Controls and Systems

University of Maryland

Department of MAGE

Project 2 Report

Name:
Harshal Shirsath
(*UID: 119247419*)

Professor:
Waseem Malik

December 20, 2022

Contents

List of Figures

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.

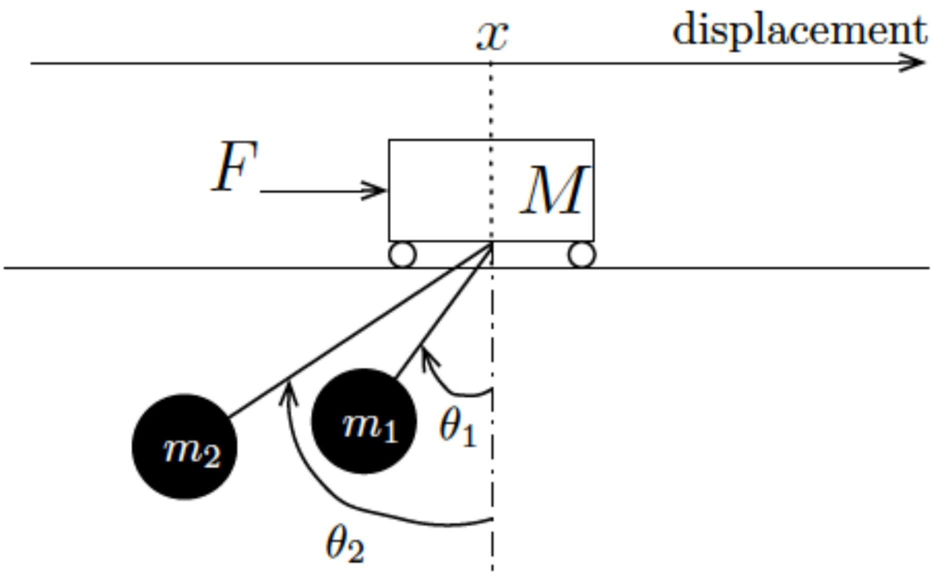

Figure 1: Crane with Two Pendulum

2 Introduction

The problem statement states two pendulums with different masses and dissimilar lengths suspended from a cart. The project directs us in designing a controller for the cart which is assembled with pendulums (aforementioned) set in motion.

This project is based on multiple concepts like :

1.Jacobian Transformation

2.Euler Lagrange Equations

3.Controllability

4.Observability

5. Linear Quadratic Regulator

6. Linear Quadratic Gaussian

7. Lyapunov Stability Equations

Additionally, other adjuvant topics were taught by Dr Waseem Malik in the coursework ENPM 667, during Fall 2022 at the University of Maryland, College Park, Maryland.

3 First Component A

Question A: Obtain the equations of motion for the system and the corresponding nonlinear state-space representation.

r(t)=

$$r_1(t) = (x - l_1 sin(\theta_1))i - l_1 cos(\theta_1)j$$

$$r_2(t) = (x - l_2 sin(\theta_2))i - l_2 cos(\theta_2)j$$

where x, θ_1, θ_2 are functions of time.

$$\dot{r_1}(t) = \frac{dr_1(t)}{dt} = \frac{d}{dt}(x - l_1 sin(\theta_1))i - l_1 cos(\theta_1)j$$
$$\dot{r_2}(t) = \frac{dr_2(t)}{dt} = \frac{d}{dt}(x - l_2 sin(\theta_2))i - l_2 cos(\theta_2)j$$

$$\dot{r_1}(t) = (\dot{x} - l_1\dot{\theta}_1 cos(\dot{\theta_1}))i + l_1\dot{\theta}_1 sin(\theta_1)j$$

$$\dot{r_2}(t) = (\dot{x} - l_2\dot{\theta}_2 cos(\dot{\theta_2}))i + l_2\dot{\theta}_2 sin(\theta_2)j$$

Kinematic Energy of the system is:

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

Whereas, the Potential Energy of the system is calculated by

$$P.E = mgh$$

$$h_1 = l_1 cos(\theta_1)$$

$$h_2 = l_2 cos(\theta_2)$$

Therefore the final Potential Energy of the two pendulums is:

$$P.E = -mgl_1 cos(\theta_1) - mgl_2 cos(\theta_2)$$

As per the Lagrange equation definition, it is the difference between kinematic energies and potential energies:

$$L = K.E - P.E$$

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

$$-(-mgl_1 cos(\theta_1) - mgl_2 cos(\theta_2))$$

$$L = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m_1(\dot{x} - l_1\dot{\theta_1}cos(\theta_1))^2 + \frac{1}{2}m_2(\dot{x} - l_2\dot{\theta_2}cos(\theta_2))^2 + \frac{1}{2}m_1(l_1\dot{\theta_1}sin(\theta_1))^2 + \frac{1}{2}m_2(l_2\dot{\theta_2}sin(\theta_2))^2$$

$$+mgl_1cos(\theta_1) + mgl_2cos(\theta_2)$$

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

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{x}} = M\ddot{x} + m_1(\ddot{x} - l_1\ddot{\theta_1}cos(\theta_1) + l_1\dot{\theta_1}^2sin(\theta_1)) + m_2(\ddot{x} - l_2\ddot{\theta_2}cos(\theta_2) + l_2\dot{\theta_2}^2sin(\theta_2)) \tag{1}$$

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

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{x}} - \frac{\partial L}{\partial x} = M\ddot{x} + m_1(\ddot{x} - l_1\ddot{\theta_1}cos(\theta_1) + l_1\dot{\theta_1}^2sin(\theta_1)) + m_2(\ddot{x} - l_2\ddot{\theta_2}cos(\theta_2) + l_2\dot{\theta_2}^2sin(\theta_2)) = F \tag{2}$$

Similarly, for θ_1

$$\frac{\partial L}{\partial \dot{\theta_1}} = m_1(\dot{x} - l_1\dot{\theta_1}cos(\theta_1))(-l_1cos(\theta_1)) + m_1(l_1\dot{\theta_1}sin(\theta_1))(l_1sin(\theta_1))$$

As we know that with the Lagrange Euler equation we usually differentiate with respect to time(t):

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta_1}} = -m_1\ddot{x}l_1cos(\theta_1) + m_1l_1^2\vec{\theta_1} + m_1\dot{x}\ddot{\theta_1}l_1sin\theta_1 \tag{3}$$

$$\frac{\partial L}{\partial \Theta_1} = m_1l_1^2\dot{\Theta}_1 - m_1\dot{x}l_1cos\Theta_1 \tag{4}$$

For Θ_2

$$\frac{\partial L}{\partial \Theta_2} = m_2(\dot{x} - l_2\dot{\theta_2}cos(\theta_2))(-l_2cos(\theta_2)) + m_2(l_2\dot{\theta_2}sin(\theta_2))(l_2sin(\theta_2)) \tag{5}$$

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta_2}} = -m_2\ddot{x}l_2cos(\theta_2) + m_2l_2^2\vec{\theta_2} + m_2\dot{x}\ddot{\theta_2}l_2sin\theta_2 \tag{6}$$

By inputs from (2),(3),(4) & (5) in the equation

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta_i}} - \frac{\partial L}{\partial \Theta_i} = 0$$

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta_1}} - \frac{\partial L}{\partial \Theta_1} = -m_1\ddot{x}l_1cos(\theta_1) + m_1l_1^2\vec{\theta_1} + m_1\dot{x}\ddot{\theta_1}l_1sin\theta_1 - m_1l_1^2\dot{\Theta}_1 + m_1\dot{x}l_1cos\Theta_1 \tag{7}$$

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta_2}} - \frac{\partial L}{\partial \Theta_2} = -m_2\ddot{x}l_2cos(\theta_2) + m_2l_2^2\vec{\theta_2} + m_2\dot{x}\ddot{\theta_1}l_2sin\theta_2 - m_2l_2^2\dot{\Theta}_2 + m_2\dot{x}l_2cos\Theta_2 \tag{8}$$

Equations (6) & (7)

$$l_1\ddot{\Theta}_1 = \ddot{x}cos\Theta_1 - gsin\Theta_1 \tag{9}$$

$$l_2\ddot{\Theta}_2 = \ddot{x}cos\Theta_2 - gsin\Theta_2 \tag{10}$$

Putting (8) & (9)

$$\begin{aligned}(M+m_1+m_2)\,\ddot{x} &= m_1\left(\ddot{x}\cos\theta_1 - g\sin\theta_1\right)\cos\theta_1 + m_2\left(\ddot{x}\cos\theta_2 - g\sin\theta_2\right)\cos\theta_2 - m_1 l_1\dot{\theta}_1^2\sin\theta_1 - m_2 l_2\dot{\theta}_2^2\sin\theta_2 + F \\ \ddot{x}\left(M+m_1\sin^2\theta_1+m_2\sin^2\theta_2\right) &= F - m_1 g\cos\theta_1\sin\theta_1 - m_2 g\cos\theta_2\sin\theta_2 - m_1 l_1\dot{\theta}_1^2\sin\theta_1 - m_2 l_2\dot{\theta}_2^2\sin\theta_2\end{aligned} \tag{11}$$

By substituting (11) in (2)

$$\begin{aligned} l_1\ddot{\theta}_1 &= \cos\theta_1\frac{\left(F - m_1 g\cos\theta_1\sin\theta_1 - m_2 g\cos\theta_2\sin\theta_2 - m_1 l_1\dot{\theta}_1^2\sin\theta_1 - m_2 l_2\dot{\theta}_2^2\sin\theta_2\right)}{\left(M+m_1\sin^2\theta_1+m_2\sin^2\theta_2\right)} - g\sin\theta_1 \\ l_2\ddot{\theta}_2 &= \cos\theta_2\frac{\left(F - m_1\,\mathrm{gcos}\,\theta_1\sin\theta_1 - m_2\,\mathrm{gos}\,\theta_2\sin\theta_2 - m_1 l_1\dot{\theta}_1^2\sin\theta_1 - m_2 l_2\dot{\theta}_2^2\sin\theta_2\right)}{\left(M+m_1\sin^2\theta_1+m_2\sin^2\theta_2\right)} - g\sin\theta_2 \end{aligned} \tag{12}$$

The non-Linear State Space Representation can be written as follows:

$$\begin{bmatrix} \ddot{x}(t) \\ \ddot{\theta}_1(t) \\ \ddot{\theta}_2(t) \end{bmatrix} = f\left(t, x(t), \dot{x}(t), \theta_1(t), \dot{\theta}_1(t), \theta_2(t), \dot{\theta}_2(t)\right) \tag{13}$$

$$\begin{bmatrix} \ddot{x}(t) \\ \ddot{\theta}_1(t) \\ \ddot{\theta}_2(t) \end{bmatrix} = \begin{bmatrix} \frac{\left(F-m_1 g\cos\theta_1\sin\theta_1-m_2 g\cos\theta_2\sin\theta_2-m_1 l_1\dot{\theta}_1^2\sin\theta_1-m_2 l_2\dot{\theta}_2^2\sin\theta_2\right)}{(M+m_1\sin^2\theta_1+m_2\sin^2\theta_2)} \\ \frac{\cos\theta_1\left(F-m_1 g\cos\theta_1\sin\theta_1-m_2 g\cos\theta_2\sin\theta_2-m_1 l_1\dot{\theta}_1^2\sin\theta_1-m_2 l_2\dot{\theta}_2^2\sin\theta_2\right)}{l_1(M+m_1\sin^2\theta_1+m_2\sin^2\theta_2)} - \frac{g\sin\theta_1}{l_1} \\ \frac{\cos\theta_2\left(F-m_1 g\cos\theta_1\sin\theta_1-m_2 g\cos^2\theta_2\sin\theta_2-m_1 l_1\dot{\theta}_1^2\sin\theta_1-m_2 l_2\dot{\theta}_2^2\sin\theta_2\right)}{l_2(M+m_1\sin^2\theta_1+m_2\sin^2\theta_2)} - \frac{g\sin\theta_2}{l_2} \end{bmatrix} \tag{14}$$

where in RHS θ_1, θ_2 are functions of time

4 Problem B:Linearized system around the equilibrium point specified

$$\begin{bmatrix} x(t) \\ \dot{x}(t) \\ \theta_1(t) \\ \dot{\theta}_1(t) \\ \theta_2(t) \\ \dot{\theta}_2(t) \end{bmatrix} \tag{15}$$

The general state-space equation for the system:

$$\mathbf{X} = A\mathbf{X} + B\mathbf{U} \tag{16}$$

Linearizing equation 17 at equilibrium point $x = 0, \theta_1 = 0$ and $\theta_2 = 0$, In this let us consider $\sin\theta \approx \theta$ and $\cos\theta \approx 1$

$$\begin{bmatrix} \ddot{x}(t) \\ \ddot{\theta}_1(t) \\ \ddot{\theta}_2(t) \end{bmatrix} = \begin{bmatrix} \frac{F}{M} - \frac{m_1 g\theta_1}{M} - \frac{m_2 g\theta_2}{M} \\ \frac{F}{Ml_1} - \frac{m_1 g\theta_1}{Ml_1} - \frac{g\theta_1}{l_1} - \frac{m_2 g\theta_2}{Ml_1} \\ \frac{F}{Ml_2} - \frac{m_2 g_2}{Ml_2} - \frac{g\theta_2}{l_2} - \frac{m_1 g\theta_1}{Ml_2} \end{bmatrix} \tag{17}$$

$$\begin{bmatrix} \ddot{x}(t) \\ \ddot{\theta}_1(t) \\ \ddot{\theta}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & -\frac{m_1 g}{M} & -\frac{m_2 g}{M} \\ 0 & -\frac{m_1 g}{Ml_1} - \frac{g}{l_1} & -\frac{m_2 g}{Ml_1} \\ 0 & -\frac{m_1 g}{Ml_2} & -\frac{m_2 g}{Ml_2} - \frac{g}{l_2} \end{bmatrix} \begin{bmatrix} x \\ \theta_1 \\ \theta_2 \end{bmatrix} + \begin{bmatrix} \frac{1}{M} \\ \frac{1}{Ml_1} \\ \frac{1}{Ml_2} \end{bmatrix} F$$

The Jacobian Matrix linearized around the given equilibrium points $\mathrm{x} = \theta_1 = \theta_2 = 0$ is given by:

$$\begin{bmatrix} \frac{\partial F_1}{\partial x} & \frac{\partial F_1}{\partial \dot{x}} & \frac{\partial F_1}{\partial \theta_1} & \frac{\partial F_1}{\partial \theta_1} & \frac{\partial F_1}{\partial \theta_2} & \frac{\partial F_1}{\partial \theta_2} \\ \frac{\partial F_2}{\partial x} & \frac{\partial F_2}{\partial \dot{x}} & \frac{\partial F_2}{\partial \theta_1} & \frac{\partial F_2}{\partial \theta_1} & \frac{\partial F_2}{\partial \theta_2} & \frac{\partial F_2}{\partial \dot{\theta}_2} \\ \frac{\partial F_3}{\partial x} & \frac{\partial F_3}{\partial \dot{x}} & \frac{\partial F_3}{\partial \theta_1} & \frac{\partial F_3}{\partial \theta_1} & \frac{\partial F_3}{\partial \theta_2} & \frac{\partial F_3}{\partial \theta_2} \\ \frac{\partial F_4}{\partial x} & \frac{\partial F_4}{\partial \dot{x}} & \frac{\partial F_4}{\partial \theta_1} & \frac{\partial F_4}{\partial \theta_1} & \frac{\partial F_4}{\partial \theta_2} & \frac{\partial F_4}{\partial \theta_2} \\ \frac{\partial F_5}{\partial x} & \frac{\partial F_5}{\partial \dot{x}} & \frac{\partial F_5}{\partial \theta_1} & \frac{\partial F_5}{\partial \theta_1} & \frac{\partial F_5}{\partial \theta_2} & \frac{\partial F_5}{\partial \theta_2} \\ \frac{\partial F_6}{\partial x} & \frac{\partial F_6}{\partial \dot{x}} & \frac{\partial F_6}{\partial \theta_1} & \frac{\partial F_6}{\partial \dot{\theta}_1} & \frac{\partial F_6}{\partial \theta_2} & \frac{\partial F_6}{\partial \dot{\theta}_2} \end{bmatrix} \tag{18}$$

The A matrix is shown below:

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{-m_1 g}{M} & 0 & \frac{-m_2 g}{M} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{(M+m_1)g}{Ml_1} & 0 & \frac{-m_2 g}{Ml_1} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{-m_1 g}{Ml_2} & 0 & \frac{(M+m_2)g}{Ml_2} & 0 \end{bmatrix} \tag{19}$$

The B matrix is shown below:

$$\begin{bmatrix} 0 \\ \dfrac{1}{M} \\ 0 \\ \dfrac{1}{Ml_1} \\ 0 \\ \dfrac{1}{Ml_2} \end{bmatrix} \tag{20}$$

Substitute the values of $\mathbf{A}$ and $\mathbf{B}$ in the state space equation to get the state equation 16 as:

$$\begin{bmatrix} \dot{x}(t) \\ \ddot{x}(t) \\ \dot{\theta}_1(t) \\ \ddot{\theta}_1(t) \\ \dot{\theta}_2(t) \\ \ddot{\theta}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{m_1 g}{M} & 0 & -\frac{m_{2g}}{M} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{m_1 g}{M_1} - \frac{g}{l_1} & 0 & -\frac{m_{2g}}{Ml_1} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{m_{1g}}{M_2} & 0 & -\frac{m_{2g}}{M_2} - \frac{g}{l_2} & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta_1 \\ \dot{\theta}_1 \\ \theta_2 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ \frac{1}{M_1} \\ 0 \\ \frac{1}{M_2} \end{bmatrix} F \tag{21}$$

```
%% Problem-C
clc
clear
syms m_1 m_2 m l_1 l_2 g
%% A matrix is given below
A=[0 1 0 0 0 0;
0 0 -(m_1*g)/m_2 0 -(m_2*g)/m 0;
0 0 0 1 0 0;
0 0 (-(m_1*g)/(m*l_1) -(g/l_1)) 0 -(m_2*g)/(m*l_1) 0;
0 0 0 0 0 1;
0 0 -(m_1*g)/(m*l_2) 0 (-(m_2*g)/(m*l_2) -(g/l_2)) 0 ];
%% B matrix is presented below
B=[0; 1/m; 0; 1/(m*l_1); 0; 1/(m*l_2)];
%% Controllability of Matrix is presented below
controllability_1=[B A*B A*A*B A*A*A*B A*A*A*A*B A*A*A*A*A*B];
% disp("Controllability Matrix (controllability_1)")
disp(controllability_1)
%% Simplifying the controllability matrix(controllability_1)
%% finding the determinant of the matrix controllability_1
disp("Determinant of Controllability Matrix (controllability_1)")
disp(simplify(det(controllability_1)))
%% Determining the rank of the controllability matrix
disp("Rank of the Controllability Matrix (controllability_1)")
rank(controllability_1)
controllability_2 = subs(controllability_1,l_1,l_2);
disp("Controllability Matrix controllability_2 after substitution")
disp(controllability_2)
%% Rank of the controllability matrix after substituting the values of  l_1 = l_2
disp("Rank of the Controllability Matrix (controllability_2)")
rank(controllability_2)
```

12/20/22 6:04 AM F:\Controls in robotics sy...\Harshal.m 1 of 1

```
%% Problem-C
clc
clear
syms m_1 m_2 m l_1 l_2 g
%% A matrix is given below
A=[0 1 0 0 0 0;
0 0 -(m_1*g)/m_2 0 -(m_2*g)/m 0;
0 0 0 1 0 0;
0 0 (-(m_1*g)/(m*l_1) -(g/l_1)) 0 -(m_2*g)/(m*l_1) 0;
0 0 0 0 0 1;
0 0 -(m_1*g)/(m*l_2) 0 (-(m_2*g)/(m*l_2) -(g/l_2)) 0 ];
%% B matrix is presented below
B=[0; 1/m; 0; 1/(m*l_1); 0; 1/(m*l_2)];
%% Controllability of Matrix is presented below
controllability_1=[B A*B A*A*B A*A*A*B A*A*A*A*B A*A*A*A*A*B];
% disp("Controllability Matrix (controllability_1)")
disp(controllability_1)
%% Simplifying the controllability matrix(controllability_1)
%% finding the determinant of the matrix controllability_1
disp("Determinant of Controllability Matrix (controllability_1)" )
disp(simplify(det(controllability_1)))
%% Determining the rank of the controllability matrix
disp("Rank of the Controllability Matrix (controllability_1)" )
rank(controllability_1)
controllability_2 = subs(controllability_1,l_1,l_2);
disp("Controllability Matrix controllability_2 after substitution" )
disp(controllability_2)
%% Rank of the controllability matrix after substituting the values of  l_1 = l_2
disp("Rank of the Controllability Matrix (controllability_2)" )
rank(controllability_2)
```

MATLAB Command Window Page 1

```
[        0,       1/m,                                                    0, ↙
- (g*m_2)/(l_2*m^2) - (g*m_1)/(l_1*m*m_2), ↙
0,                          ((g^2*m_1)/(l_1*m) + (g*m_2*(g/l_2 + (g*m_2)/(l_2*m)))/m)/ ↙
(l_2*m) + ((g*m_1*(g/l_1 + (g*m_1)/(l_1*m)))/m_2 + (g^2*m_1*m_2)/(l_2*m^2))/(l_1*m)]
[      1/m,         0,                  - (g*m_2)/(l_2*m^2) - (g*m_1)/(l_1*m*m_2), ↙
0,                          ((g^2*m_1)/(l_1*m) + (g*m_2*(g/l_2 + (g*m_2)/(l_2*m)))/m)/ ↙
(l_2*m) + ((g*m_1*(g/l_1 + (g*m_1)/(l_1*m)))/m_2 + (g^2*m_1*m_2)/(l_2*m^2))/(l_1*m), ↙
0]
[        0, 1/(l_1*m),                                                    0, - ↙
(g/l_1 + (g*m_1)/(l_1*m))/(l_1*m) - (g*m_2)/(l_1*l_2*m^2), ↙
0, ((g*m_2*(g/l_1 + (g*m_1)/(l_1*m)))/(l_1*m) + (g*m_2*(g/l_2 + (g*m_2)/(l_2*m)))/ ↙
(l_1*m))/(l_2*m) + ((g/l_1 + (g*m_1)/(l_1*m))^2 + (g^2*m_1*m_2)/(l_1*l_2*m^2))/(l_1*m)]
[1/(l_1*m),         0, - (g/l_1 + (g*m_1)/(l_1*m))/(l_1*m) - (g*m_2)/(l_1*l_2*m^2), ↙
0, ((g*m_2*(g/l_1 + (g*m_1)/(l_1*m)))/(l_1*m) + (g*m_2*(g/l_2 + (g*m_2)/(l_2*m)))/ ↙
(l_1*m))/(l_2*m) + ((g/l_1 + (g*m_1)/(l_1*m))^2 + (g^2*m_1*m_2)/(l_1*l_2*m^2))/(l_1*m), ↙
0]
[        0, 1/(l_2*m),                                                    0, - ↙
(g/l_2 + (g*m_2)/(l_2*m))/(l_2*m) - (g*m_1)/(l_1*l_2*m^2), ↙
0, ((g*m_1*(g/l_1 + (g*m_1)/(l_1*m)))/(l_2*m) + (g*m_1*(g/l_2 + (g*m_2)/(l_2*m)))/ ↙
(l_2*m))/(l_1*m) + ((g/l_2 + (g*m_2)/(l_2*m))^2 + (g^2*m_1*m_2)/(l_1*l_2*m^2))/(l_2*m)]
[1/(l_2*m),         0, - (g/l_2 + (g*m_2)/(l_2*m))/(l_2*m) - (g*m_1)/(l_1*l_2*m^2), ↙
0, ((g*m_1*(g/l_1 + (g*m_1)/(l_1*m)))/(l_2*m) + (g*m_1*(g/l_2 + (g*m_2)/(l_2*m)))/ ↙
(l_2*m))/(l_1*m) + ((g/l_2 + (g*m_2)/(l_2*m))^2 + (g^2*m_1*m_2)/(l_1*l_2*m^2))/(l_2*m), ↙
0]

Determinant of Controllability Matrix (controllability_1)
-(g^6*(l_1 - l_2)^2*(m*m_2 - m*m_1 + m_1*m_2)^2)/(l_1^6*l_2^6*m^8*m_2^2)

Rank of the Controllability Matrix (controllability_1)

ans =

     6

Controllability Matrix controllability_2 after substitution
[        0,       1/m,                                                  0, ↙
- (g*m_2)/(l_2*m^2) - (g*m_1)/(l_2*m*m_2), ↙
0,                        ((g^2*m_1)/(l_2*m) + (g*m_2*(g/l_2 + (g*m_2)/(l_2*m)))/m)/ ↙
(l_2*m) + ((g*m_1*(g/l_2 + (g*m_1)/(l_2*m)))/m_2 + (g^2*m_1*m_2)/(l_2*m^2))/(l_2*m)]
[      1/m,         0,                - (g*m_2)/(l_2*m^2) - (g*m_1)/(l_2*m*m_2), ↙
0,                        ((g^2*m_1)/(l_2*m) + (g*m_2*(g/l_2 + (g*m_2)/(l_2*m)))/m)/ ↙
(l_2*m) + ((g*m_1*(g/l_2 + (g*m_1)/(l_2*m)))/m_2 + (g^2*m_1*m_2)/(l_2*m^2))/(l_2*m), ↙
0]
[        0, 1/(l_2*m),                                                  0, - ↙
(g/l_2 + (g*m_1)/(l_2*m))/(l_2*m) - (g*m_2)/(l_2^2*m^2), ↙
0, ((g/l_2 + (g*m_1)/(l_2*m))^2 + (g^2*m_1*m_2)/(l_2^2*m^2))/(l_2*m) + ((g*m_2*(g/l_2 + ↙
(g*m_1)/(l_2*m)))/(l_2*m) + (g*m_2*(g/l_2 + (g*m_2)/(l_2*m)))/(l_2*m))/(l_2*m)]
[1/(l_2*m),         0, - (g/l_2 + (g*m_1)/(l_2*m))/(l_2*m) - (g*m_2)/(l_2^2*m^2), ↙
0, ((g/l_2 + (g*m_1)/(l_2*m))^2 + (g^2*m_1*m_2)/(l_2^2*m^2))/(l_2*m) + ((g*m_2*(g/l_2 + ↙
(g*m_1)/(l_2*m)))/(l_2*m) + (g*m_2*(g/l_2 + (g*m_2)/(l_2*m)))/(l_2*m))/(l_2*m), ↙
```

```
0]
[        0, 1/(l_2*m),                                                      0, - ↙
(g/l_2 + (g*m_2)/(l_2*m))/(l_2*m) - (g*m_1)/(l_2^2*m^2), ↙
0, ((g/l_2 + (g*m_2)/(l_2*m))^2 + (g^2*m_1*m_2)/(l_2^2*m^2))/(l_2*m) + ((g*m_1*(g/l_2 + ↙
(g*m_1)/(l_2*m)))/(l_2*m) + (g*m_1*(g/l_2 + (g*m_2)/(l_2*m)))/(l_2*m))/(l_2*m)]
[1/(l_2*m),         0, - (g/l_2 + (g*m_2)/(l_2*m))/(l_2*m) - (g*m_1)/(l_2^2*m^2), ↙
0, ((g/l_2 + (g*m_2)/(l_2*m))^2 + (g^2*m_1*m_2)/(l_2^2*m^2))/(l_2*m) + ((g*m_1*(g/l_2 + ↙
(g*m_1)/(l_2*m)))/(l_2*m) + (g*m_1*(g/l_2 + (g*m_2)/(l_2*m)))/(l_2*m))/(l_2*m), ↙
0]

Rank of the Controllability Matrix (controllability_2)

ans =

     4

>>
```

```
%% Problem-D
clc
clear
%% LQR linearized system
%% Parameters are
m=1000;
m_1=100;
m_2=100;
l_1=20;
l_2=10;
g=9.81;
%% defining the state matrices
A=[0 1 0 0 0 0;
   0 0 -(m_1*g)/m 0 -(m_2*g)/m 0;
   0 0 0 1 0 0;
   0 0 (-(m_1*g)/(m*l_1) -(g/l_1)) 0 -(m_2*g)/(m*l_1) 0;
   0 0 0 0 0 1;
   0 0 -(m_1*g)/(m*l_2) 0 (-(m_2*g)/(m*l_2) -(g/l_2)) 0 ];

B=[0; 1/m; 0; 1/(m*l_1); 0; 1/(m*l_2)];
C=[1 0 0 0 0 0; 0 0 1 0 0 0;0 0 0 0 1 0];
D=0;

%% defining LQR inputs
%LQR Inputs are
Quad = [1000 0 0 0 0 0;
     0 0 0 0 0 0;
     0 0 1000000 0 0 0;
     0 0 0 0 0 0;
     0 0 0 0 1000000 0;
     0 0 0 0 0 0]
R = 0.1 ;

%Gain Calculation from LQR are
K = lqr(A,B,Quad,R)

%% LQR gain K is as follows
Aq = [(A-(B*K))];
Bq = [B];
Cq = [C];
Dq = [D];
```

```
%states
states = {'x' 'x_dot' 'q_1' 'q_1_dot' 'q_2' 'q_2_dot'};
% LQR leads to output which are:
inputs = {'f'};
% LQR leads to output which are:
outputs = {'x' ;'q_1';'q_2'};
sys_cal = ss(Aq,Bq,Cq,Dq,'statename',states,'inputname',inputs,'outputname',outputs);
%states space model is created
t = 0:0.1:200;
f = 50*ones(size(t));
%response simulations are:
[y,t,x]=lsim(sys_cal,f,t);

%% plots for the responses are:
[AX,H1,H2] = plotyy(t,y(:,1),[t,t],[y(:,2),y(:,3)],'plot');
set(get(AX(1),'Ylabel'),'String','position of cart (m)');
set(get(AX(2),'Ylabel'),'String','angle of pendulum (radians)');
title('Step Response(LQR Control)');
```

```
MATLAB Command Window                                          Page 1

Quad =

        1000           0           0           0           0           0
           0           0           0           0           0           0
           0           0     1000000           0           0           0
           0           0           0           0           0           0
           0           0           0           0     1000000           0
           0           0           0           0           0           0


K =

   1.0e+03 *

    0.1000    0.5535    1.1624    3.6032    2.0948    1.2374

>>
```

Figure 2: Controllability matrix

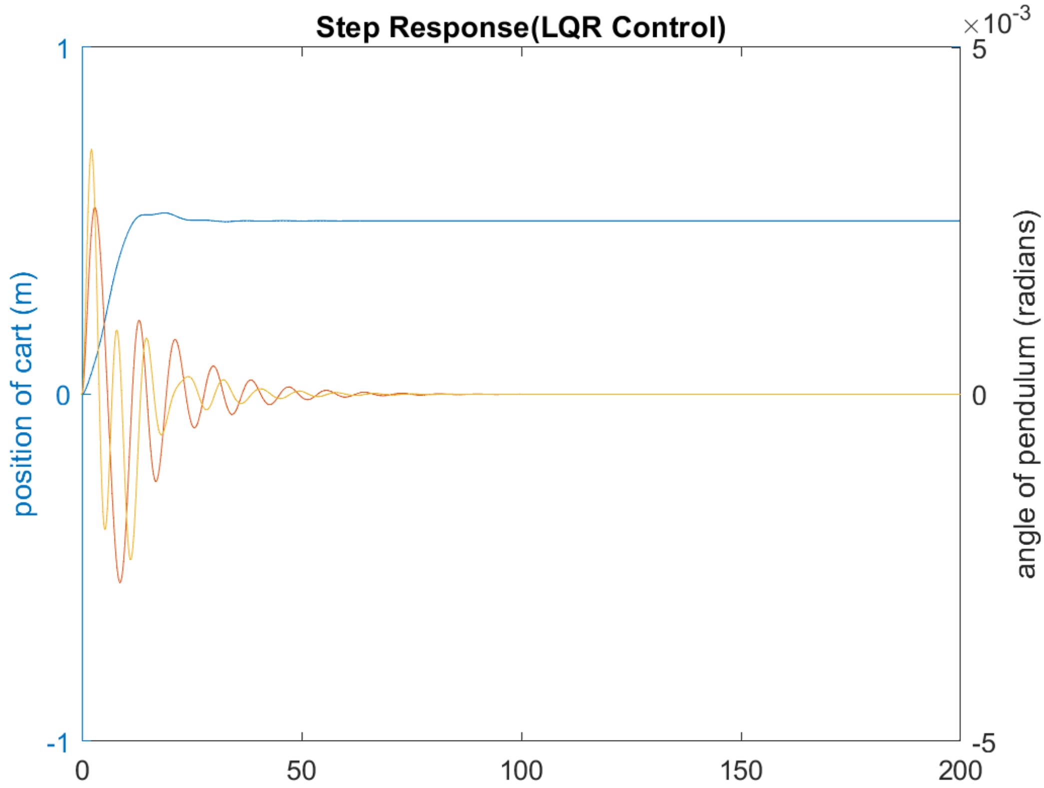

Figure 3: Step Response

```
%% PROBLEM-E
clc
clear

syms m_1 m_2 m l_1 l_2 g
% %% A matrix will be
A=[0 1 0 0 0 0; 0 0 -(m_1*g)/m 0 -(m_2*g)/m 0;
   0 0 0 1 0 0;
   0 0 (-(m_1*g)/(m*l_1) -(g/l_1)) 0 -(m_2*g)/(m*l_1) 0;
   0 0 0 0 0 1;
   0 0 -(m_1*g)/(m*l_2) 0 (-(m_2*g)/(m*l_2) -(g/l_2)) 0 ];

%% While using Matrix C we are calculating different output
L_1=[1 0 0 0 0 0];                  % C matrix for X as output
L_2=[1 0 0 0 0 0;
    0 0 1 0 0 0;
    0 0 0 0 1 0];                  % C matrix for X,theta_1,theta_2 as output
L_3=[0 0 1 0 0 0; 0 0 0 0 1 0];     % C matrix for theta_1,theta_2 as output
L_4=[1 0 0 0 0 0; 0 0 0 0 1 0];     % C matrix for X,theta_2 as output
%% Checking Observability of output - X
O_1=[L_1 ;L_1*A; L_1*A^2; L_1*A^3; L_1*A^4; L_1*A^5];
disp('Rank of O_1 = ')
disp(rank(O_1))
if rank(O_1)==6
    disp('System Observable, x(t) requested')
else
    disp('System NOT Observable, x(t) requested')
end
disp('      ')
disp('      ')
%% Checking Observability of output - X,theta_1,theta_2
O_2=[L_2 ;L_2*A; L_2*A^2; L_2*A^3; L_2*A^4; L_2*A^5];
disp('Rank of O_2 = ')
disp(rank(O_2))
if rank(O_2)==6
    disp('System Observable, when x(t), theta_1, theta_2 are requested')
else
    disp('System NOT Observable, when x(t), theta_1, theta_2 are requested')
end
disp('      ')
disp('      ')
```

```
%% Checking Observability of output - theta_1,theta_2
O_3=[L_3 ;L_3*A; L_3*A^2; L_3*A^3; L_3*A^4; L_3*A^5];
disp('Rank of O_3= ')
disp(rank(O_3))
if rank(O_3)==6
    disp('System Observable, when theta_1, theta_2 are requested')
else
    disp('System NOT Observable, when theta_1, theta_2 are requested')
end
disp('       ')
disp('       ')
%% Checking Observability of output - X,theta_2
O_4=[L_4 ;L_4*A; L_4*A^2; L_4*A^3; L_4*A^4; L_4*A^5];
disp('Rank of O_4 = ')
disp(rank(O_4))
if rank(O_4)==6
    disp('System Observable, when x(t), theta_2 are requested')
else
    disp('System NOT Observable, when x(t), theta_2 are requested')
end
```

MATLAB Command Window Page 1

```
Rank of O_1 =
     6

System Observable, x(t) requested


Rank of O_2 =
     6

System Observable, when x(t), theta_1, theta_2 are requested


Rank of O_3=
     4

System NOT Observable, when theta_1, theta_2 are requested


Rank of O_4 =
     6

System Observable, when x(t), theta_2 are requested
>>
```

```
%% Problem-F_x
clc
clear

%% Parameters are
m_1=100;
m_2=100;
m=1000;
g=9.8;
l_1=20;
l_2=10;

%% State Matrices A,B,C & D
A=[0 1 0 0 0 0;
   0 0 -(m_1*g)/m 0 -(m_2*g)/m 0;
   0 0 0 1 0 0;
   0 0 (-(m_1*g)/(m*l_1) -(g/l_1)) 0 -(m_2*g)/(m*l_1) 0;
   0 0 0 0 0 1;
   0 0 -(m_1*g)/(m*l_2) 0 (-(m_2*g)/(m*l_2) -(g/l_2)) 0 ];
B=[0; 1/m; 0; 1/(m*l_1); 0; 1/(m*l_2)];
C=[1 0 0 0 0 0];
D=0;

%% From LQR we will find the gain matrix
Quad = [1000 0 0 0 0 0;
     0 0 0 0 0 0;
     0 0 1000000 0 0 0;
     0 0 0 0 0 0;
     0 0 0 0 1000000 0;
     0 0 0 0 0 0]

%LQR Input
Reg = 0.1 ;

%Gained Calculation from LQR are presented below
K = lqr(A,B,Quad,Reg)
Aq = [(A-(B*K))];
Bq = [B];
Cq = [C];
Dq = [D];
```

```
%eigenvalues dynamics are changed using gain
l=eig(Aq)

%states
states = {'x' 'x_dot' 'q_1' 'q_1_dot' 'q_2' 'q_2_dot'};

% input after lqr
inputs = {'f'};

% output after lqr
outputs = {'x'};

%creates statesspace model
sys_cal_1 = ss(Aq,Bq,Cq,Dq,'statename',states,'inputname',inputs,'outputname',outputs);
t = 0:0.1:200;
f = 50*ones(size(t));

%simulates response
[y,t,x]=lsim(sys_cal_1,f,t);
plot(t,y(:,1));

%% Finding 'best' observer matrix
%finding the best poles
P = 10*[l.'];

%Values of observer are found using  pole placment
L = place(A',C',P)'
A_L = [(A-(L*C))];
B_L = [B];
C_L = [C];
D_L = [D];

%states
states = {'x' 'x_dot' 'p1' 'p1_dot' 'p2' 'p2_dot'};

% input for observer
inputs = {'f'};

% output after observer
outputs = {'x'};
```

```
%creates statesspace model
sys_cal_2 = ss(A_L,B_L,C_L,D_L,'statename',states,'inputname',inputs,'outputname',outputs)
T = 0:0.1:200;
F_1 = 50*ones(size(T));

%simulates response
[y,T,x]=lsim(sys_cal_2,F_1,T);

%plotting the response
plot(t,y(:,1));
title('Observer x');
```

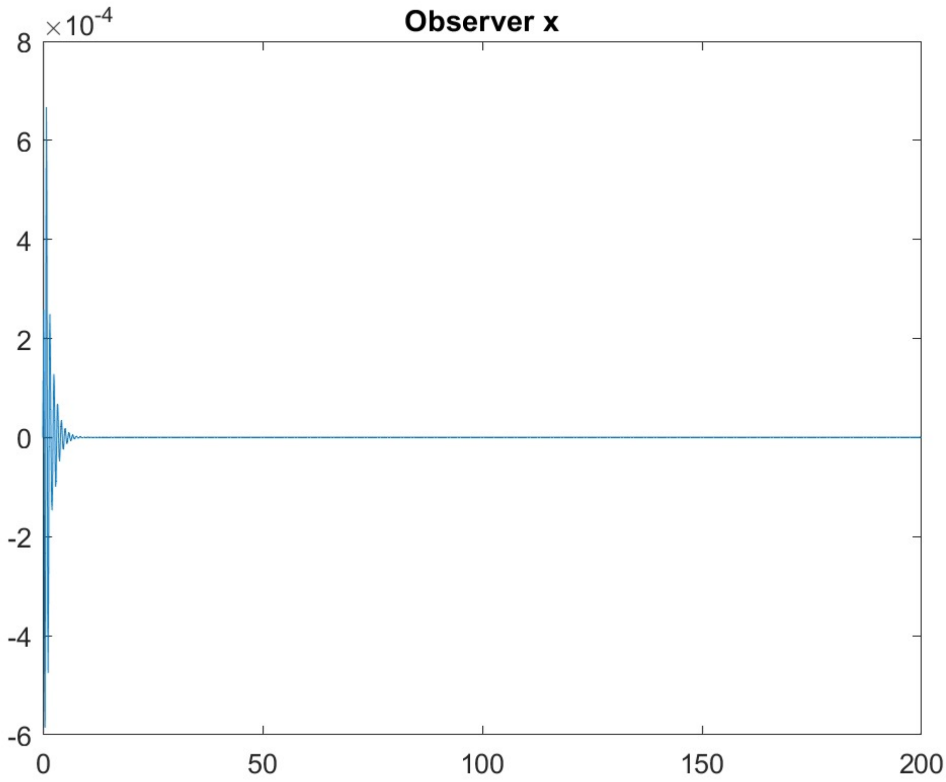

Figure 4: Observer for x

```
%% Problem1F_T_12
clc
clear

%% Parameters are
m_1=100;
m_2=100;
l_1=20;
l_2=10;
m=1000;
g=9.8;
A=[0 1 0 0 0 0; 0 0 -(m_1*g)/m 0 -(m_2*g)/m 0;
   0 0 0 1 0 0;
   0 0 (-(m_1*g)/(m*l_1) -(g/l_1)) 0 -(m_2*g)/(m*l_1) 0;
   0 0 0 0 0 1;
   0 0 -(m_1*g)/(m*l_2) 0 (-(m_2*g)/(m*l_2) -(g/l_2)) 0 ];
B=[0; 1/m; 0; 1/(m*l_1); 0; 1/(m*l_2)];
C=[1 0 0 0 0 0; 0 0 1 0 0 0;0 0 0 0 1 0];
D=0;
%% Best Observer for X,theta_1,theta_2 as output
%% Finding the gain matrix from LQR
Quad = [1000 0 0 0 0 0;
     0 0 0 0 0 0;
     0 0 1000000 0 0 0;
     0 0 0 0 0 0;
     0 0 0 0 1000000 0;
     0 0 0 0 0 0]

%LQR Input
Reg = 0.1 ;

%Gain Calculation from LQR
K = lqr(A,B,Quad,Reg)
Aq = [(A-(B*K))];
Bq = [B];
Cq = [C];
Dq = [D];

%eigenvalues dynamics are changed using gain
l=eig(Aq)
```

```
%states
states = {'x' 'x_dot' 'q_1' 'q_1_dot' 'q_2' 'q_2_dot'};

% input for lqr
inputs = {'f'};

% output after lqr
outputs = {'x';'q_1';'q_2'};

%creates statesspace model
sys_cal_1 = ss(Aq,Bq,Cq,Dq,'statename',states,'inputname',inputs,'outputname',outputs);
T = 0:0.1:200;
F_1 = 50*ones(size(T));

%simulates response
[y,T,x]=lsim(sys_cal_1,F_1,T);
[A_X,H_1,H_2] = plotyy(T,y(:,1),[T,T],[y(:,2),y(:,3)],'plot');

%% Finding 'best' observer matrix
 %finding the best poles
P = 10*[l.']

%Values of observer are found using  pole placment
L = place(A',C',P)'
A_L = [(A-(L*C))];
B_L = [B];
C_L= [C];
D_L = [D];

%states
states = {'x' 'x_dot' 'q_1' 'q_1_dot' 'q_2' 'q_2_dot'};

% input for observer
inputs = {'f'};

% output after observer
outputs = {'x';'q_1';'q_2'};

%creates statesspace model
sys_cal_2 = ss(A_L,B_L,C_L,D_L,'statename',states,'inputname',inputs,'outputname',outputs)
T = 0:0.1:200;
```

```
F_1 = 50*ones(size(T));

%simulating response
[y,T,x]=lsim(sys_cal_2,F_1,T);

% Plotting the response
[A_X,H_1,H_2] = plotyy(T,y(:,1),[T,T],[y(:,2),y(:,3)],'plot');
title('Observer for x, theta_1 and theta_2');
```

```
MATLAB Command Window                                          Page 1


Quad =

        1000           0           0           0           0           0
           0           0           0           0           0           0
           0           0     1000000           0           0           0
           0           0           0           0           0           0
           0           0           0           0     1000000           0
           0           0           0           0           0           0


K =

   1.0e+03 *

    0.1000    0.5536    1.1630    3.6050    2.0958    1.2371


l =

  -0.1584 + 1.0534i
  -0.1584 - 1.0534i
  -0.1945 + 0.2008i
  -0.1945 - 0.2008i
  -0.0759 + 0.7319i
  -0.0759 - 0.7319i


P =

  Columns 1 through 4

  -1.5842 +10.5338i  -1.5842 -10.5338i  -1.9447 + 2.0083i  -1.9447 - 2.0083i

  Columns 5 through 6

  -0.7590 + 7.3194i  -0.7590 - 7.3194i


L =

    2.4464    3.5773    2.9584
   54.7801   -1.6741   14.3287
   -2.7614    3.1898    7.3290
  -12.4463   22.4325   13.6191
   -4.6894   -6.1291    2.9396
    4.3348  -16.0632   26.5996

>>
```

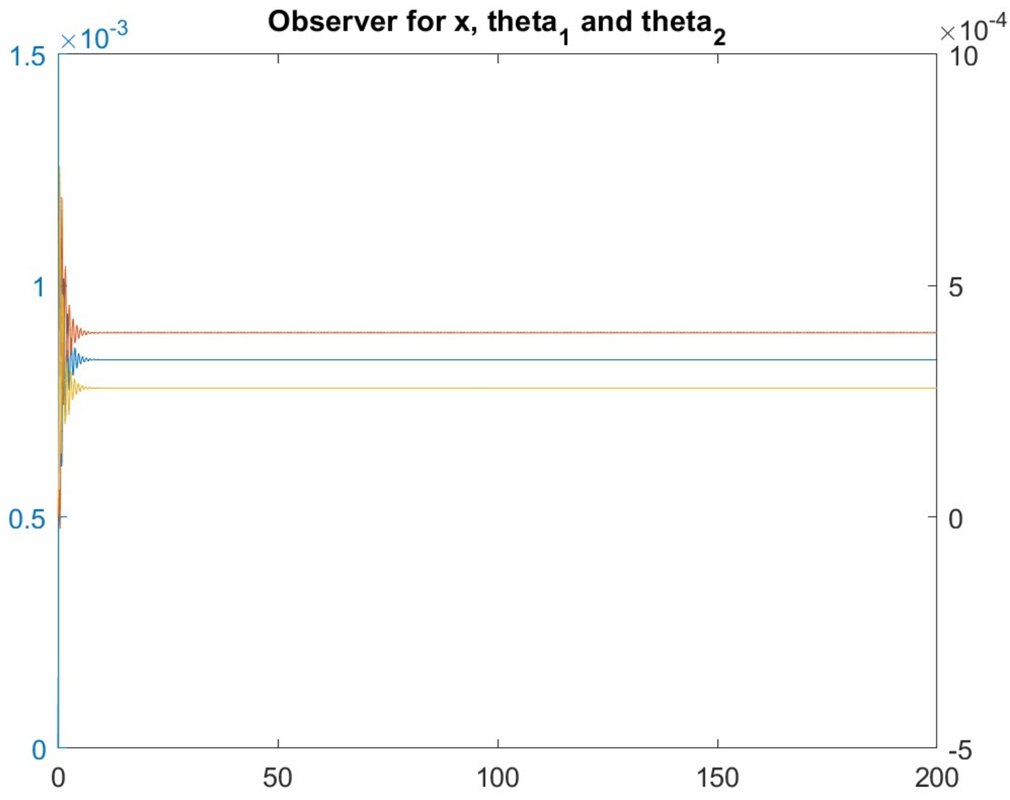

Figure 5: Observer for x, θ_1, θ_2

```
%% Problem1F_T_22
clc
clear
%% Best Observer for X,theta2 as output
%% Given Paramaters
m_1=100;
m_2=100;
m=1000;
l_1=20;
l_2=10;
g=9.8;

%% State Matrices
A=[0 1 0 0 0 0;
   0 0 -(m_1*g)/m 0 -(m_2*g)/m 0;
   0 0 0 1 0 0;
   0 0 (-(m_1*g)/(m*l_1) -(g/l_1)) 0 -(m_2*g)/(m*l_1) 0;
   0 0 0 0 0 1;
   0 0 -(m_1*g)/(m*l_2) 0 (-(m_2*g)/(m*l_2) -(g/l_2)) 0 ];
```

```
B=[0; 1/m; 0; 1/(m*l_1); 0; 1/(m*l_2)];
C=[1 0 0 0 0 0; 0 0 0 0 1 0];
D=0;

%% Finding the gain matrix from LQR
Quad = [1000 0 0 0 0 0;
     0 0 0 0 0 0;
     0 0 1000000 0 0 0;
     0 0 0 0 0 0;
     0 0 0 0 1000000 0;
     0 0 0 0 0 0]

%LQR parameters
Reg = 0.1 ;

%LQR Input
K = lqr(A,B,Quad,Reg)

%Gain Calculation from LQR
Aq = [(A-(B*K))];
Bq = [B];
Cq = [C];
Dq = [D];

%eigenvalues dynamics are changed using gain
l=eig(Aq)

%states
states = {'x' 'x_dot' 'q_1' 'q_1_dot' 'q_2' 'q_2_dot'};

% input for lqr
inputs = {'f'};

% output after lqr
outputs = {'x' ;'q_2'};

%creates statesspace model
sys_cl = ss(Aq,Bq,Cq,Dq,'statename',states,'inputname',inputs,'outputname',outputs);
t = 0:0.1:200;
f = 50*ones(size(t));
[y,t,~]=lsim(sys_cl,f,t);
```

```
%simulates response
[AX,H1,H2] = plotyy(t,y(:,1),[t,t],[y(:,2)],'plot');

%% Finding 'best' observer matrix
P = 10*[l.']
L = place(A',C',P)'
A_L = [(A-(L*C))];
B_L = [B];
C_L = [C];
D_L= [D];

%states
states = {'x' 'x_dot' 'q_1' 'q_1_dot' 'q_2' 'q_2_dot'};

% input for observer
inputs = {'f'};

% output after obserever
outputs = {'x' ;'q_2'};

%creates statesspace model
sys_c2 = ss(A_L,B_L,C_L,D_L,'statename',states,'inputname',inputs,'outputname',outputs);
t = 0:0.1:200;
f = 50*ones(size(t));
[y,t,x]=lsim(sys_c2,f,t);

%simulates response
[AX,H1,H2] = plotyy(t,y(:,1),[t,t],[y(:,2)],'plot');
title('Best Observer for x1 and theta2');
```

```
MATLAB Command Window                                          Page 1


Quad =

        1000           0           0           0           0           0
           0           0           0           0           0           0
           0           0     1000000           0           0           0
           0           0           0           0           0           0
           0           0           0           0     1000000           0
           0           0           0           0           0           0


K =

   1.0e+03 *

    0.1000    0.5536    1.1630    3.6050    2.0958    1.2371


l =

  -0.1584 + 1.0534i
  -0.1584 - 1.0534i
  -0.1945 + 0.2008i
  -0.1945 - 0.2008i
  -0.0759 + 0.7319i
  -0.0759 - 0.7319i


P =

  Columns 1 through 4

  -1.5842 +10.5338i  -1.5842 -10.5338i  -1.9447 + 2.0083i  -1.9447 - 2.0083i

  Columns 5 through 6

  -0.7590 + 7.3194i  -0.7590 - 7.3194i


L =

    5.8788    1.7539
  100.8960  -21.4786
 -336.2784   42.5440
 -606.5829  112.4774
   -2.0545    2.6971
  -19.9612   76.5010

>>
```

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

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

If there is U_∞ such that:

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

Disturbances will be rejected if they are Gaussian in nature.