ENPM667 - Control of Robotic Systems

Technical Report of Final Project

James Clark School of Engineering
Rohith Vikram Saravanan
UID: 119474198
Nazrin Gurbanova
UID: 120426469
Ishan Kharat
UID: 120427145

(2023)

Table of Contents

Introduction

This project report involves acquiring the state space representation for a crane featuring two suspended loads actuated by external force F. The main objective of this project is to design both an LQR and LQG controller for the system.

To initiate with, the system's equations of motion to be used in the Lagrangian method are calculated. As a result, in the second chapter, we end up obtaining a nonlinear state-space format for the crane model.

Subsequently, the system is linearized around equilibrium points in the third chapter. With the linearized system the equations of motion in the state-space form are redefined. Forming new state-space representation is followed by establishing conditions under which the linearized system becomes controllable.

Before devising LQR controller, controllability under the provided conditions is evaluated in the next phase. In case the system proves to be controllable, an LQR controller is designed. The results are simulated based on initial conditions and applied to both the linearized and nonlinear systems. Finally, the stability of the system is assessed via Lyapunov’s indirect method.

Having designed the LQR controller, observability is needed to be checked for the output vectors. Luenberger observer is derived for each specified output vector, taking the system's observability into consideration. Then, for both the linearized and nonlinear systems simulations are conducted to monitor the response to initial conditions and unit step input. Ultimately, output feedback controller is designed for the smallest output vector with the help of LQG method and the result is applied to the nonlinear system. Simulation results have been provided depicting the performance of LQG controller.

A. Equations of motion and the corresponding nonlinear state-space representation

Given: Consider a crane that moves along a 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 m_1 and m_2, and the lengths of the cables are l_1 and l_2, respectively. The following figure depicts the crane and associated variables used throughout this project.

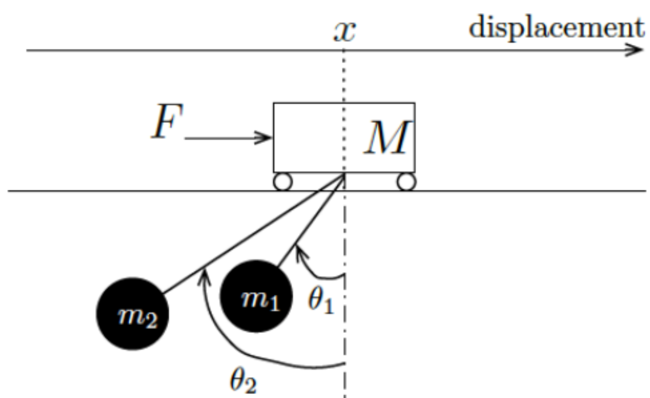

Figure 1 Crane with two loads

I. Lagrange equations

The equations for the position of the cables with m_1 and m_2 mass & θ_1 and θ_2 angles:

	Cable 1	Cable 2
Along x	$x - l_1 sin\theta_1$	$x - l_2 sin\theta_2$
Along y	$-l_1 cos\theta_1$	$-l_2 cos\theta_2$

Figure 2 Equations for the position of the cables along x and y

Using table above, the equations as a function of θ_1 and θ_2:

$$x_1 = (x - l_1 sin\theta_1)\hat{\imath} - l_1 cos\theta_1 \hat{\jmath} \quad (1)$$
$$x_2 = (x - l_2 sin\theta_2)\hat{\imath} - l_2 cos\theta_2 \hat{\jmath} \quad (2)$$

Differentiating the position equations, we get the ones for velocity:

$$v_1 = \left(\dot{x} - l_1\dot{\theta}_1 \cos\theta_1\right)\hat{\imath} + l_1\dot{\theta}_1 \sin\theta_1 \hat{\jmath} \quad (3)$$
$$v_2 = \left(\dot{x} - l_2\dot{\theta}_2 \cos\theta_2\right)\hat{\imath} + l_2\dot{\theta}_2 \sin\theta_2 \hat{\jmath} \quad (4)$$

Kinetic Energy (K) and Potential energy (P):

$$K = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m_1\left(\dot{x} - l_1\dot{\theta}_1 \cos\theta_1\right)^2 + \frac{1}{2}m_1\left(l_1\dot{\theta}_1 \sin\theta_1\right)^2 + \frac{1}{2}m_2\left(\dot{x} - l_2\dot{\theta}_2 \cos\theta_2\right)^2 + \frac{1}{2}m_2\left(l_2\dot{\theta}_2 \sin\theta_2\right)^2 \qquad (5)$$

$$P = -m_1 g l_1 cos\theta_1 - m_2 g l_2 cos\theta_2 \qquad (6)$$

Total energy:

$$L = K - P = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m_1\left(\dot{x} - l_1\dot{\theta}_1 \cos\theta_1\right)^2 + \frac{1}{2}m_1\left(l_1\dot{\theta}_1 \sin\theta_1\right)^2 + \frac{1}{2}m_2\left(\dot{x} - l_2\dot{\theta}_2 \cos\theta_2\right)^2 + \frac{1}{2}m_2\left(l_2\dot{\theta}_2 \sin\theta_2\right)^2 + m_1 g l_1 cos\theta_1 + m_2 g l_2 cos\theta_2 \qquad (7)$$

$$L = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m_1\dot{x}^2 - m_1\dot{x}l_1\dot{\theta}_1 \cos\theta_1 + \frac{1}{2}m_1 l_1^2\dot{\theta}_1^2 \cos^2\theta_1 + \frac{1}{2}m_1 l_1^2\dot{\theta}^2{}_1 \sin^2\theta_1 + \frac{1}{2}m_2\dot{x}^2 - m_2\dot{x}l_2\dot{\theta}_2 \cos\theta_2 + \frac{1}{2}m_2 l_2^2\dot{\theta}^2{}_2 \cos^2\theta_2 + \frac{1}{2}m_2 l_2^2\dot{\theta}_2^2 \sin^2\theta_2 + m_1 g l_1 cos\theta_1 + m_2 g l_2 cos\theta_2 \qquad (8)$$

Simplifying the equation:

$$L = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}(m_1 + m_2)\dot{x}^2 + \frac{1}{2}m_1 l_1^2\dot{\theta}_1^2 + \frac{1}{2}m_2 l_2^2\dot{\theta}_2^2 - \dot{x}\left(m_1\dot{x}l_1\dot{\theta}_1 \cos\theta_1 + m_2 l_2\dot{\theta}_2 \cos\theta_2\right) + m_1 g l_1 cos\theta_1 + m_2 g l_2 cos\theta_2 \qquad (9)$$

Lagrange equations:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) - \left(\frac{\partial L}{\partial x}\right) = F \qquad (10)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta_1}}\right) - \left(\frac{\partial L}{\partial \theta_1}\right) = 0 \qquad (11)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta_2}}\right) - \left(\frac{\partial L}{\partial \theta_2}\right) = 0 \qquad (12)$$

II. $\ddot{x}$ calculation

Computation of the Lagrangian equation for 2.10:

$$\left(\frac{\partial L}{\partial \dot{x}}\right) = M\dot{x} + (m_1 + m_2)\dot{x} - m_1 l_1 \dot{\theta}_1 \cos\theta_1 - m_2 l_2 \dot{\theta}_2 \cos\theta_2 \qquad (13)$$

Differentiating to the first term for the equation:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) = M\ddot{x} + (m_1 + m_2)\ddot{x} - \left(m_1 l_1 \ddot{\theta}_1 \cos\theta_1 - m_1 l_1 \dot{\theta}_1^2 sin\theta_1\right) - \left(m_2 l_2 \ddot{\theta}_2 \cos\theta_2 - m_2 l_2 \dot{\theta}_2^2 sin\theta_2\right) \ (14)$$

Second term calculation gives zero as L is not dependent on x:

$$\left(\frac{\partial L}{\partial x}\right) = 0 \qquad (15)$$

Hence, Lagrangian equation 1 becomes:

$$M\ddot{x} + (m_1 + m_2)\ddot{x} - \left(m_1 l_1 \ddot{\theta}_1 \cos\theta_1 - m_1 l_1 \dot{\theta}_1^2 sin\theta_1\right) - \left(m_2 l_2 \ddot{\theta}_2 \cos\theta_2 - m_2 l_2 \dot{\theta}_2^2 sin\theta_2\right) = F \quad (16)$$

$$(M + m_1 + m_2)\ddot{x} - m_1 l_1 \ddot{\theta}_1 \cos\theta_1 + m_1 l_1 \dot{\theta}_1^2 sin\theta_1 - m_2 l_2 \ddot{\theta}_2 \cos\theta_2 + m_2 l_2 \dot{\theta}_2^2 sin\theta_2 = F \qquad (17)$$

Obtaining $\ddot{x}$ from Eq 17:

$$\ddot{x} = \frac{\left(F + m_1 l_1 \ddot{\theta}_1 \cos\theta_1 - m_1 l_1 \dot{\theta}_1^2 sin\theta_1 + m_2 l_2 \ddot{\theta}_2 \cos\theta_2 - m_2 l_2 \dot{\theta}_2^2 sin\theta_2\right)}{(M + m_1 + m_2)} \qquad (18)$$

III. $\ddot{\theta}_1$ calculation

Moving onto calculation of Eq 11:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta_1}}\right) - \left(\frac{\partial L}{\partial \theta_1}\right) = 0 \qquad (11)$$

$$\left(\frac{\partial L}{\partial \dot{\theta_1}}\right) = m_1 l_1^2 \dot{\theta}_1 - m_1 \dot{x} l_1 cos\theta_1 \qquad (19)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta_1}}\right) = m_1 l_1^2 \ddot{\theta}_1 - m_1 \ddot{x} l_1 cos\theta_1 + m_1 \dot{x} l_1 \dot{\theta}_1 sin\theta_1 \qquad (20)$$

$$\frac{\partial L}{\partial \theta_1} = m_1 l_1 \dot{\theta}_1 \dot{x} \sin\theta_1 - m_1 l_1 g sin\theta_1 \qquad (21)$$

Hence, Lagrangian equation 2 becomes:

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

$$m_1 l_1^2 \ddot{\theta}_1 - m_1 \ddot{x} l_1 cos\theta_1 + m_1 l_1 g sin\theta_1 = 0 \qquad (22)$$

$\ddot{\theta}_1$ can be obtained:

$$\ddot{\theta}_1 = \frac{\ddot{x}cos\theta_1 - gsin\theta_1}{l_1} \qquad (23)$$

IV. $\ddot{\theta}_2$ calculation

Moving onto calculation of Eq 12:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta_2}}\right) - \left(\frac{\partial L}{\partial \theta_2}\right) = 0 \qquad (12)$$

$$\frac{\partial L}{\partial \dot{\theta_2}} = m_2 l_2^2 \dot{\theta}_2 - m_2 \dot{x} l_2 cos\theta_2 \qquad (24)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta_2}}\right) = m_2 l_2^2 \ddot{\theta}_2 - m_2 \ddot{x} l_2 cos\theta_2 + m_2 \dot{\theta}_2 \dot{x} l_2 sin\theta_2 \qquad (25)$$

$$\frac{\partial L}{\partial \theta_2} = m_2 \dot{x} l_2 \dot{\theta}_2 sin\theta_2 - m_2 l_2 g sin\theta_2 \qquad (26)$$

Hence, Lagrangian equation 3 becomes:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta_2}}\right) - \left(\frac{\partial L}{\partial \theta_2}\right) = m_2 l_2^2 \ddot{\theta}_2 - m_2 \ddot{x} l_2 cos\theta_2 + m_2 \dot{\theta}_2 \dot{x} l_2 sin\theta_2 - m_2 \dot{x} l_2 \dot{\theta}_2 sin\theta_2 + m_2 l_2 g sin\theta_2 =$$
$$m_2 l_2^2 \ddot{\theta}_2 - m_2 \ddot{x} l_2 cos\theta_2 + m_2 l_2 g sin\theta_2 = 0 \qquad (27)$$

$\ddot{\theta}_2$ can be obtained: $\ddot{\theta}_2 = \frac{\ddot{x}cos\theta_2 - gsin\theta_2}{l_2}$ (28)

V. State-Space representation (Non-linear system)

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

B. Linearization of the non-linear system

The obtained state-space representation highlights the crane system as a non-linear system; hence, it is required to linearize the system around the equilibrium point $(x = 0, \theta_1 = 0, \theta_2 = 0)$. Let's assume the limiting condition at equilibrium:

$$\theta_1^2 \approx 0$$
$$\theta_2^2 \approx 0$$
$$sin\theta_1 \approx \theta_1$$
$$sin\theta_2 \approx \theta_2$$
$$cos\theta_1 \approx 1$$
$$cos\theta_2 \approx 1$$

With this assumptions, linearized system:

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

Force can be derived as an input from this equation:

$$\dot{X} = AX + BU$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{gm_1}{M} & 0 & -\frac{gm_2}{M} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{g(M+m_1)}{Ml_1} & 0 & -\frac{gm_2}{Ml_1} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{gm_1}{Ml_2} & 0 & -\frac{g(M+m_2)}{Ml_2} & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ \frac{1}{Ml_1} \\ 0 \\ \frac{1}{Ml_2} \end{bmatrix}$$

$$Y = CX + DU$$

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

Controllability matrix (n=6): $Controllability\ matrix = [B \quad AB \quad A^2B \quad A^3B \quad A^4B \quad A^5B]$

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

where

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

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

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

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

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

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

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

MATLAB has been used to derive these outputs. In addition, the rank of the matrix has been found using MATLAB and it is 6.

C. Controllability conditions for the linearized system

If a linear time-invariant system is controllable, its Grammian controllability matrix will demonstrate invertibility. Moreover, a full-rank controllability matrix ensures the invertibility of the Grammian controllability matrix. Evaluating the rank of the controllability matrix serves as a means to ascertain the system's controllability.

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

where

$$\sigma_1 = -\frac{g\,m_1}{M^2\,l_2} - \frac{g\,m_2}{M^2\,l_2}$$

$$\sigma_2 = \frac{\frac{g^2\,m_2\,(M+m_1)}{\sigma_8} + \frac{g^2\,m_2\,(M+m_2)}{\sigma_8}}{M\,l_2} + \frac{\frac{g^2\,(M+m_1)^2}{\sigma_8} + \frac{g^2\,m_1\,m_2}{\sigma_8}}{M\,l_2}$$

$$\sigma_3 = \frac{\frac{g^2\,m_1\,(M+m_1)}{\sigma_8} + \frac{g^2\,m_1\,(M+m_2)}{\sigma_8}}{M\,l_2} + \frac{\frac{g^2\,(M+m_2)^2}{\sigma_8} + \frac{g^2\,m_1\,m_2}{\sigma_8}}{M\,l_2}$$

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

$$\sigma_5 = -\frac{g\,(M+m_2)}{\sigma_8} - \frac{g\,m_1}{\sigma_8}$$

$$\sigma_6 = -\frac{g\,(M+m_1)}{\sigma_8} - \frac{g\,m_2}{\sigma_8}$$

$$\sigma_7 = \frac{g^2\,m_1\,m_2}{M^2\,l_2}$$

$$\sigma_8 = M^2\,{l_2}^2$$

To acquire the condition when the system is controllable, $l_1 = l_2, l_1 = 0$ or $l_2 = 0$ is applied and rank is found. In each situation the system is uncontrollable. The conditions when the system becomes controllable may be accepted as $l_1 \neq l_2$ and $l_1 \neq 0, l_2 \neq 0$.

D. LQR controller and Lyapunov stability

Design of LQR controller

The next step involves inputting the mass and length values of the system to design an LQR controller. This controller's parameters will be adjusted to minimize weight. After implementing the controller, the system's response to initial conditions will be simulated in MATLAB. At the end, the stability of the closed-loop system will be checked utilizing Lyapunov's indirect method; whether it exhibits local or global stability will be analyzed.

For the LQR Controller, if (A, B) is stabilizable, then we can look for k that minimizes the following cost:

$$J\left(k, \vec{X}(0)\right) = \int_0^\infty \vec{X}^T(t) Q \vec{X}(t) + \vec{U}_k{}^T R \vec{U}_k(t) dt$$

where Q and R are positive definite matrices.

Optimal solution: $K = -R^{-1} B_K^T P$

P is the symmetric positive definite solution for Stationary Ricatti Equation:

$$A^T P + PA - PBR^{-1} \mathrm{B}^{\mathrm{T}} P = -Q$$

The selection of Q depends on the desired level of performance error, whereas R is chosen based on the actuator cost. To preserve the input used, R is increased to penalize the actuator, while for improved performance, the value of Q is augmented to penalize the error.

A trial-and-error process is done to determine the values for Q and R. Firstly, initial values for both of them are chosen with the help of which gain matrix is acquired. Following this, with respect to the output response the values of Q and R are adjusted iteratively to reach favorable outcomes. Validation of the Q and R matrices involves plotting various output graphs, enabling assessment and confirmation of the chosen values' effectiveness in achieving the desired control behavior.

```
Controllability_matrix = 6×6
10^-3 ×
         0    1.0000         0   -0.1472         0    0.1419
    1.0000         0   -0.1472         0    0.1419         0
         0    0.0500         0   -0.0319         0    0.0227
    0.0500         0   -0.0319         0    0.0227         0
         0    0.1000         0   -0.1128         0    0.1249
    0.1000         0   -0.1128         0    0.1249         0
```

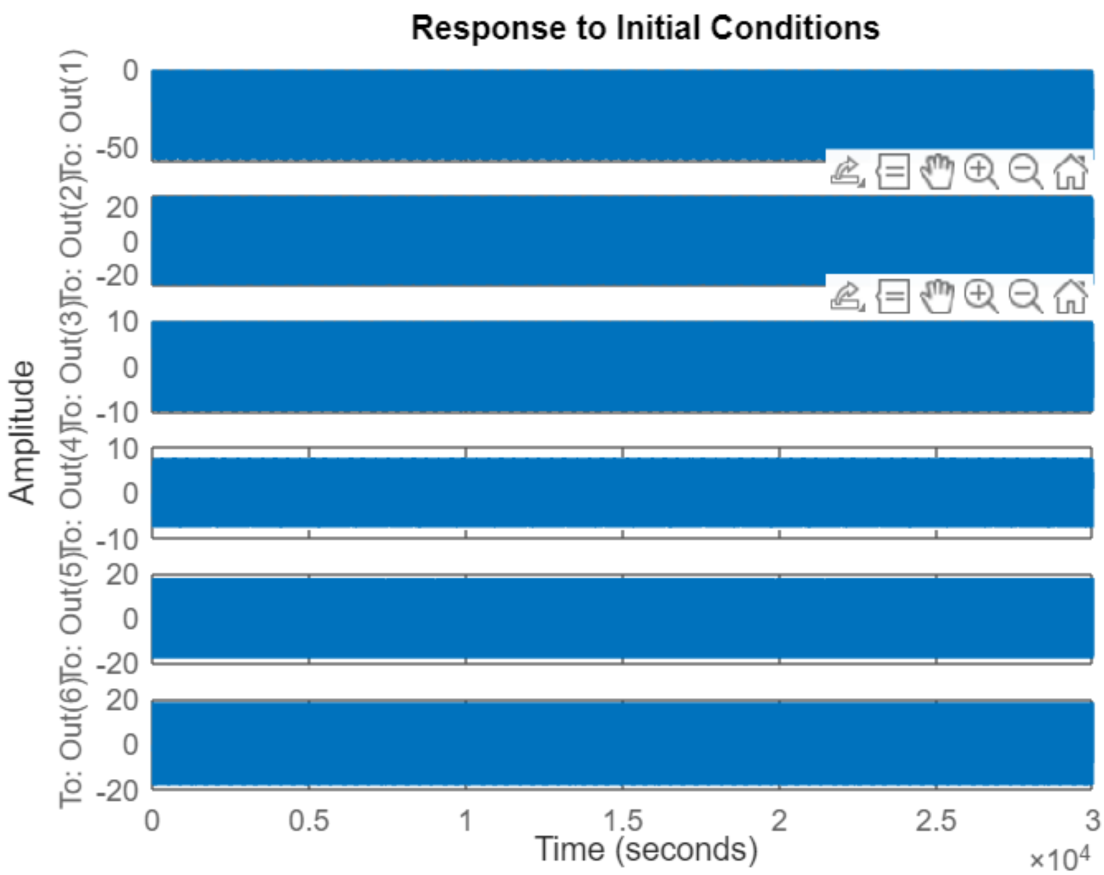

Figure 3 LQR response for open-loop system

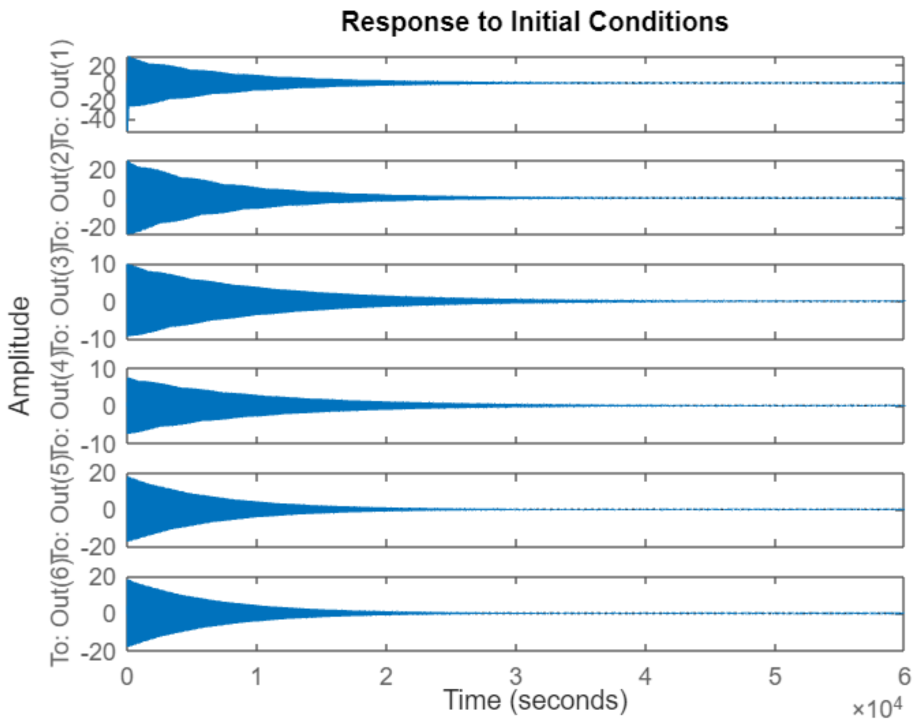

Figure 4 LQR response for closed-loop system - with linear state feedback

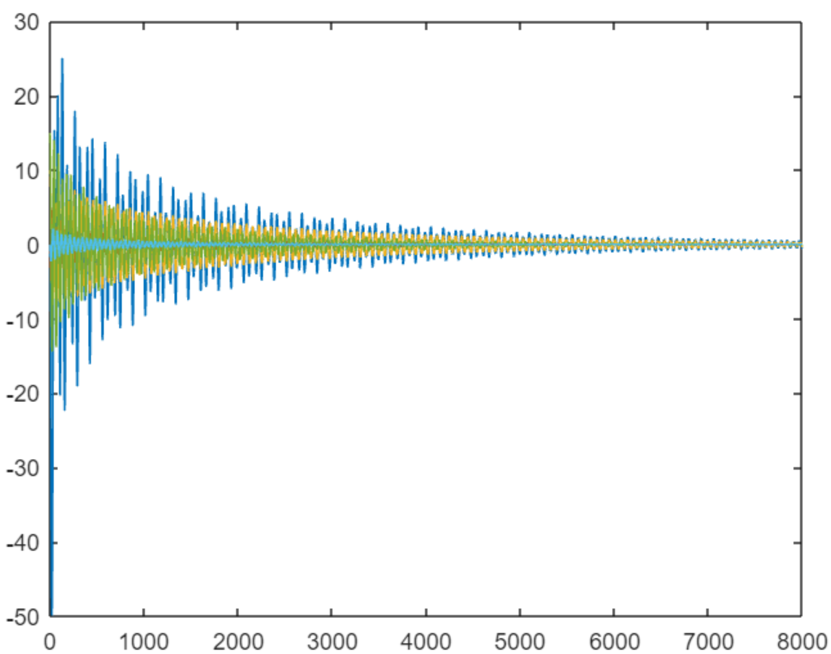

Figure 5 Plot for non-linear system using LQR controller

Lyapunov’s indirect method stability

According to Lyapunov's indirect method, the eigenvalues of the pair (A, Bk) need to be in the left half-plane. Below are the eigenvalues for the system incorporating the LQR controller:

EigenValues = 6×1 complex

-0.0001 + 0.7285i

-0.0001 - 0.7285i

-0.0001 + 1.0430i

-0.0001 - 1.0430i

-0.0243 + 0.0243i

-0.0243 - 0.0243i

It is apparent from the provided eigenvalues that all reside in the left half-plane. As per Lyapunov’s indirect stability criterion, it affirms that the system is stable.

E. Observability of the system

The linear state equation is considered observable over the interval $[t_0, t_f]$ if every initial state $\mathrm{x}(\mathrm{t}_0) = \mathrm{x}_0$ can be uniquely identified by its corresponding response yt within the range of $[t_0, t_f]$.

Whenever, the pair (A^T, C^T) is controllable then we say that (A,C) is observable.

The rank of the observability matrix can be determined by using:

$$rank[C^T\ A^T C^T\ \ \dots (A^T)^{n-1} C^T]$$

$$n = 6$$

$$[C^T \quad A^T C^T \quad (A^2)^T C^T \quad (A^3)^T C^T \quad (A^4)^T C^T \quad (A^5)^T C^T]$$

when rank is full, then the system is observable.

The output for the question 1 - (E) is shown below for all the given output vectors:

–> For the output vector - x(t) Rank of the observability matrix is 6. So, the system is observable.

–> For the output vector - (θ1(t), θ2(t)) Rank of the observability matrix is 4. So, the system is not observable.

–> For the output vector - (x(t), θ2(t)) Rank of the observability matrix is 6. So, the system is observable –> For the output vector - (x(t), θ1(t), θ2,(t)) Rank of the observability matrix is 6. So, the system is observable.

F. Luenberger observer

The formulation of Luenberger observer is as follows:

$$\dot{\hat{X}} = A\hat{X} + BU + L(Y - \hat{Y}) = A\hat{X} + BU + LC(X - \hat{X})$$

Based on the true output values, Luenberger observer corrects the state estimate aiming to choose suitable value for L. The error between estimated and actual state is required to exponentially decrease to zero; therefore, $X_e = (X - \hat{X}) \to 0$

Considering error dynamics:

$$\dot{X} - \dot{\hat{X}} = AX + BU - A\hat{X} - BU - LC(X - \hat{X}) = AX - A\hat{X} - LC(X - \hat{X}) = (A - LC)(X - \hat{X})$$

$$\dot{X}_e = (A - LC)X_e$$

To ensure stability of the system $(A^T - C^T L^T)$ should have negative real parts. Using pole placement methods, values for L can be obtained. Usually, it is needed to acquire the estimate of the state fast; hence, poles are placed apart from the imaginary axis. However, to do this there is a limit. In the conditions when there is a noise in the system, the noise is more likely to amplify its magnitude and lead to an undesired estimate.

While simulating, it has been proved that poles those are placed far from the linearized system tend to converge to the actual state position fast, although initial error of estimate became large. Additionally, while using on the non-linear system the error did not reduce. Having changed between linear and non-linear options for the following poles resulted in a suitable response for each output vector.

Leunberger Observer on non-linear system ($f(\hat{X}, U)$ needs the formula as follows:

$$\dot{\hat{X}} = f(\hat{X}, U) + LC(X - \hat{X})$$

```
L1 = 6×1
   14.0000
   74.3814
 -160.8343
  162.8432
  -22.2329
 -318.2588
```

```
L3 = 6×2
    8.2735    1.8102
   20.8628    9.5757
  -15.5887  -16.2526
    5.8626   -0.6985
    0.5281    5.7265
    2.0580    7.0962
```

```
Poles_placements = 6×1
   -0.5000
   -1.0000
   -2.0000
   -2.5000
   -3.5000
   -4.5000
L1 = 6×1
   14.0000
   74.3814
 -160.8343
  162.8432
  -22.2329
 -318.2588
L3 = 6×2
    8.2735    1.8102
   20.8628    9.5757
  -15.5887  -16.2526
    5.8626   -0.6985
    0.5281    5.7265
    2.0580    7.0962
L4 = 6×3
    6.7691    0.5147    0.0000
   10.5617    0.4905   -0.9809
    0.5635    5.7309    0.0000
    1.6443    7.1457   -0.0490
    0.0000    0.0000    1.5000
    0.0000   -0.0981   -0.5791
```

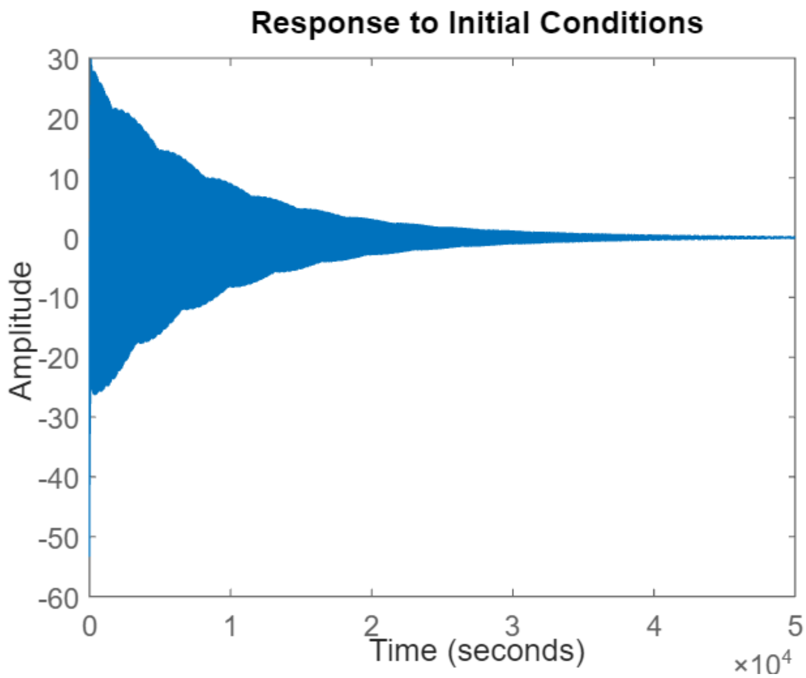

Figure 6 Leunberger Observer's response for initial conditions of X(t)

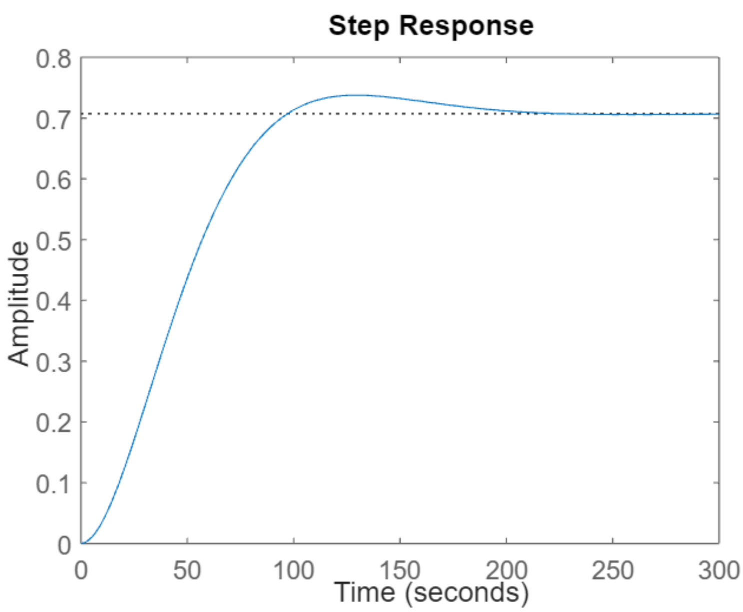

Figure 7 Leunberger Observer's response for unit step input for X(t)

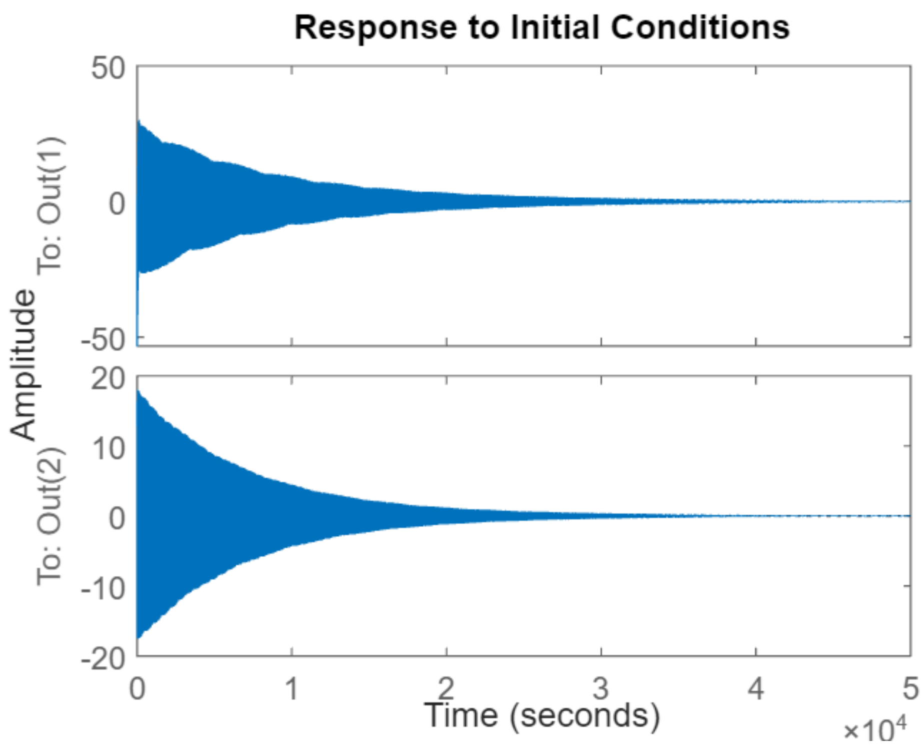

Figure 8 Leunberger Observer's response for initial conditions for $X(t), \theta_2$

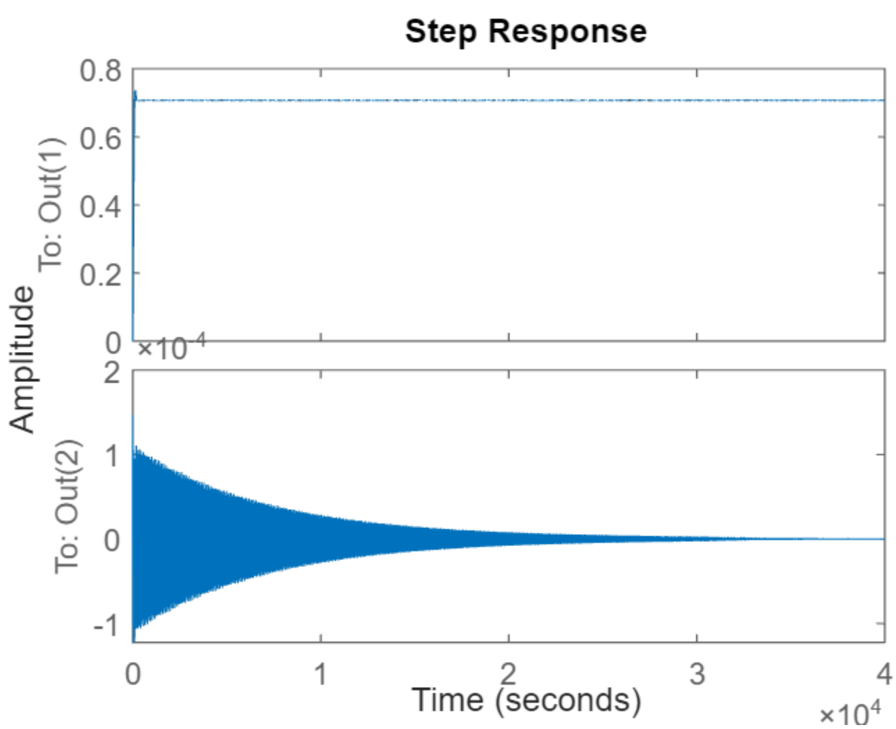

Figure 9 Leunberger Observer's response for unit step input for $X(t), \theta_2$

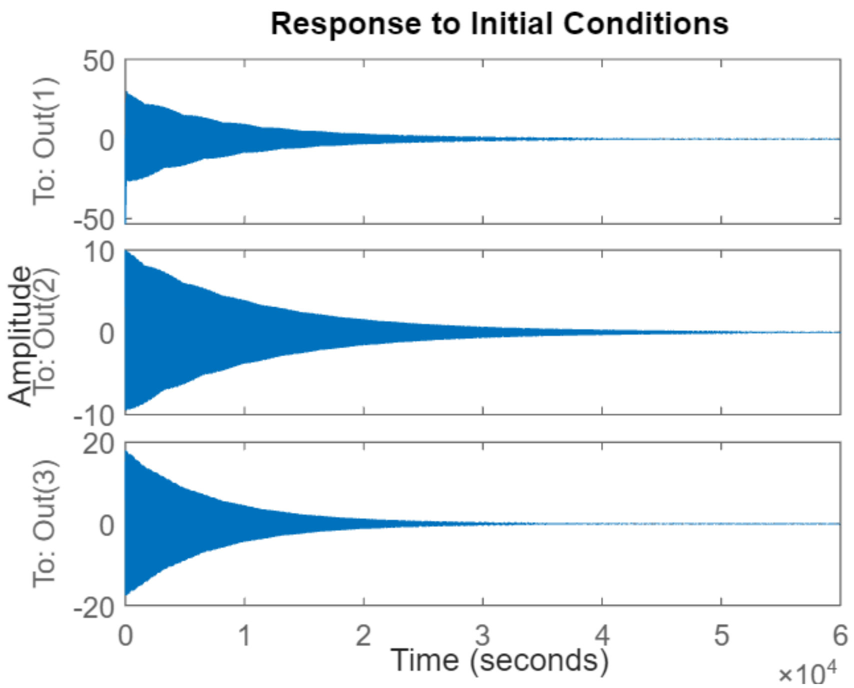

Figure 10 Leunberger Observer's response for initial conditions for $X(t), \theta_1, \theta_2$

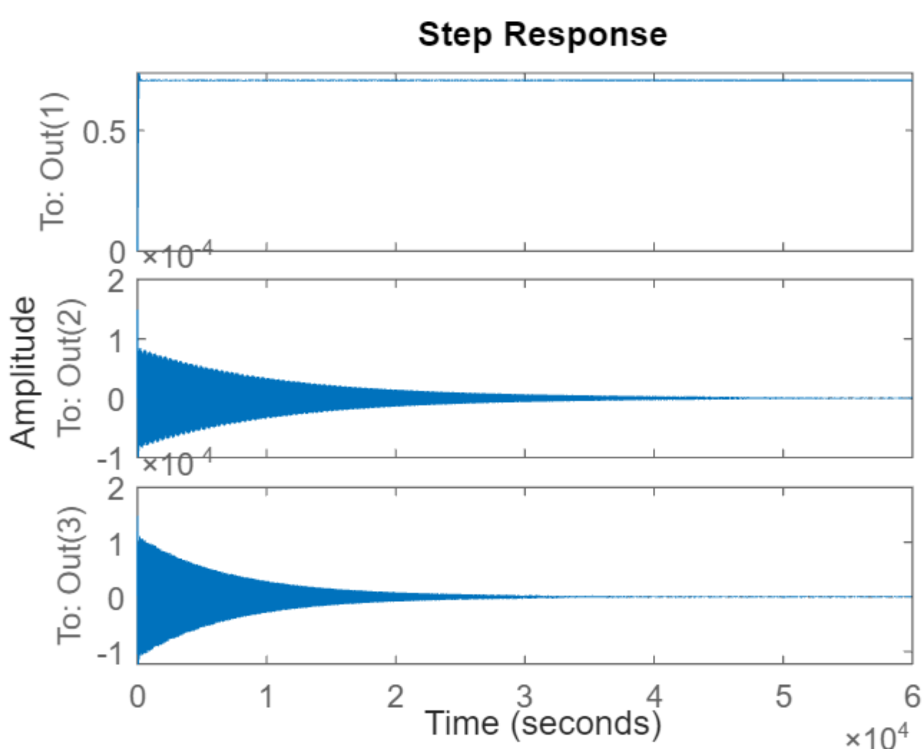

Figure 11 Leunberger Observer's response for unit step input for $X(t), \theta_1, \theta_2$

Nonlinear systems observer gain matrices:

```
Non linear systems
L1 = 6×1
   14.0000
   74.3814
 -160.8343
  162.8432
  -22.2329
 -318.2588
L3 = 6×2
    8.2735    1.8102
   20.8628    9.5757
  -15.5887  -16.2526
    5.8626   -0.6985
    0.5281    5.7265
    2.0580    7.0962
L4 = 6×3
    6.7691    0.5147    0.0000
   10.5617    0.4905   -0.9809
    0.5635    5.7309    0.0000
    1.6443    7.1457   -0.0490
    0.0000    0.0000    1.5000
    0.0000   -0.0981   -0.5791
```

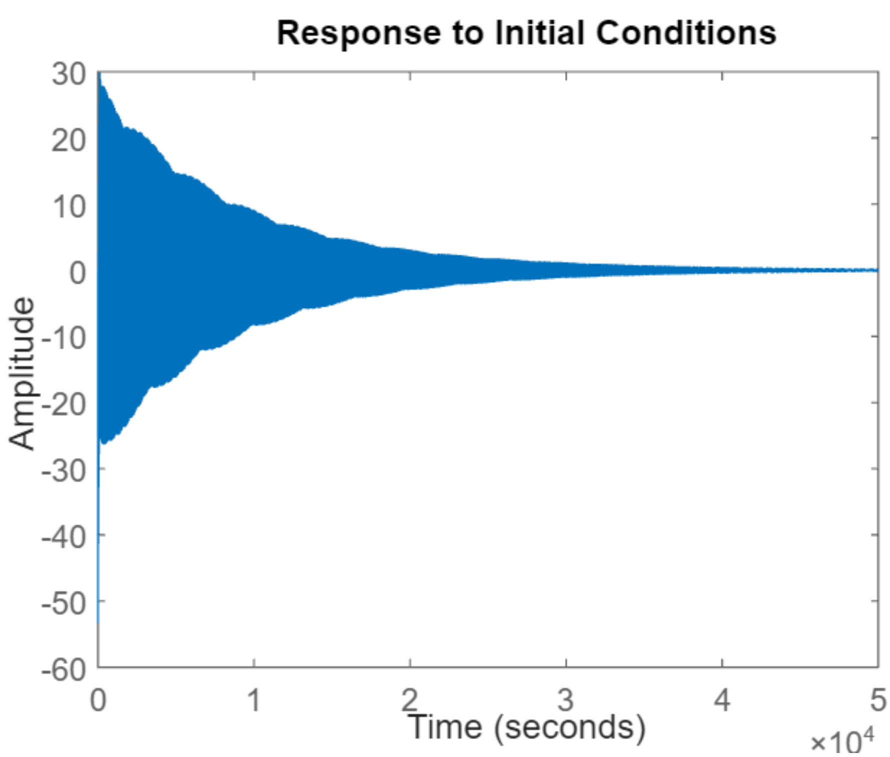

Figure 12 Leunberger Observer's initial response condition for X(t)

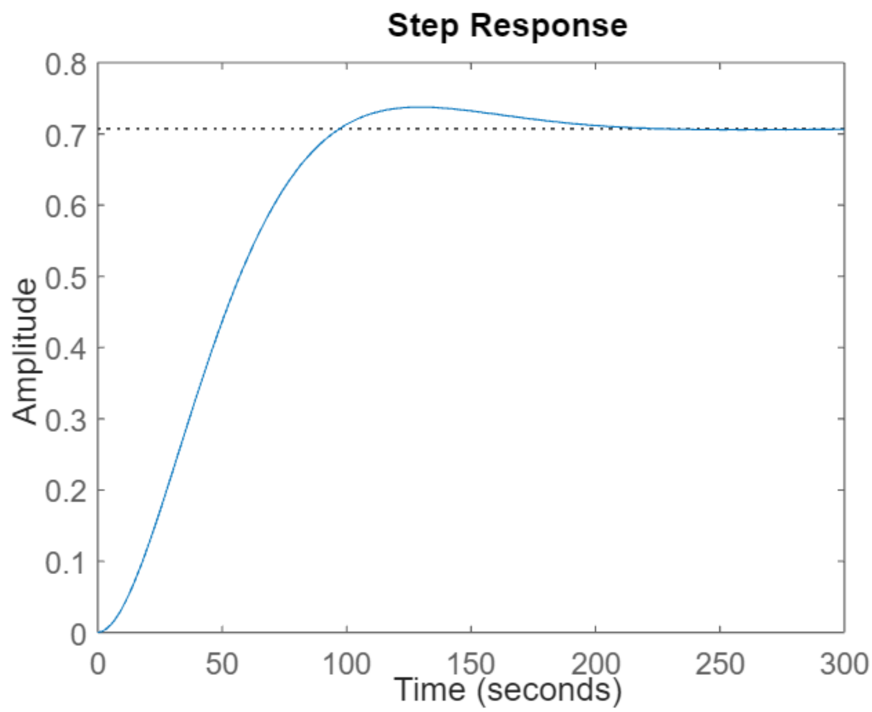

Figure 13 Leunberger Observer's response for unit step input for X(t)

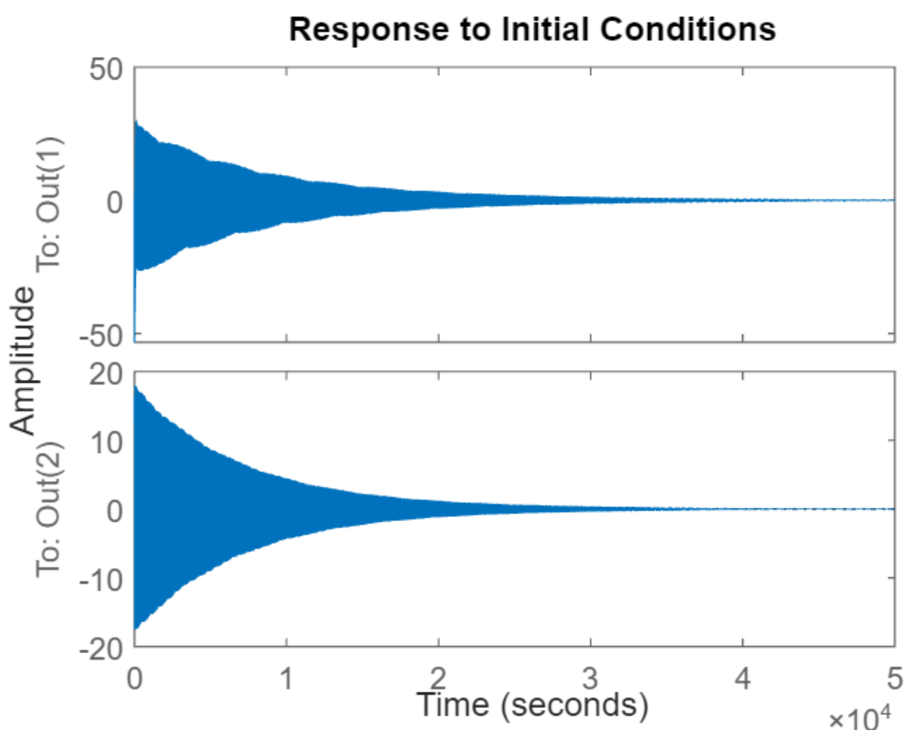

Figure 14 Leunberger Observer's initial response condition for $X(t), \theta_2$

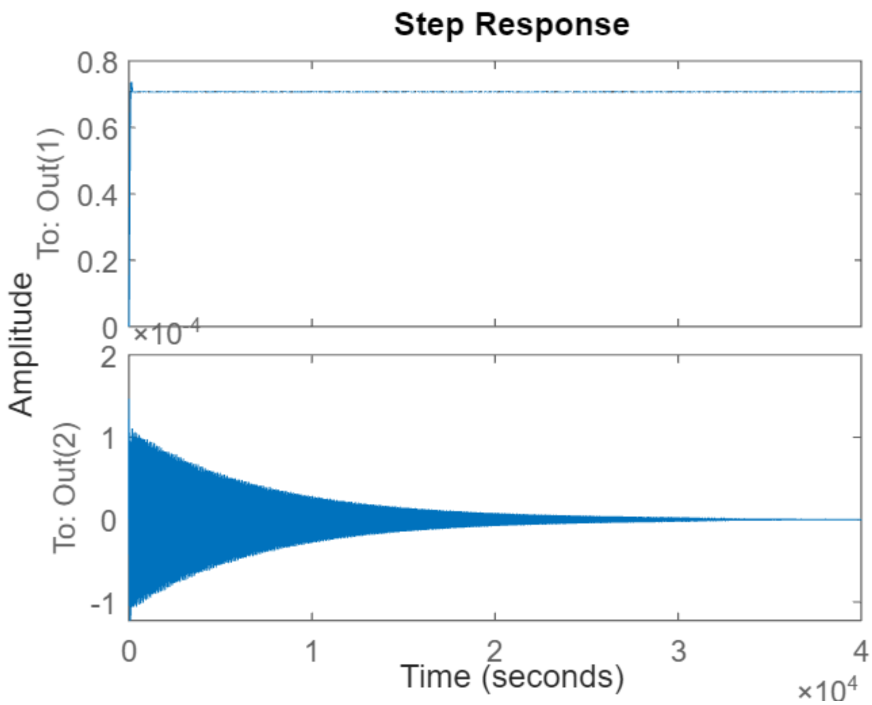

Figure 15 Leunberger Observer's response for unit step input for $X(t), \theta_2$

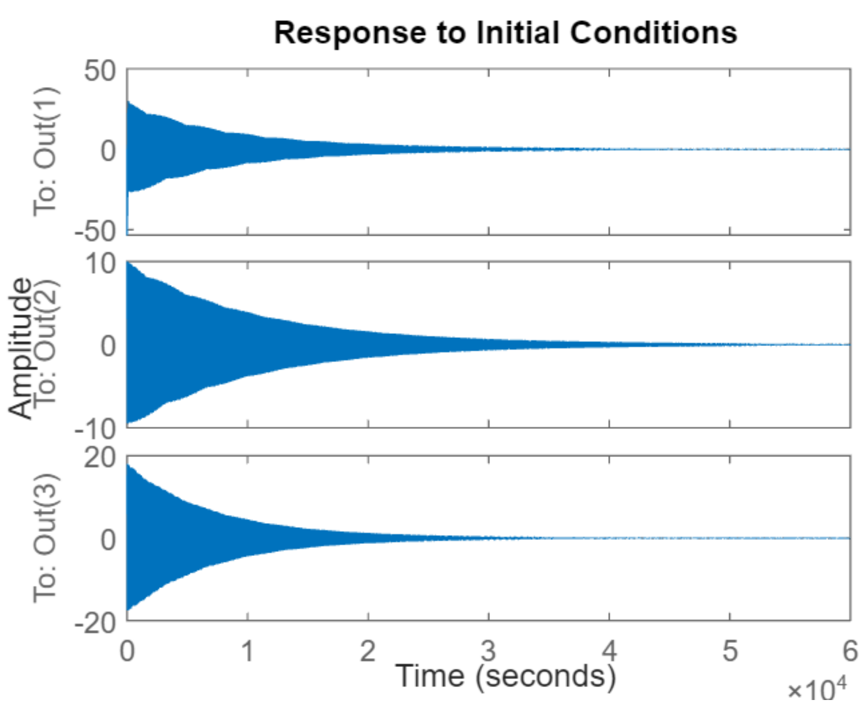

Figure 16 Leunberger Observer's initial response condition for $X(t), \theta_1, \theta_2$

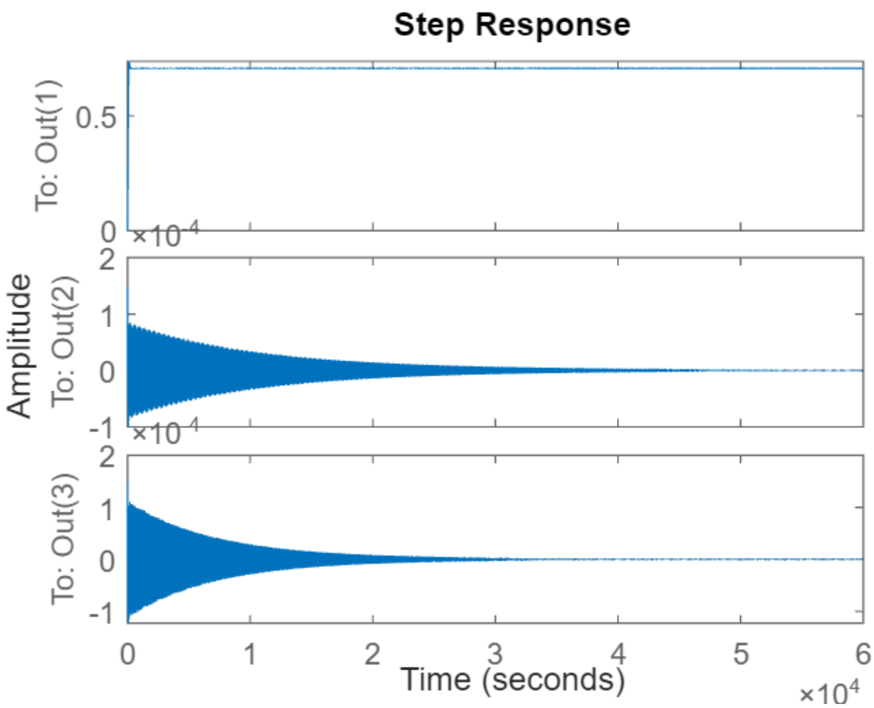

Figure 17 Leunberger Observer's response for unit step input for $X(t), \theta_1, \theta_2$

G. Kalman filter and LQG controller

To optimize Reference Tracking, we aim to minimize the following cost function in order to asymptotically track a constant reference on x:

$$\int_0^\infty \left(X(t) - X(d)\right)^T (t) Q \left(X(t) - X(d)\right) + (U_k - U_\infty)^T R\, (U_k - U_\infty) dt$$

In case there is U_∞ such that $AX_d + B_k U_\infty$, then optimal solution for the system is as follows:

$$U(t) = K(X(t) - X(d))$$

where

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

P is the symmetric positive definite matrix.

The controller is adjusted to reduce the optimal reference tracking cost function.

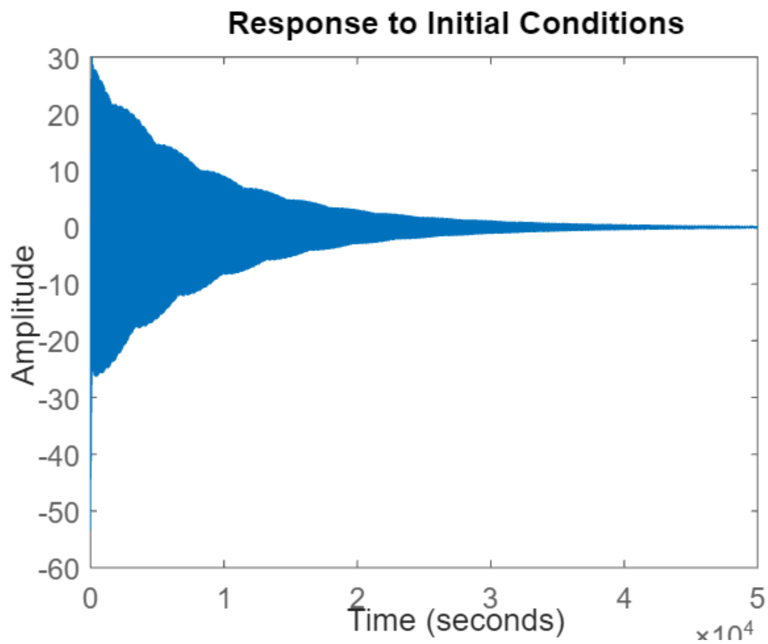

Figure 18 LQG controller's performance response for initial conditions – System 1

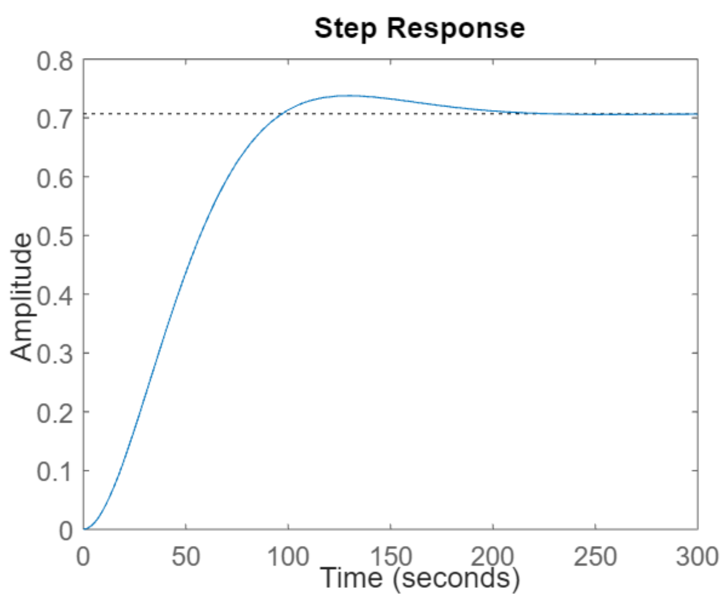

Figure 19 LQG controller's performance response for unit step input – System 1

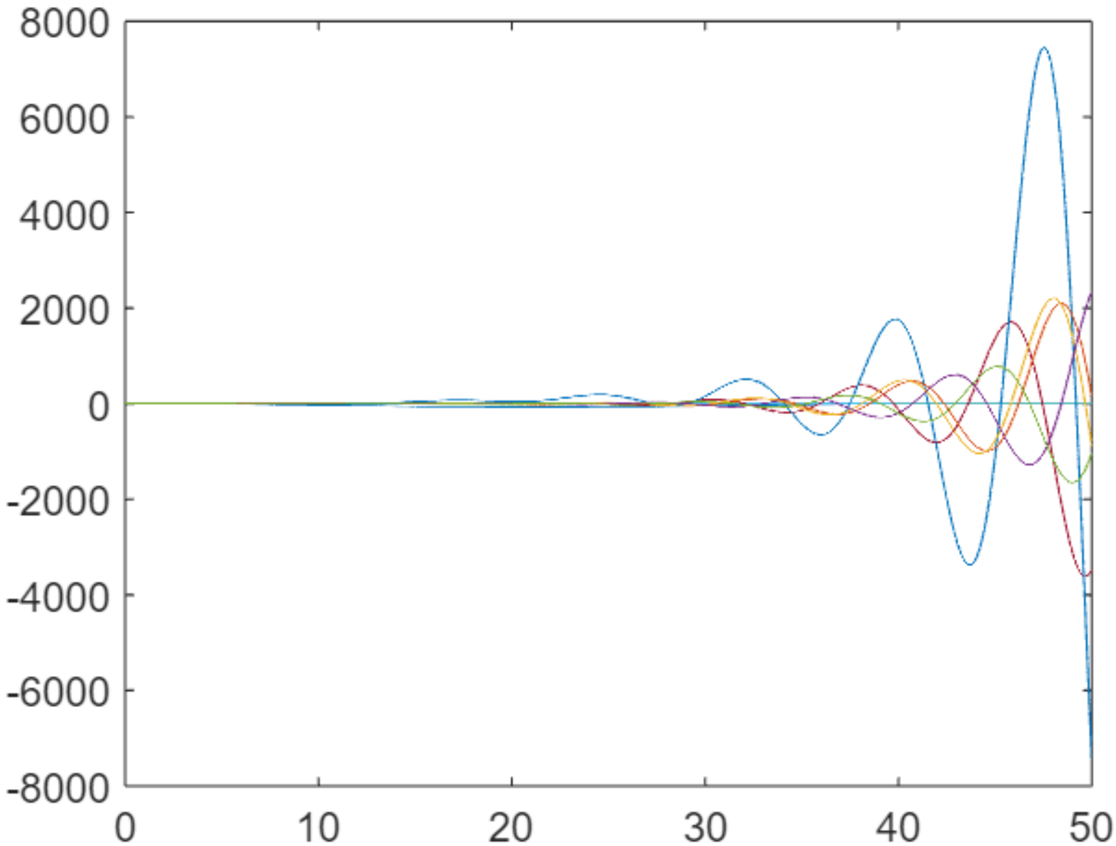

Figure 20 Plot for non-linear system using LQG controller

As far as the cart's design is concerned, it can accommodate constant force disturbances. Controllers will take into account force disturbances based on the assumption that they are Gaussian.

APPENDIX

Problem_C

```
% given variables
% M_c - Mass of the cart; m1 - mass of the load1; m2 - mass of load2
% l1 - lenght of cable1; l2 - length of the cable2; g - gravitation


syms M m1 m2 l1 l2 g;


% A Matrix
disp("A_matrix")


A = [0 1 0 0 0 0;
            0 0 -(g*m1)/M 0 -(g*m2)/M 0;
            0 0 0 1 0 0;
            0 0 -(g*(M+m1))/(M*l1) 0 -(g*m2)/(M*l1) 0;
            0 0 0 0 0 1;
            0 0 -(g*m1)/(M*l2) 0 -(g*(M+m2))/(M*l2) 0]


% B matrix
disp("B_matrix")


B = [0;
            1/M;
            0;
            1/(M*l1);
            0;
            1/(M*l2)]


% C matrix
disp("C_matrix")


C = [1 0 0 0 0 0;
            0 1 0 0 0 0;
            0 0 1 0 0 0;
            0 0 0 1 0 0;
            0 0 0 0 1 0;
            0 0 0 0 0 1]


% computing the Controllability matrix
```

```
disp("Computing the controllability matrix")
disp("Controllability matrix = [B AB A2_B A3_B A4_B A5_B]")
AB = A*B;
A2_B = A*A*B;
A3_B = A*A*A*B;
A4_B = A*A*A*A*B;
A5_B = A*A*A*A*A*B;


% controllability matrix
disp("controllability_matrix")
Controllability_matrix = [B AB A2_B A3_B A4_B A5_B]


% checking the controllability of the system
disp("checking the controllability of the system")


% computing the determinant of the controllability matrix
disp("determinant of the controllability matrix")
disp(simplify(det(Controllability_matrix)));
disp("Since the Determinant of the controllability matrix is not zero; Hence the
system is controllable")


% Computing the rank of the controllability matrix
disp("Rank of the controllability matrix")
disp(rank(Controllability_matrix));
disp("Since the rank of the controllability is 6, which is full rank; hence the
system is controllable")


% Obtaining conditions on M, m1, m2, l1, l2 for which the linearized system is
controllable
% From the determinant of the controllability matrix we can see that if l1 = l2,
then the determinant becomes zero
% Hence taking the condition l1 = l2 and substituting l1 = l2 in
% controllability matrix
disp("Computing the controllability matrix for the condition l1 = l2")
Control_matrix_l1_l2 = subs(Controllability_matrix,l1,l2)
disp("determinant of the new controllability matrix")
disp(simplify(det(Control_matrix_l1_l2)));
disp("rank of the new controllability matrix")
disp(rank(Control_matrix_l1_l2));
disp("Since the determinant of the new controllability matrix is zero and the
rank is 4 which is not full rank")
disp("Hence the system becomes uncontrollable when l1=l2 and remains controllable
for the rest")
```

Problem_D

```
% substituting the values given in the question for M, m1, m2, l1, l2, g


% Now computing the controllability of the system by substituting these
% values into the controllability matrix


Controllability_matrix = [B AB A2_B A3_B A4_B A5_B]


disp("Checking the the rank of the controllability matrix to check the systems
controllability")
disp(rank(Controllability_matrix));
disp("Since the rank of the matrix is 6 which is full rank, hence the system is
controllable")


disp("choosing the initial condition")
X0 = [0; 0; 10; 0; 15; 0];


% Choosing the values of R and Q which is used to calculate the LQR cost function
R = 0.5;
Q = diag([1, 1, 1, 1, 1, 1])


% we know that the D=0
D = 0;


% LQR response for open loop system
openLoopSystem = ss(A,B,C,D);
figure
initial(openLoopSystem,X0)


% Computing Gain matrix(K), Positive definte matrix(P), Eigen values
[K_matrix, P_matrix, EigenValues] = lqr(A,B,Q,R);
K_matrix
P_matrix
EigenValues
Ac = A-(B*K_matrix);


% LQR response for closed loop system- linear state feedback
closedLoopSystem = ss(Ac,B,C,D);
```

```
figure
initial(closedLoopSystem,X0)


% Non-linear system using LQR controller
Timespan = 0:0.1:8000;
[t1,x1] = ode45(@LQR_controller,Timespan,X0);
% Plot the function output
plot(t1,x1)
```

```
function X_dot = LQR_controller(T1,X)
% Given Initial conditions
% Mass of the Crane, mass of load1, mass of load2, Cable length 1,
% Cable length 2, gravity


M= 1000;
m1= 100;
m2= 100;
l1= 20;
l2= 10;
g= 9.81;


% A matrix
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 -(g*(M+m2))/(M*l2) 0];


% B matrix
B=[0;
   1/M;
   0;
   1/(M*l1);
   0;
   1/(M*l2)];


% Choosing the values of R and Q which is used to calculate the LQR cost function
Q = diag([1, 1, 1, 1, 1, 1]);
```

```
R = 0.05;
[K_matrix] = lqr(A,B,Q,R);


% Input to the system which is the force
Force = -K_matrix*X;
X_dot = zeros(6,1);


% X_dot
X_dot(1) = X(2);


% X_Doubledot
X_dot(2)=(Force-(g/2)*(m1*sind(2*X(3))+m2*sind(2*X(5)))-
(m1*l1*(X(4)^2)*sind(X(3)))-
(m2*l2*(X(6)^2)*sind(X(5))))/(M+m1*((sind(X(3)))^2)+m2*((sind(X(5)))^2));


% Theta_1 dot
X_dot(3)= X(4);


% Theta_1 Doubledot;
X_dot(4)= (X_dot(2)*cosd(X(3))-g*(sind(X(3))))/l1';


% Theta_2 Dot
X_dot(5)= X(6);


% Theta_2 Doubledot;
X_dot(6)= (X_dot(2)*cosd(X(5))-g*(sind(X(5))))/l2;
end
```

Problem_E

```
% from the given output vectors in ques computing the observability of the
```

```
% system

% output vectors: x, theta1 theta2, x theta2, x theta1 theta2

% Computing the C matrix for the corresponding output vectors

% C matrix for output vector1
C_O1 = [1 0 0 0 0 0];

% C matrix for the output vector2
C_O2 = [0 0 1 0 0 0;
        0 0 0 0 1 0];

% C matrix for the output vector3
C_O3 = [1 0 0 0 0 0;
        0 0 0 0 1 0];

% C matrix for the output vector4
C_O4 = [1 0 0 0 0 0;
        0 0 1 0 0 0;
        0 0 0 0 1 0];

% computing the observability matrix for all the output vector conditions

O1 = [C_O1' A'*C_O1' A'*A'*C_O1' A'*A'*A'*C_O1' A'*A'*A'*A'*C_O1'
A'*A'*A'*A'*A'*C_O1']
O2 = [C_O2' A'*C_O2' A'*A'*C_O2' A'*A'*A'*C_O2' A'*A'*A'*A'*C_O2'
A'*A'*A'*A'*A'*C_O2']
O3 = [C_O3' A'*C_O3' A'*A'*C_O3' A'*A'*A'*C_O3' A'*A'*A'*A'*C_O3'
A'*A'*A'*A'*A'*C_O3']
O4 = [C_O4' A'*C_O4' A'*A'*C_O4' A'*A'*A'*C_O4' A'*A'*A'*A'*C_O4'
A'*A'*A'*A'*A'*C_O4']

% Computing the rank of the observable matrix for all the output vector
% conditions
disp(rank(O1))
disp("Since the rank of the observablity matrix is 6, which is full rank, hence
the system is observable for x(t) ")
disp(rank(O2))
```

```
disp("Since the rank of the matrix is 4, not fullrank, hence the system is not
observable for theta(1) theta(2)")
disp(rank(O3))
disp("Since the rank of the observablity matrix is 6, which is full rank, hence
the system is observable for x(t) theta(2) ")
disp(rank(O4))
disp("Since the rank of the observablity matrix is 6, which is full rank, hence
the system is observable for x(t) theta(1) theta(2) ")
```

Prob_F

```
% considering 12 state variables for the system
% The first 6 are actual variables and the other 6 are estimated variables
% which are used to compute the error between the estimated state and true
% state.


% State variables  = [x x_dot theta1 theta1_dot theta2 theta2_dot
Estimated_variables]


% Initial conditions for luenberger observer
X_i = [0, 0, 10, 0, 15, 0, 0, 0, 0, 0, 0, 0];


Poles_placements = [-0.5; -1; -2; -2.5; -3.5; -4.5]


% computing the gain matrix using the lqr controller function
K_matrix =lqr(A,B,Q,R);


% computing the luenberger matrix for all the output vector conditions
% which are observable
L1 = place(A',C_01',Poles_placements)'
L3 = place(A',C_03',Poles_placements)'
L4 = place(A',C_04',Poles_placements)'


% for output vectors x(t)
% Luenberger A matrix
Ac_1 = [(A-(B*K_matrix)) (B*K_matrix);
        zeros(size(A)) (A-(L1*C_01))];
% Luenberger B matrix
Bc = [B;zeros(size(B))];
% Luenberger C matrix
Cc_1 = [C_01 zeros(size(C_01))];
```

```
% for output vectors x(t) theta(2)
Ac_3 = [(A-B*K_matrix) B*K_matrix;
        zeros(size(A)) (A-L3*C_03)];
% Luenberger C matrix
Cc_3 = [C_03 zeros(size(C_03))];


% for output vectors x(t) theta(1) theta(2)
% Luenberger A matrix
Ac_4 = [(A-B*K_matrix) B*K_matrix;
        zeros(size(A)) (A-L4*C_04)];
% Luenberger C matrix
Cc_4 = [C_04 zeros(size(C_04))];


% Initial and step response for linearsystem1
disp("Linear System 1")
linearSystem_1 = ss(Ac_1, Bc, Cc_1,D);
figure
initial(linearSystem_1,X_i)
figure
step(linearSystem_1)


% Initial and step response for linearsystem3
disp("Linear System 3")
linearSystem_3 = ss(Ac_3, Bc, Cc_3,D);
figure
initial(linearSystem_3,X_i)
figure
step(linearSystem_3)


% Initial and step response for linearsystem4
disp("Linear System 4")
linearSystem_4 = ss(Ac_4, Bc, Cc_4, D);
figure
initial(linearSystem_4,X_i)
figure
step(linearSystem_4)


% Non-linear Sysytems
disp("Non linear systems")


% A matrix for the non linear system
```

```
A_nl=[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 -(g*(M+m2))/(M*l2) 0];
% B matrix for the non linear system
B_nl = [0;
        1/M;
        0;
        1/(M*l1);
        0;
        1/(M*l2)];


% Considering the systems with outputvectors which are observable


% Computing the gain matrix using the lqr function for the non-linear
% system
K_matrix =lqr(A_nl,B_nl,Q,R);


% computing the luenberger matrix for all the output vector conditions
% which are observable for non-linear systems
L1 = place(A_nl',C_01',Poles_placements)'
L3 = place(A_nl',C_03',Poles_placements)'
L4 = place(A_nl',C_04',Poles_placements)'


% for output vectors x(t)
% Luenberger A matrix
Ac_1 = [(A_nl-B_nl*K_matrix) B_nl*K_matrix;
        zeros(size(A_nl)) (A_nl-L1*C_01)];
% Luenberger B matrix
Bc = [B_nl;zeros(size(B_nl))];
% Luenberger C matrix
Cc_1 = [C_01 zeros(size(C_01))];


% for output vectors x(t) theta(2)
% Luenberger A matrix
Ac_3 = [(A_nl-B_nl*K_matrix) B_nl*K_matrix;
        zeros(size(A_nl)) (A_nl-L3*C_03)];
% Luenberger C matrix
Cc_3 = [C_03 zeros(size(C_03))];
```

```
% for output vectors x(t) theta(1) theta(2)
% Luenberger A matrix
Ac_4 = [(A_nl-B_nl*K_matrix) B_nl*K_matrix;
        zeros(size(A_nl)) (A_nl-L4*C_04)];
% Luenberger C matrix
Cc_4 = [C_04 zeros(size(C_04))];


% Initial and step response for non-linearsystem1
nonLinearSystem_1 = ss(Ac_1, Bc, Cc_1,D);
figure
initial(nonLinearSystem_1,X_i)
figure
step(nonLinearSystem_1)


% Initial and step response for non-linearsystem3
nonLinearSystem_3 = ss(Ac_3, Bc, Cc_3,D);
figure
initial(nonLinearSystem_3,X_i)
figure
step(nonLinearSystem_3)


% Initial and step response for non-linearsystem4
nonLinearSystem_4 = ss(Ac_4, Bc, Cc_4, D);
figure
initial(nonLinearSystem_4,X_i)
figure
step(nonLinearSystem_4)
```

Problem_G

```
% LQG Method


% considering 12 state variables for the system
% The first 6 are actual variables and the other 6 are estimated variables
% which are used to compute the error between the estimated state and true
% state.


% State variables  = [x x_dot theta1 theta1_dot theta2 theta2_dot
Estimated_variables]


% Initial conditions
X_i = [0, 0, 10, 0, 15, 0, 0, 0, 0, 0, 0, 0];
```

```
% computing the gain matrix using the lqr controller function
K_matrix =lqr(A,B,Q,R);


% Process noise term
U_d = diag([0.6, 0.6, 0.6, 0.6, 0.6, 0.6]);
% Measurement noise term
V_t = 1;


% choosing the desired smallest output vector out of the observable output
% vectors
% Gain matrix for Kalman Filter for output vector x(t)
K_filter = lqr(A', C_01', U_d, V_t)';


% Initial and step response for the linearsystem_1
disp("Linear System 1")
linearSystem_1 = ss([(A-B*K_matrix) B*K_matrix; zeros(size(A)) (A-
K_filter*C_01)], [B;zeros(size(B))],[C_01 zeros(size(C_01))], D);
figure
initial(linearSystem_1,X_i)
figure
step(linearSystem_1)




% non-linear system control using LQG
disp("LQG non-linear control")
X_i = [0, 0, 10, 0, 15, 0, 0, 0, 0, 0, 0, 0];
Timespan = 0:0.01:50;
[T1,X1] = ode45(@LQG_controller,Timespan,X_i);
plot(T1,X1)
```

```
function X_dot = LQG_controller(T1,X)
% Given Initial conditions
% Mass of the Crane, mass of load1, mass of load2, Cable length 1,
% Cable length 2, gravity
M= 1000;
m1= 100;
m2= 100;
l1= 20;
l2= 10;
```

```
g= 9.81;


% A matrix
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 -(g*(M+m2))/(M*l2) 0];


% B matrix
B=[0; 1/M; 0; 1/(M*l1); 0; 1/(M*l2)];


% Choosing the values for Q and R for estimating the cost function
Q = diag([1, 1, 1, 1, 1, 1]);
R = 0.5;


% Choosing the smallest output vector from the observable output vector
C_01 = [1 0 0 0 0 0];


% D
D = 0;


[K_matrix] = lqr(A,B,Q,R);
F = - K_matrix*X(1:6);


% Process noise term
U_d = diag([0.6, 0.6, 0.6, 0.6, 0.6, 0.6]);
% Measurement noise term
V_t=1;
% Gain matrix for Kalman Filter for C1
K_filter =lqr(A', C_01', U_d, V_t)';
% Standard Deviation considering the output vector x(t)
Standard_Deviation = (A-K_filter*C_01)*X(7:12);
X_dot = zeros(12,1);


% X(1) = x
X_dot(1) = X(2); % X_Dot


% X(2) = x_dot
```

```
X_dot(2) = (F-(g/2)*(m1*sind(2*X(3))+m2*sind(2*X(5)))-
(m1*l1*(X(4)^2)*sind(X(3)))-
(m2*l2*(X(6)^2)*sind(X(5))))/(M+m1*((sind(X(3)))^2)+m2*((sind(X(5)))^2)); %
X_Doubledot


% X(3) = Theta_1
X_dot(3) = X(4); % Theta_1 dot


% X(4) = Theta_1 dot
X_dot(4) = (X_dot(2)*cosd(X(3))-g*(sind(X(3))))/l1'; % Theta_1 Doubledot;


% X(5)= Theta_2
X_dot(5) = X(6); % Theta_2 Dot


% X(6)= Theta_2 dot;
X_dot(6) = (X_dot(2)*cosd(X(5))-g*(sind(X(5))))/l2; % Theta_2 Doubledot;


% X(7), X(8), X(9), X(10), X(11), X(12)
X_dot(7) = X(2)-X(10);
X_dot(8) = X_dot(2)-Standard_Deviation(2);
X_dot(9) = X(4)-X(11);
X_dot(10) = X_dot(4)-Standard_Deviation(4);
X_dot(11) = X(6)-X(12);
X_dot(12) = X_dot(6)-Standard_Deviation(6);
end
```