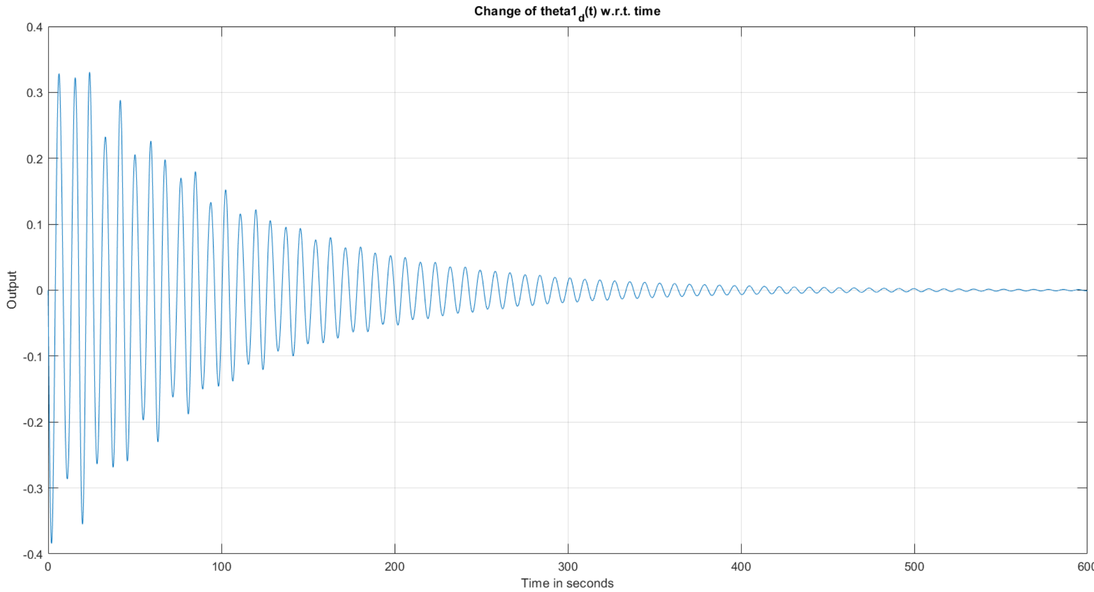

Figure 16: Response of $\theta_1 d$ when LQG controller is applied to non-linear system

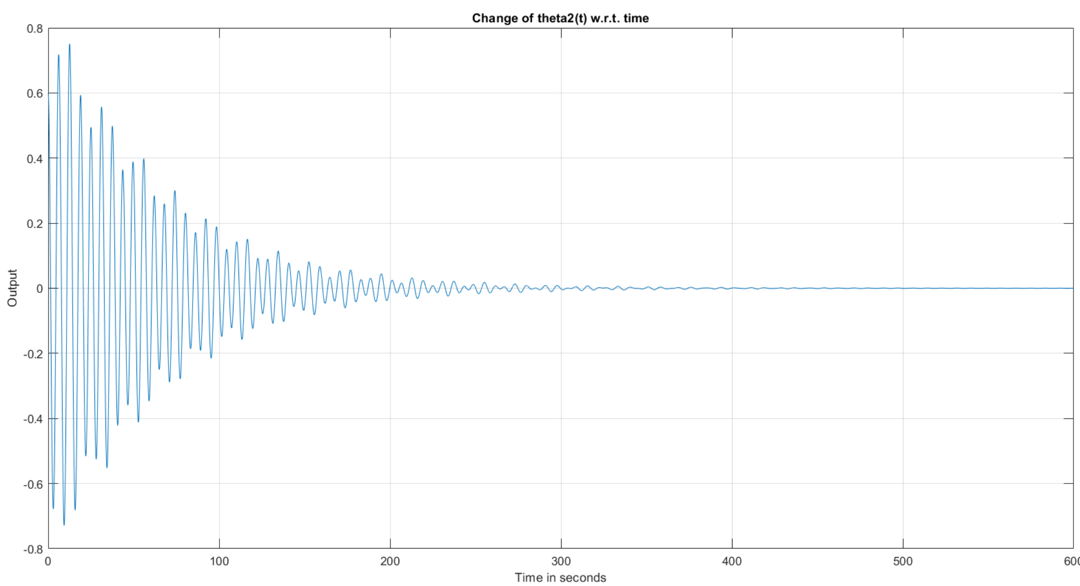

Figure 17: Response of θ_2 when LQG controller is applied to non-linear system

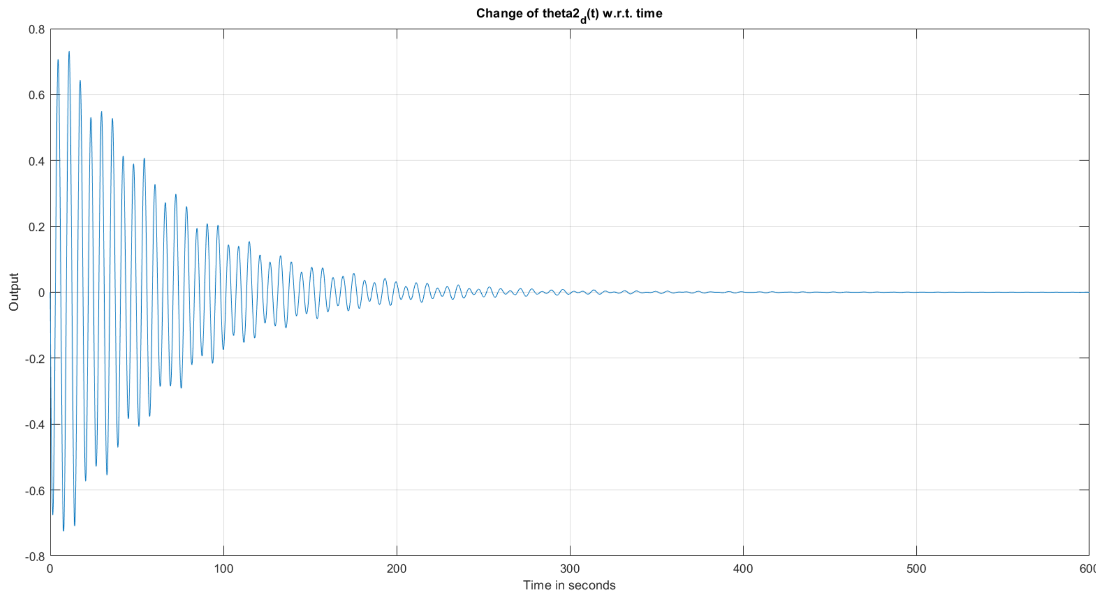

Figure 18: Response of $\theta_2 d$ when LQG controller is applied to non-linear system

Re configuring the controller to track a constant reference x:
We wish to minimize the following cost:

$$\int_0^\infty (\overrightarrow{X(t)} - \overrightarrow{X_d})^T Q(\overrightarrow{X(t)} - \overrightarrow{X_d}) - (\overrightarrow{U_k(t)} - \overrightarrow{U_\infty})^T R(\overrightarrow{U_k(t)} - \overrightarrow{U_\infty})\, dt$$

If there is U_∞ such that:

$$A\overrightarrow{X_d} + B_k\overrightarrow{U_\infty} = 0$$

To asymptotically track a constant reference x we have to minimize the above cost function.

If the constant force disturbances applied on the cart are Gaussian in nature our design will be able to reject these disturbances.