Project 2

University of Maryland

December 2022

Instructors:
Waseem Ansar Malik

Students:
Sai Teja Gilukara(119369623)

Rohit Reddy Pakala(119399125)

Semester:
Fall 2022

Course code:
ENPM667

Contents

1 Introduction

- The project aims to build LQR and LQG controllers for a crane that has two loads suspended from it by cables connected to the crane's base. We know that these loads have different masses (m1, m2) and the cables have length l1 and l2
- In this project, we will use the Lagrangian approach to find the system's equations of motion. Then obtain a non-linear state space representation of the system. The system would then be linearized around an equilibrium point, and its state space representation would be generated.
- Then, depending on M,m_1, m_2, l_1, l_2, we proceed to determine the controllability conditions.
- After the controllability conditions are obtained we will be equipped with all that is needed to design an LQR controller for the crane and load system. Before we start with designing the LQR controller, the system needs to be checked whether it is controllable or not, in this case the system is controllable, then the LQR controller is designed. The simulation responses are recorded for two scenarios by applying the LQR controller to the original non linear system and the linearized system. We simulate the responses by adjusting the LQR parameters until we get the suitable response and then we perform Lyapunov analysis of this closed loop system to certify the stability.
- Now after designing an LQR controller for the system, we consider the parameters that we used to obtain controllability conditions and determine if the system observability for some output vectors.
- The problem statement for the project continues to ask us to determine the best Luenberger Observer for each of the output vectors of they are observable and then simulate the response to input conditions and unit step input when applied to the linearized and non linearized system.
- The final step is to design an output feedback controller for the smallest output vector using the LQG method and illustrate its performance in form of a simulation.

2 Equations of motion

- Position for mass m_1 as a function of θ_1

$$x_{m_1} = (x - l_1 \sin(\theta_1))\,\hat{i} + (-l_1 \cos(\theta_1))\,\hat{j} \tag{1}$$

- Differentiate above equation w.r.t to time for velocity:

$$v_{m_1} = \left(\dot{x} - l_1 \cos(\theta_1)\,\dot{\theta_1}\right)\hat{i} + (l_1 \sin(\theta_1)\,\dot{\theta_1})\hat{j} \tag{2}$$

- Position of mass m_2 as a function of θ_2 :

$$x_{m_2} = (x - l_2 \sin(\theta_2))\,\hat{i} + (-l_2 \cos(\theta_2))\,\hat{j} \tag{3}$$

* * *

- Differentiate above equation w.r.t to time for velocity:

$$v_{m_2} = \left(\dot{x} - l_2 \cos\left(\theta_2\right) \dot{\theta_2}\right) \hat{i} + \left(l_2 \sin\left(\theta_2\right) \dot{\theta_2}\right) \hat{j} \tag{4}$$

- Using the above equation (2) and (4), Kinetic energy can be derived as:

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

- Potential energy of the system, by using (1) and (3) is given by:

$$P = -m_1 g l_1 \cos\left(\theta_1\right) - m_2 g l_2 \cos\left(\theta_2\right) = -g \left[m_1 l_1 \cos\left(\theta_1\right) + m_2 g l_2 \cos\left(\theta_2\right)\right. \tag{6}$$

- w.k.t the Lagrange equation is the difference between Kinetic and Potential energies:

$$L = K - P \tag{7}$$

$$\begin{gathered} L = K \cdot E - P \cdot E \\ \text{L} = \frac{1}{2} M \dot{x}^2 + \frac{1}{2} m_1 \dot{x}^2 + \frac{1}{2} m_1 l_1^2 \dot{\theta_1}^2 \cos^2\left(\theta_1\right) - m_1 l_1 \dot{\theta}_1 \dot{x} \cos\left(\theta_1\right) + \frac{1}{2} m_1 l_1^2 \dot{\theta_1}^2 \sin^2\left(\theta_1\right) \\ + \frac{1}{2} m_2 \dot{x}^2 + \frac{1}{2} m_2 l_2^2 \dot{\theta}_2^2 \cos^2\left(\theta_2\right) - m_2 l_2 \dot{\theta}_2 \dot{x} \cos\left(\theta_2\right) + \frac{1}{2} m_2 l_2^2 \dot{\theta}_2^2 \sin^2\left(\theta_1\right) + g\left[m_1 l_1 \cos\left(\theta_1\right) + \right. \\ \left. m_2 l_2 \cos\left(\theta_2\right)\right] \end{gathered} \tag{8}$$

- On simplifying we get L:

$$\begin{gathered} \text{L} = \frac{1}{2} M \dot{x}^2 + \frac{1}{2}\left(m_1 + m_2\right) \dot{x}^2 + \frac{1}{2} m_1 l_1^2 \dot{\theta_1}^2 + \frac{1}{2} m_2 l_2^2 \dot{\theta_2}^2 - \\ \dot{x}(m_1 l_1 \dot{\theta}_1 \cos\left(\theta_1\right) + m_2 l_2 \dot{\theta}_2 \cos(\theta_2)) + g[m_1 l_1 \cos(\theta_1) + m_2 l_2 \cos(\theta_2)] \end{gathered} \tag{9}$$

- Lyapunov Equations related to the state variables considered for the system are defined as follows:

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

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_1}\right) - \left(\frac{\partial L}{\partial \theta_1}\right) = 0 \tag{11}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_2}\right) - \left(\frac{\partial L}{\partial \theta_2}\right) = 0 \tag{12}$$

* * *

- The computation of the above stated equations results in these following relations - Relation 1:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) - \left(\frac{\partial L}{\partial x}\right) = F \tag{13}$$

$$\frac{\partial L}{\partial \dot{x}} = M\dot{x} + (m_1 + m_2)\dot{x} - m_1 l_1 \dot{\theta}_1 \cos(\theta_1) - m_2 l_2 \dot{\theta}_2 \cos(\theta_2) \tag{14}$$

$$\begin{gathered}\frac{d}{dt}(\frac{\partial L}{\partial \dot{x}}) = M\ddot{x} + (m_1 + m_2)\ddot{x} - [m_1 l_1 \ddot{\theta}_1 \cos(\theta_1) \\ -m_1 l_1 \dot{\theta}_1^2 \sin(\theta_1)] - [m_2 l_2 \ddot{\theta}_2 \cos(\theta_2) - m_2 l_2 \dot{\theta}_2^2 \sin(\theta_2)]\end{gathered} \tag{15}$$

- here:

$$\frac{\partial L}{\partial x} = 0 \tag{16}$$

- On solving, we get force equation as:

$$[M + m_1 + m_2]\,\ddot{x} - m_1 l_1 \ddot{\theta}_1 \cos(\theta_1) + m_1 l_1 \dot{\theta_1}^2 \sin(\theta 1) - m_2 l_2 \ddot{\theta}_2 \cos(\theta_2) + m_2 l_2 \dot{\theta_2}^2 \sin(\theta_2) = F \tag{17}$$

- Now, we know that:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_1}\right) - \left(\frac{\partial L}{\partial \theta_1}\right) = 0 \tag{18}$$

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

- Differentiating w.r.t time:

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

- Also here:

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

- Now, combining equations (20) and (21):

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

- By cancelling out the equivalent terms, we get the following equation, Which is the second Lagrange Equation:

$$m_1 l_1^2 \ddot{\theta}_1 - m_1 \ddot{x} l_1 \cos(\theta_1) + m_1 l_1 g \sin(\theta_1) = 0 \tag{23}$$

- For third equation from the Lagrange equation:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_2}\right)-\left(\frac{\partial L}{\partial \theta_2}\right)=0 \tag{24}$$

$$\frac{\partial L}{\partial \dot{\theta}_2}=m_2 l_2^2 \dot{\theta}_2-m_2 \dot{x} l_2 \cos \left(\theta_2\right) \tag{25}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_2}\right)=m_2 l_2^2 \ddot{\theta}_2-\left[m_2 \ddot{x} l_2 \cos \left(\theta_2\right)-m_2 \dot{\theta}_2 \dot{x} l_2 \sin \left(\theta_2\right)\right] \tag{26}$$

$$\left(\frac{\partial L}{\partial \theta_2}\right)=m_2 \dot{x} l_2 \dot{\theta}_2 \sin \left(\theta_2\right)-m_2 l_2 g \sin \left(\theta_2\right) \tag{27}$$

- Hence we write :

$$m_2 l_2^2 \ddot{\theta}_2-m_2 l_2 \ddot{x} \cos \left(\theta_2\right)+m_2 \dot{\theta}_2 \dot{x} l_2 \sin \left(\theta_2\right)-m_2 \dot{\theta}_2 \dot{x} l_2 \sin \left(\theta_2\right)+m_2 g l_2 \sin \left(\theta_2\right) \tag{28}$$

- After simplification, we get:

$$m_2 l_2^2 \ddot{\theta}_2-m_2 l_2 \ddot{x} \cos \left(\theta_2\right)+m_2 g l_2 \sin \left(\theta_2\right)=0 \tag{29}$$

- We first write the equations for the double differentiation components of some of our state variables, as inferred from the equations above, before linearizing about the specified equilibrium points:

$$\ddot{x}=\tfrac{1}{M+m_1+m_2}\left[m_1 l_1 \ddot{\theta}_1 \cos \theta_1+m_2 l_2 \ddot{\theta}_2 \cos \theta_2-m_1 l_1 \dot{\theta}_1^2 \sin \theta_1-m_2 l_2 \dot{\theta}_2^2 \sin \theta_2+F\right] \tag{30}$$

$$\ddot{\theta}_1=\frac{\ddot{x} \cos \theta_1}{l_1}-\frac{g \sin \theta_1}{l_1} \tag{31}$$

$$\ddot{\theta}_2=\frac{\ddot{x} \cos \theta_2}{l_2}-\frac{g \sin \theta_2}{l_2} \tag{32}$$

2.1 Non linear state space representation

- Considering the system states, we can write the state space form of the non linear system as follows

$$\begin{bmatrix}\dot{x}\\ \ddot{x}\\ \dot{\theta}_1\\ \ddot{\theta}_1\\ \dot{\theta}_2\\ \ddot{\theta}_2\end{bmatrix}=\begin{bmatrix}\dot{x}\\ \frac{-m_1 g \sin \theta_1 \cos \theta_2-m_2 g \sin \theta_2 \cos \theta_2-m_1 l_1 \dot{\theta}_1^2 \sin \theta_1-m_2 l_2 \dot{\theta}_2^2 \sin \theta_2+F}{M+m_1+m_2-m_1 \cos^2 \theta_1-m_2 \cos^2 \theta_2}\\ \dot{\theta}_1\\ \frac{-m_1 g \sin \theta_1 \cos \theta_2-m_2 g \sin \theta_2 \cos \theta_2-m_1 l_1 \dot{\theta}_1^2 \sin \theta_1-m_2 l_2 \dot{\theta}_2^2 \sin \theta_2+F}{(M+m_1+m_2-m_1 \cos^2 \theta_1-m_2 \cos^2 \theta_2) l_1}-\frac{g \sin \theta_1}{l_1}\\ \dot{\theta}_2\\ \frac{-m_1 g \sin \theta_1 \cos \theta_2-m_2 g \sin \theta_2 \cos \theta_2-m_1 l_1 \dot{\theta}_1^2 \sin \theta_1-m_2 l_2 \dot{\theta}_2^2 \sin \theta_2+F}{(M+m_1+m_2-m_1 \cos^2 \theta_1-m_2 \cos^2 \theta_2) l_2}-\frac{g \sin \theta_2}{l_2}\end{bmatrix} \tag{33}$$

3 Linear State Space Representation of System

Linearizing a system around an equilibrium point involves finding a linear approximation of the system near the equilibrium point. This involves finding the Jacobian matrix of the system at the equilibrium point, which describes how the system's variables change in response to small perturbations.

To linearize a system around an equilibrium point, you can follow these steps:

- Identify the equilibrium point of the system. This is a point at which the system is in a state of balance, where the forces acting on the system are in equilibrium.
- Find the Jacobian matrix of the system at the equilibrium point. The Jacobian matrix is a matrix of partial derivatives that describes how the system's variables change in response to small perturbations.
- Linearize the system by replacing the nonlinear terms in the system's equations with their linear approximations. This can be done by using the Taylor series expansion to approximate the nonlinear terms.
- Solve the linearized system to find the response of the system to small perturbations.

Linearizing a system around an equilibrium point can be useful for analyzing the stability of the system, as well as for designing controllers to stabilize the system. It can also be used to simplify the analysis and design of complex systems by reducing the number of variables and equations that need to be considered.

- As discussed above, the derived equation of motion of the cart system with two pendulum and its represented state space form. We can say that, due to the sin and cos components, the equations is non linear and it is difficult to solve non linear equations.
- To approach this linearization the system around equilibrium point $x = 0, \theta_1 = 0$ and $\theta_2 = 0$ can be done.
  The limiting condition at equilibrium, ,

$$\begin{gathered}
\sin\theta_1 \approx \theta_1 \\
\sin\theta_2 \approx \theta_2 \\
\cos\theta_1 \approx 1 \\
\cos\theta_2 \approx 1 \\
\dot{\theta}_1^2 = \dot{\theta}_2^2 \approx 0
\end{gathered}$$

- With these approximations, we arrive at these equations:

$$\begin{gathered}
\ddot{x} = \frac{1}{M+m_1+m_2}\left[m_1 l_1 \ddot{\theta}_1 \cos\theta_1 + m_2 l_2 \ddot{\theta}_2 \cos\theta_2 - m_1 l_1 \dot{\theta}_1^2 \sin\theta_1 - m_2 l_2 \dot{\theta}_2^2 \sin\theta_2 + F\right] \\
\ddot{\theta}_1 = \frac{\ddot{x}\cos\theta_1}{l_1} - \frac{g\sin\theta_1}{l_1} \\
\ddot{\theta}_2 = \frac{\ddot{x}\cos\theta_2}{l_2} - \frac{g\sin\theta_2}{l_2}
\end{gathered} \tag{34}$$

- The system of equations represented in Eqn 34 can be presented in state space form. Taking $x, \dot{x}, \theta_1, \dot{\theta}_1, \theta_2$ and $\dot{\theta}_2$ as state variables, as shown below,

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta}_1 \\ \ddot{\theta}_1 \\ \dot{\theta}_2 \\ \ddot{\theta}_2 \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \frac{-m_1 g\sin\theta_1\cos\theta_2 - m_2 g\sin\theta_2\cos\theta_2 - m_1 l_1\dot{\theta}_1^2\sin\theta_1 - m_2 l_2\dot{\theta}_2^2\sin\theta_2 + F}{M+m_1+m_2-m_1\cos^2\theta_1 - m_2\cos^2\theta_2} \\ \dot{\theta}_1 \\ \frac{-m_1 g\sin\theta_1\cos\theta_2 - m_2 g\sin\theta_2\cos\theta_2 - m_1 l_1\dot{\theta}_1^2\sin\theta_1 - m_2 l_2\dot{\theta}_2^2\sin\theta_2 + F}{(M+m_1+m_2-m_1\cos^2\theta_1 - m_2\cos^2\theta_2)l_1} - \frac{g\sin\theta_1}{l_1} \\ \frac{-m_1 g\sin\theta_1\cos\theta_2 - m_2 g\sin\theta_2\cos\theta_2 - m_1 l_1\dot{\theta}_1^2\sin\theta_1 - m_2 l_2\dot{\theta}_2^2\sin\theta_2 + F}{(M+m_1+m_2-m_1\cos^2\theta_1 - m_2\cos^2\theta_2)l_2} - \frac{g\sin\theta_2}{l_2} \end{bmatrix} \tag{35}$$

$$y = CX + DU \tag{36}$$

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

- This is same as $\dot{X} = AX + BU$. Input force on cart F is represented as U.

4 Obtaining controllability conditions

For Linear Time-Varying Systems

- Rank (W(A,B)) = Full Rank where W(A,B) is Grammian matrix
- The Full Rank is = order of the matrix (n).

$$R = \begin{bmatrix} B & AB & A^2B & A^3B & A^4B & A^5B \end{bmatrix}$$

```
syms m1 m2 M g l1 l2 F
%m1, m2 are masses and l1, l2 are lengths respectively
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*g)+(m1*g))/(M*l1) 0 -g*m2/(M*l1) 0;
    0 0 0 0 0 1;
    0 0 -m1*g/(M*l2) 0 -((M*g)+(m2*g))/(M*l2) 0];
disp(A)
```

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{g\,m_1}{M} & 0 & -\frac{g\,m_2}{M} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{M\,g+g\,m_1}{M\,l_1} & 0 & -\frac{g\,m_2}{M\,l_1} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{g\,m_1}{M\,l_2} & 0 & -\frac{M\,g+g\,m_2}{M\,l_2} & 0 \end{pmatrix}$$

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

$$\begin{pmatrix} 0 \\ \frac{1}{M} \\ 0 \\ \frac{1}{M\,l_1} \\ 0 \\ \frac{1}{M\,l_2} \end{pmatrix}$$

```
%controlability matrix is given by
Ct = [B A*B (A^2)*B (A^3)*B (A^4)*B (A^5)*B]
```

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

where

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

$$\sigma_2 = \frac{\frac{{\sigma_7}^2}{M^2\,{l_2}^2} + \frac{g^2\,m_1\,m_2}{\sigma_8}}{M\,l_2} + \frac{\frac{g\,m_1\,\sigma_7}{M^2\,{l_2}^2} + \frac{g\,m_1\,\sigma_9}{\sigma_8}}{M\,l_1}$$

$$\sigma_3 = \frac{\frac{{\sigma_9}^2}{M^2\,{l_1}^2} + \frac{g^2\,m_1\,m_2}{\sigma_8}}{M\,l_1} + \frac{\frac{g\,m_2\,\sigma_9}{M^2\,{l_1}^2} + \frac{g\,m_2\,\sigma_7}{\sigma_8}}{M\,l_2}$$

$$\sigma_4 = \frac{\frac{g\,m_1\,\sigma_9}{M^2\,l_1} + \frac{g^2\,m_1\,m_2}{M^2\,l_2}}{M\,l_1} + \frac{\frac{g\,m_2\,\sigma_7}{M^2\,l_2} + \frac{g^2\,m_1\,m_2}{M^2\,l_1}}{M\,l_2}$$

$$\sigma_5 = -\frac{\sigma_7}{M^2\,{l_2}^2} - \frac{g\,m_1}{\sigma_8}$$

$$\sigma_6 = -\frac{\sigma_9}{M^2\,{l_1}^2} - \frac{g\,m_2}{\sigma_8}$$

$$\sigma_7 = M\,g + g\,m_2$$

$$\sigma_8 = M^2\,l_1\,l_2$$

$$\sigma_9 = M\,g + g\,m_1$$

```
Rank = rank(Ct)
Detetminant = det(Ct)
cc=simplify(det(Ct))
```

Determinant = $-\frac{g^6\,{l_1}^2 - 2\,g^6\,l_1\,l_2 + g^6\,{l_2}^2}{M^6\,{l_1}^6\,{l_2}^6}$

cc = $-\frac{g^6\,(l_1 - l_2)^2}{M^6\,{l_1}^6\,{l_2}^6}$

```
%The system is controllable only if the controllability matrix is full rank.
%Rank is 6 in this case, so system is controllable
disp("Rank"); rank(Cability)
```

Rank
ans = 6

```
%As the result is zero the system is not controllable / Uncontrollable
%So getting a condition for controlability
Ct1 = subs(Ct,l1,l2) %substituting l1=l2
```

$$
\text{Ct1} = \begin{pmatrix} 0 & \frac{1}{M} & 0 & \sigma_1 & 0 & \sigma_4 \\ \frac{1}{M} & 0 & \sigma_1 & 0 & \sigma_4 & 0 \\ 0 & \frac{1}{M\,l_2} & 0 & \sigma_5 & 0 & \sigma_2 \\ \frac{1}{M\,l_2} & 0 & \sigma_5 & 0 & \sigma_2 & 0 \\ 0 & \frac{1}{M\,l_2} & 0 & \sigma_6 & 0 & \sigma_3 \\ \frac{1}{M\,l_2} & 0 & \sigma_6 & 0 & \sigma_3 & 0 \end{pmatrix}
$$

where

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

$$\sigma_2 = \frac{\frac{{\sigma_9}^2}{\sigma_{11}} + \sigma_7}{M\,l_2} + \frac{\frac{g\,m_2\,\sigma_9}{\sigma_{11}} + \frac{g\,m_2\,\sigma_{10}}{\sigma_{11}}}{M\,l_2}$$

$$\sigma_3 = \frac{\frac{{\sigma_{10}}^2}{\sigma_{11}} + \sigma_7}{M\,l_2} + \frac{\frac{g\,m_1\,\sigma_9}{\sigma_{11}} + \frac{g\,m_1\,\sigma_{10}}{\sigma_{11}}}{M\,l_2}$$

$$\sigma_4 = \frac{\frac{g\,m_1\,\sigma_9}{M^2\,l_2} + \sigma_8}{M\,l_2} + \frac{\frac{g\,m_2\,\sigma_{10}}{M^2\,l_2} + \sigma_8}{M\,l_2}$$

$$\sigma_5 = -\frac{\sigma_9}{\sigma_{11}} - \frac{g\,m_2}{\sigma_{11}}$$

$$\sigma_6 = -\frac{\sigma_{10}}{\sigma_{11}} - \frac{g\,m_1}{\sigma_{11}}$$

$$\sigma_7 = \frac{g^2\,m_1\,m_2}{\sigma_{11}}$$

$$\sigma_8 = \frac{g^2\,m_1\,m_2}{M^2\,l_2}$$

$$\sigma_9 = M\,g + g\,m_1$$

$$\sigma_{10} = M\,g + g\,m_2$$

$$\sigma_{11} = M^2\,{l_2}^2$$

```
disp('the system is uncontrollable when l1=l2 as det is zero')
```

```
the system is uncontrollable when l1=l2 as det is zero
```

```
disp("the system is controllable when l1 != l2, l1 !=0, and l2 !=0 ")
```

```
the system is controllable when l1 != l2, l1 !=0, and l2 !=0
```

5 LQR controller design

- For the LQR Controller, if A,B_k is stabilizable, and minimize k that minimizes the cost:

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

5.1 Check for controllability

D. LQR Controller

```
As = double(subs(A, {g, l1, l2, m1, m2, M}, {9.8, 20, 10, 100, 100, 1000}))
```

```
As = 6x6
         0    1.0000         0         0         0         0
         0         0   -0.9800         0   -0.9800         0
         0         0         0    1.0000         0         0
         0         0   -0.5390         0   -0.0490         0
         0         0         0         0         0    1.0000
         0         0   -0.0980         0   -1.0780         0
```

```
Bs = double(subs(B, {g, l1, l2, m1, m2, M}, {9.8, 20, 10, 100, 100, 1000}))
```

```
Bs = 6x1
1.0e-03 *

         0
    1.0000
         0
    0.0500
         0
    0.1000
```

```
cont = ctrb(As,Bs)
```

```
cont = 6x6
1.0e-03 *

         0    1.0000         0   -0.1470         0    0.1417
    1.0000         0   -0.1470         0    0.1417         0
         0    0.0500         0   -0.0319         0    0.0227
    0.0500         0   -0.0319         0    0.0227         0
         0    0.1000         0   -0.1127         0    0.1246
    0.1000         0   -0.1127         0    0.1246         0
```

```
if (rank(cont)==6)
    disp("Rank of cont is quals to A Hence the system is controllable")
else
    disp("Rank of cont is not equal to A, system is uncontrollable")
end
```

```
Rank of cont is quals to A Hence the system is controllable
```

```
% Assumin the Q matrix
Q =[5 0 0 0 0 0;
    0 0 0 0 0 0 ;
    0 0 5000 0 0 0;
    0 0 0 0 0 0 ;
    0 0 0 0 5000 0;
    0 0 0 0 0 0]
```

```
Q = 6x6
           5           0           0           0           0           0
           0           0           0           0           0           0
           0           0        5000           0           0           0
           0           0           0           0           0           0
           0           0           0           0        5000           0
           0           0           0           0           0           0
```

```
R=0.01
```

```
R = 0.0100
```

```
C = eye(6)
```

```
C = 6x6
     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
```

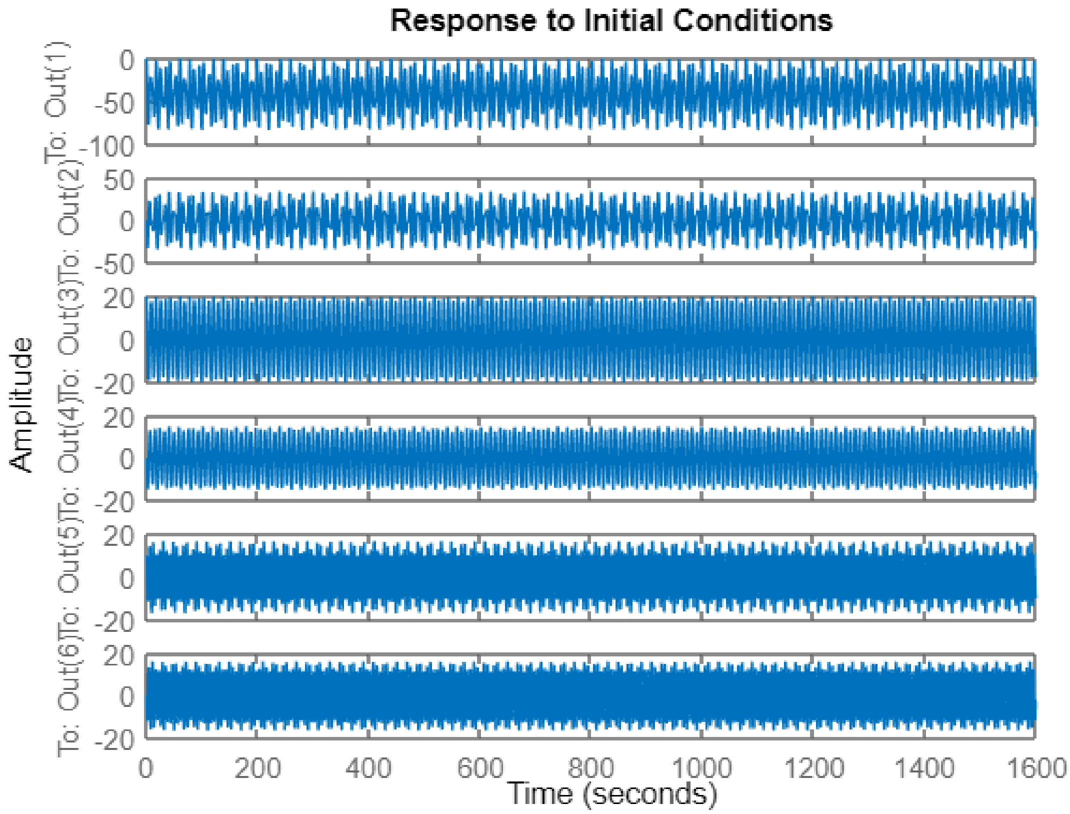
```
D = 0;
%initial state is given below
initial_state = [0;0;20;0;10;0]
```

```
initial_state = 6x1
     0
     0
    20
     0
    10
     0
```

```
%ss is the MATLAB function to find the state space representation of the
%system
sys_1 = ss(As, Bs, C, D)
```

```
sys_1 =

  A =
             x1      x2      x3      x4      x5      x6
   x1         0       1       0       0       0       0
   x2         0       0   -0.98       0   -0.98       0
   x3         0       0       0       1       0       0
   x4         0       0  -0.539       0  -0.049       0
   x5         0       0       0       0       0       1
   x6         0       0  -0.098       0  -1.078       0

  B =
             u1
   x1         0
   x2     0.001
   x3         0
   x4     5e-05
   x5         0
   x6    0.0001

  C =
       x1  x2  x3  x4  x5  x6
   y1   1   0   0   0   0   0
   y2   0   1   0   0   0   0
   y3   0   0   1   0   0   0
   y4   0   0   0   1   0   0
   y5   0   0   0   0   1   0
   y6   0   0   0   0   0   1

  D =
       u1
   y1   0
   y2   0
   y3   0
   y4   0
   y5   0
   y6   0

Continuous-time state-space model.
```

```
figure
initial(sys_1,initial_state)
```




```
%lqr is in-buit LQR controller function
[K_matrix_gain, S_matrix, P] = lqr(As, Bs, Q, R);
K_matrix_gain, S_matrix, P
```

```
K_matrix_gain = 1x6
   22.3607  234.6518  131.1793  678.6548  174.6940  360.1045

S_matrix = 6x6
1.0e+05 *

    0.0005    0.0028    0.0015   -0.0051    0.0008   -0.0026
    0.0028    0.0287    0.0210   -0.0501    0.0111   -0.0268
    0.0015    0.0210    1.3083    0.0230    0.0017   -0.0903
   -0.0051   -0.0501    0.0230    2.5161    0.1147   -0.0783
    0.0008    0.0111    0.0017    0.1147    0.6957    0.0065
   -0.0026   -0.0268   -0.0903   -0.0783    0.0065    0.6675

P = 6x1 complex
  -0.0197 + 0.7282i
  -0.0197 - 0.7282i
  -0.0364 + 1.0432i
  -0.0364 - 1.0432i
```

```
  -0.0961 + 0.0966i
  -0.0961 - 0.0966i
```

```
%eigen values
eig(S_matrix)
```

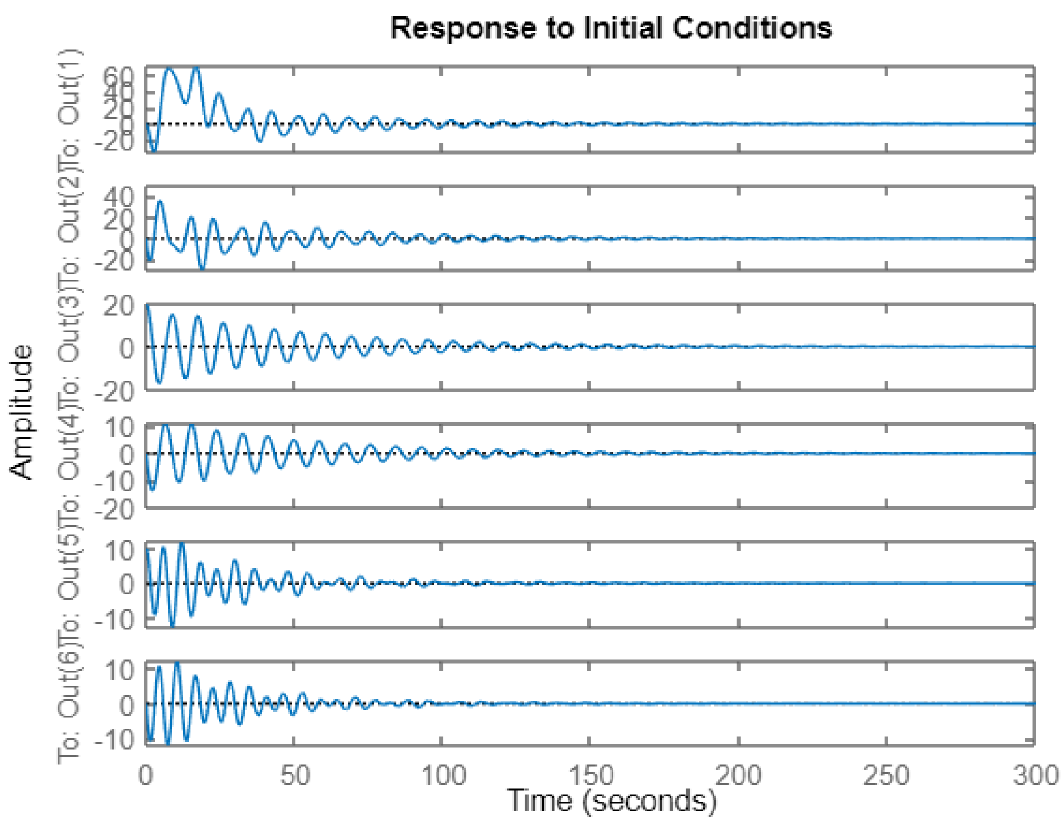
```
ans = 6x1
1.0e+05 *

    0.0003
    0.0261
    0.6502
    0.6918
    1.3204
    2.5281
```

```
sys_2 = ss(As-(Bs*K_matrix_gain),Bs,C,D)
```

```
sys_2 =

  A =
              x1          x2          x3          x4          x5          x6
   x1          0           1           0           0           0           0
   x2   -0.02236     -0.2347      -1.111     -0.6787      -1.155     -0.3601
   x3          0           0           0           1           0           0
   x4  -0.001118    -0.01173     -0.5456    -0.03393    -0.05773    -0.01801
   x5          0           0           0           0           0           1
   x6  -0.002236    -0.02347     -0.1111    -0.06787      -1.095    -0.03601

  B =
           u1
   x1       0
   x2   0.001
   x3       0
   x4   5e-05
   x5       0
   x6  0.0001

  C =
       x1  x2  x3  x4  x5  x6
   y1   1   0   0   0   0   0
   y2   0   1   0   0   0   0
   y3   0   0   1   0   0   0
   y4   0   0   0   1   0   0
   y5   0   0   0   0   1   0
   y6   0   0   0   0   0   1

  D =
```

```
        u1
   y1    0
   y2    0
   y3    0
   y4    0
   y5    0
   y6    0

Continuous-time state-space model.
```

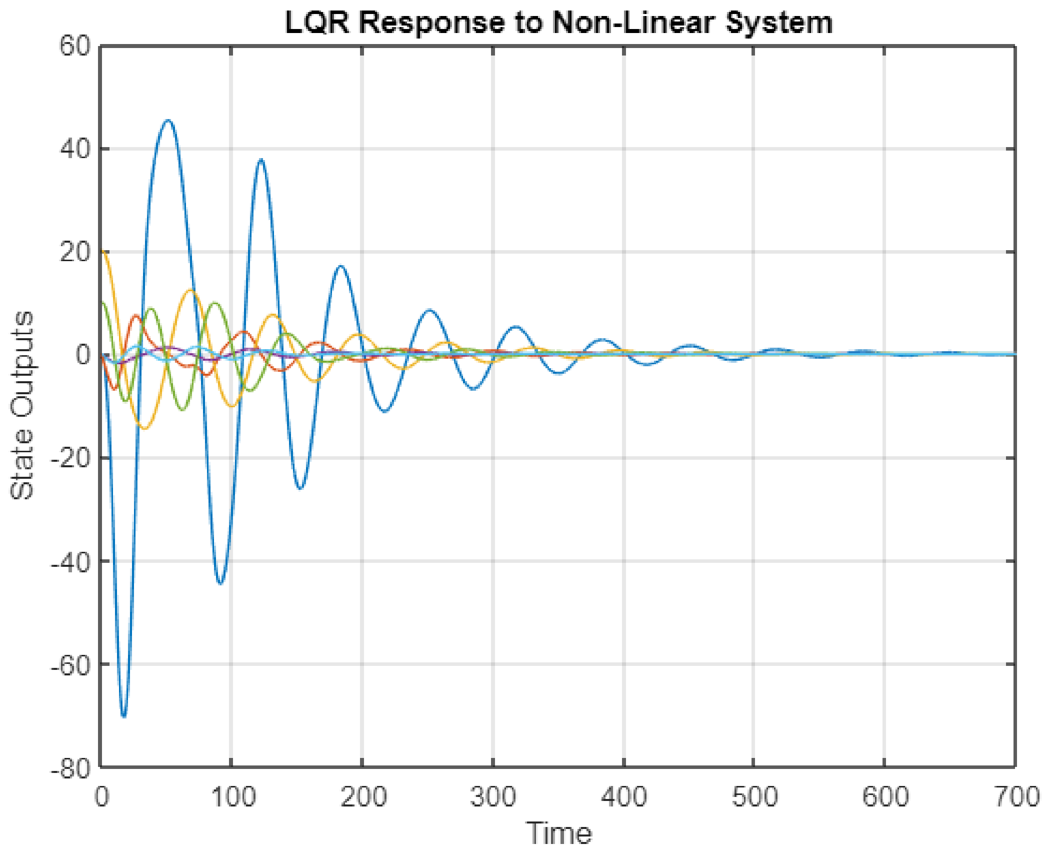
```
figure
initial(sys_2,initial_state)
```



Non-linear

```
% x, theta_1 and theta_2 values are defined
%"timespan"
    time = 0:0.1:700;
%ode45 function is used for definining a differential eqn
    [timeperiod,y1] = ode45(@ydot_func,time,initial_state);
%plotting output of the function as 2D graph
```

```
    plot(timeperiod,y1)
    title('LQR Response to Non-Linear System')
    xlabel('Time')
    ylabel('State Outputs')
    grid on
```




```
function yfunc = ydot_func(timeperiod,y)
M=1000;
m1=100;
m2=100;
l1=20;
l2=10;
g=9.81;
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=[0; 1/M; 0; 1/(M*l1); 0; 1/(M*l2)];
Q=[5 0 0 0 0 0;
    0 0 0 0 0 0 ;
```

```
    0 0 5000 0 0 0;
    0 0 0 0 0 0 ;
    0 0 0 0 5000 0;
    0 0 0 0 0 0];
R=0.1;
[K_Gain_mat, S_matrix, P] = lqr(A,B,Q,R);
F=-K_Gain_mat*y;
yfunc=zeros(6,1);
yfunc(1) = y(2);
yfunc(2)=(F-(g/2)*(m1*sind(2*y(3))+m2*sind(2*y(5)))-(m1*l1*(y(4)^2)*sind(y(3))) ...
-(m2*l2*(y(6)^2)*sind(y(5))))/(M+m1*((sind(y(3)))^2)+m2*((sind(y(5)))^2));%X_DD
yfunc(3)= y(4);
yfunc(4)= (yfunc(2)*cosd(y(3))-g*(sind(y(3))))/l1';
yfunc(5)= y(6); %theta 2D
yfunc(6)= (yfunc(2)*cosd(y(5))-g*(sind(y(5))))/l2;
end
```

6 Observability of the system

- When we consider a LTI discrete system in the state space form
  $X(k+1) = AX(k) \quad X(0) = X$
- This will have an output of Y=CX+DU
- the Observability matrix for this is $O\left(A, C\right) = \quad [CCACA^2 \ldots\ldots CA^{n-1}]T$
- When the $Rank(O(A, C))$ =s full rank, then the system that we consider is observable.

E. Observability

```
%Jacobian linearization of the non linear system of dual pendulum suspended on a crane
%From the problem statement, the output vectors are
% x(t), (theta1(t),theta2(t)), (x(t), theta1(t)),and (x(t),theta1(t),theta2(t)) ...
% as C1,C2,C3, and C4 respectively
%because Y = CX+DU, we take C matrix values such that it accounts for the
%state variable
C1 =[1 0 0 0 0 0];
%coressponding to theta1 and theta2
C2 =[0 0 1 0 0 0;
     0 0 0 0 1 0];
%corresponding to x and theta2
C3 =[1 0 0 0 0 0;
     0 0 0 0 1 0];
%corresponding to x, theta1, and theta2
C4 =[1 0 0 0 0 0;
     0 0 1 0 0 0;
     0 0 0 0 1 0];
%let observability matrix is defined as O_1, O_2, O_3, and O_4
O_1 = [C1' A'*C1' A'*A'*C1' A'*A'*A'*C1' A'*A'*A'*A'*C1' A'*A'*A'*A'*A'*C1'];
O_2 = [C2' A'*C2' A'*A'*C2' A'*A'*A'*C2' A'*A'*A'*A'*C2' A'*A'*A'*A'*A'*C2'];
O_3 = [C3' A'*C3' A'*A'*C3' A'*A'*A'*C3' A'*A'*A'*A'*C3' A'*A'*A'*A'*A'*C3'];
O_4 = [C4' A'*C4' A'*A'*C4' A'*A'*A'*C4' A'*A'*A'*A'*C4' A'*A'*A'*A'*A'*C4'];
rankArr = [rank(O_1),rank(O_2),rank(O_3),rank(O_4)];
for r = 1:4
    disp("Rank of the observability Matrix: of O_"+r+" is :"+rankArr(r))
    if rankArr(r)==6
        disp('System is observable')
    else
        disp('System is not observable')
    end
end
```

```
Rank of the observability Matrix: of O_1 is :6
System is observable
Rank of the observability Matrix: of O_2 is :4
System is not observable
Rank of the observability Matrix: of O_3 is :6
System is observable
Rank of the observability Matrix: of O_4 is :6
System is observable
```

7 Luenberger Observer

- The state space representation of the linear system:

$$\dot{X}(t) = AX(t) + Bu(t)$$
$$y(t) = CX(t) + Du(t)$$

- Luenberger observer for the input vectors with which system is observable. The observer system is,

$$\hat{X}(t) = A\hat{X}(t) + Bu(t) + L(y(t) - \hat{y}(t))$$
$$\hat{y}(t) = C\hat{X}(t) + Du(t)$$

- The code will simulate the linear and non linear system given with the initial condition and a step input at time $t = 20$ s.

7.1 Simulation of response to input conditions and unit step input

7.1.1 When applied to Linearized system

F. Luenberger Observer

```
%luenberger observer for linearized system
desired_poles = [-1;-3;-2;-4;-6;-8]
```

```
desired_poles = 6x1
    -1
    -3
    -2
    -4
    -6
    -8
```

```
%state feedback matrix
k = lqr(As,Bs,Q,R)
```

```
k = 1x6
   22.3607   234.6518   131.1793   678.6548   174.6940   360.1045
```

```
% output vectors as per pole placements
L_1 = place(As',C1',desired_poles)';
L_3 = place(As',C3',desired_poles)';
L_4 = place(As',C4',desired_poles)';
Bs_C = [Bs;zeros(size(B))]
```

```
Bs_C = 12x1
1.0e-03 *

         0
```

```
    1.0000
         0
    0.0500
         0
    0.1000
         0
         0
         0
         0
```

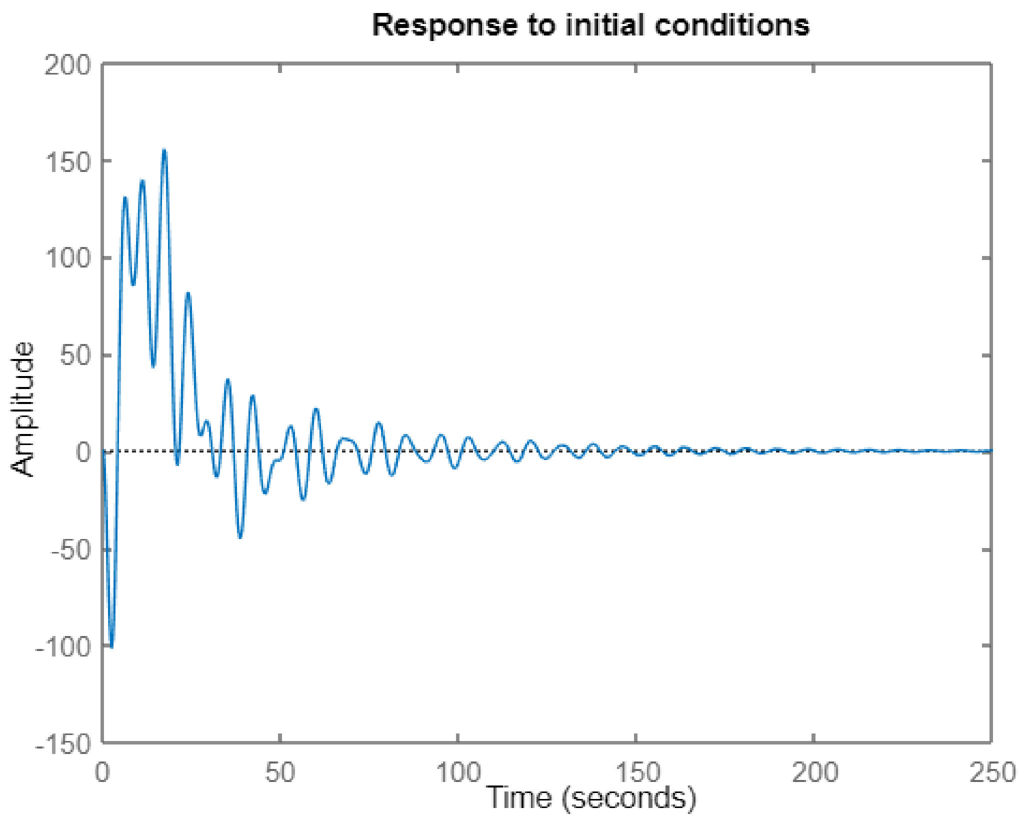
```
% output vector 1
A_C1 = [(As-Bs*k) Bs*k; zeros(size(As)) (As-L_1*C1)];
C_C1 = [C1 zeros(size(C1))];
% ouput vector 3
A_C3 = [(As-Bs*k) Bs*k; zeros(size(As)) (As-L_3*C3)];
C_C3 = [C3 zeros(size(C3))];
%output vector 4
A_C4 = [(As-Bs*k) Bs*k; zeros(size(As)) (As-L_4*C4)];
C_C4 = [C4 zeros(size(C4))];
% intial conditions
initial_state = [1;0;30;2;60;2;zeros(6,1)]
```

```
initial_state = 12x1
     1
     0
    30
     2
    60
     2
     0
     0
     0
     0
```

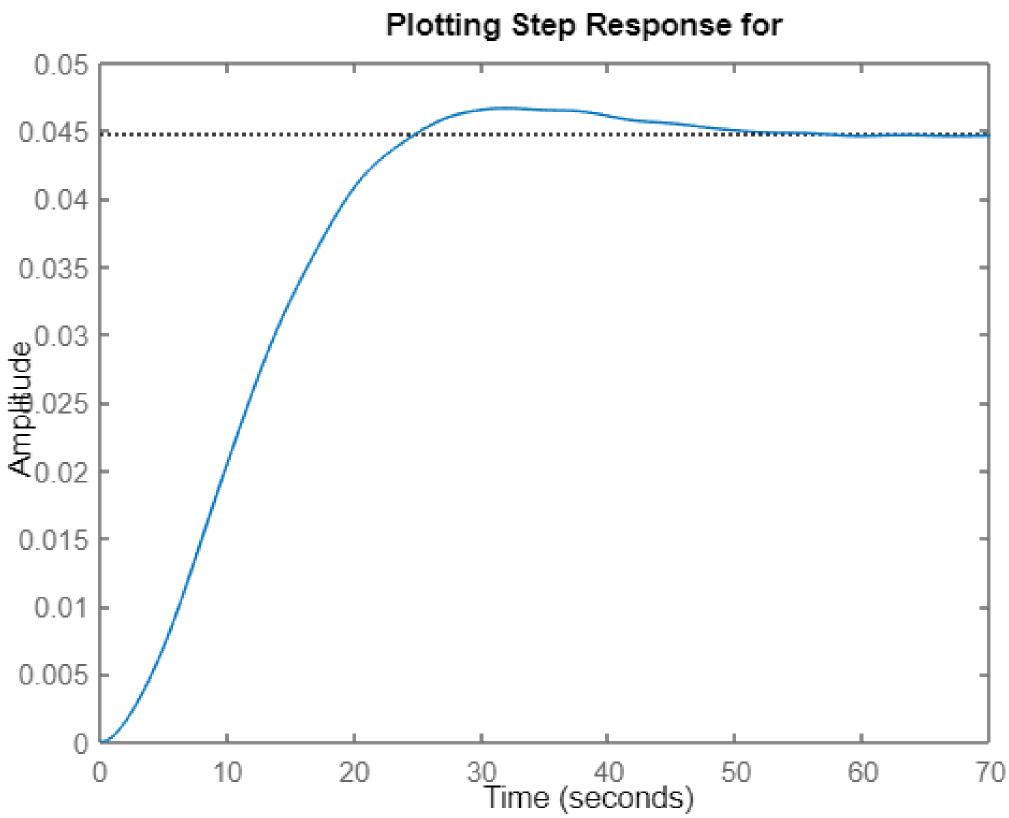
```
sys1 = ss(A_C1, Bs_C, C_C1,D);
sys3 = ss(A_C3, Bs_C, C_C3,D);
sys4 = ss(A_C4, Bs_C, C_C4,D);
disp("Observer error with x(t) output vector")
```

```
Observer error with x(t) output vector
```

```
figure
initial(sys1,initial_state)
title("Response to initial conditions");
```



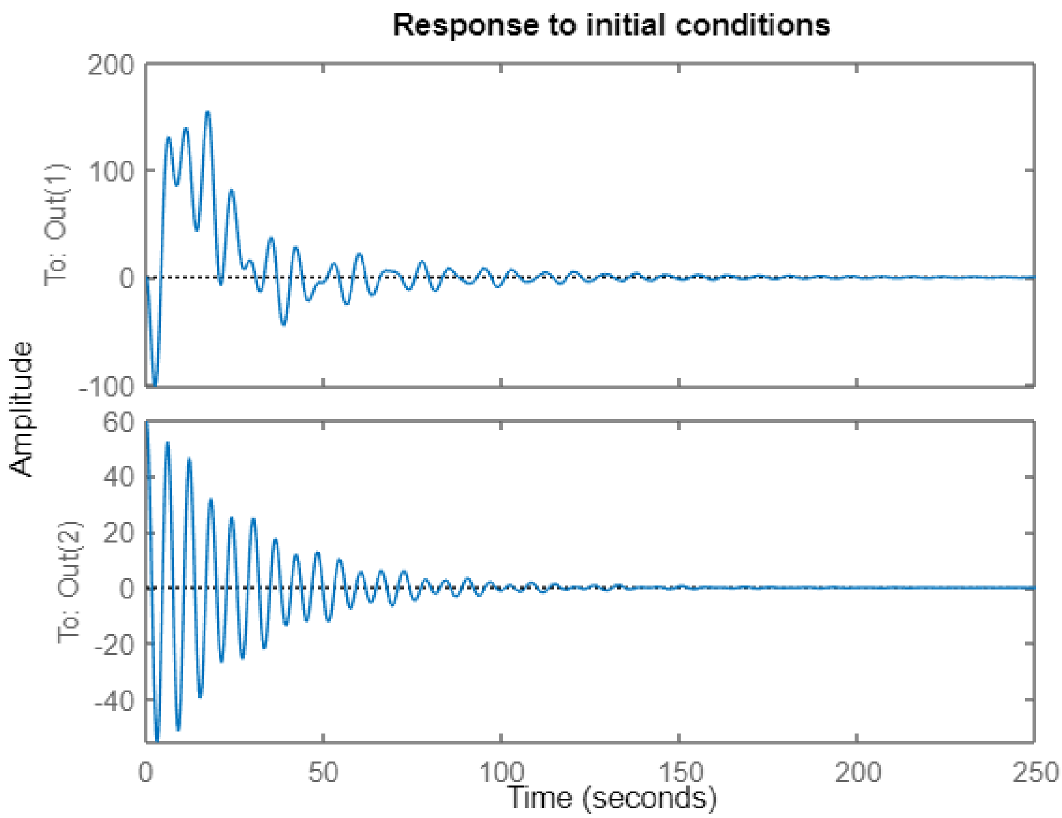

```
figure
step(sys1)
title("Plotting Step Response for ")
```



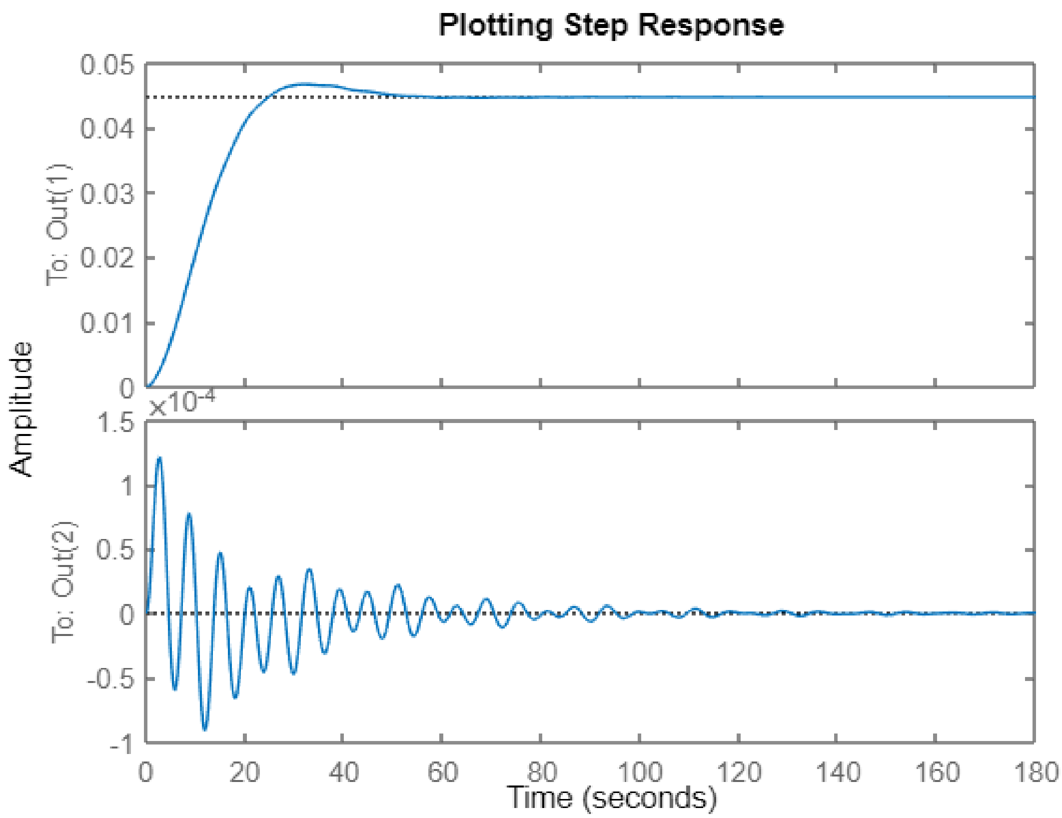

```
disp("Observer error with x(t), theta2(t)")
```

```
Observer error with x(t), theta2(t)
```

```
figure
initial(sys3,initial_state)
title("Response to initial conditions")
```



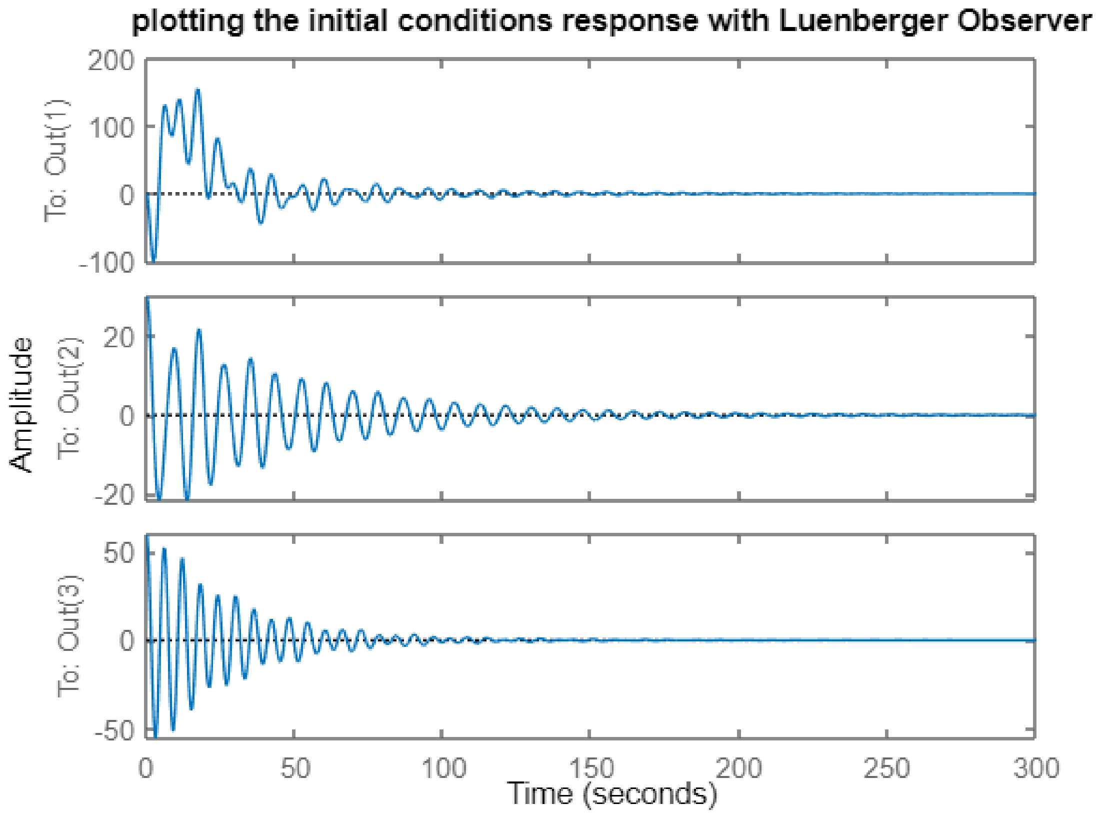

```
figure
step(sys3)
title("Plotting Step Response")
```



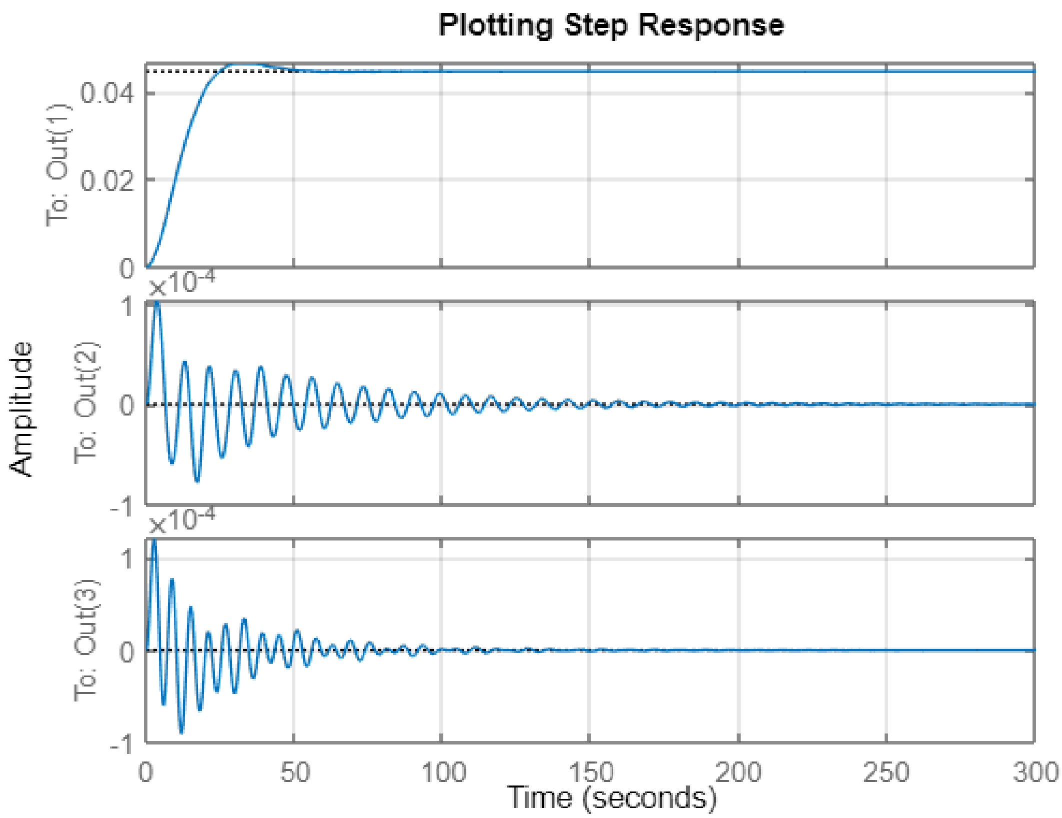

```
disp("Observer error with x(t), theta1(t), theta2(t)")
```

```
Observer error with x(t), theta1(t), theta2(t)
```

```
figure
initial(sys4,initial_state)
title("plotting the initial conditions response with Luenberger Observer")
```

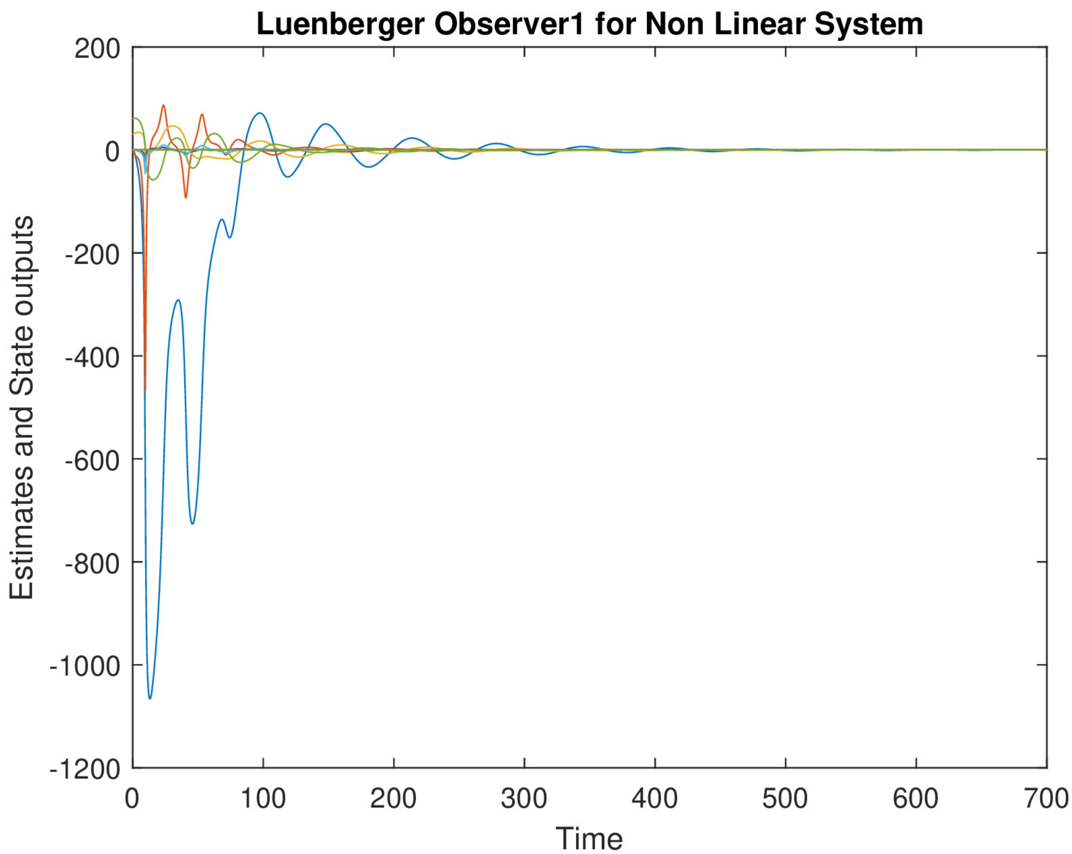



```
figure
step(sys4)
title("Plotting Step Response")
grid on
```

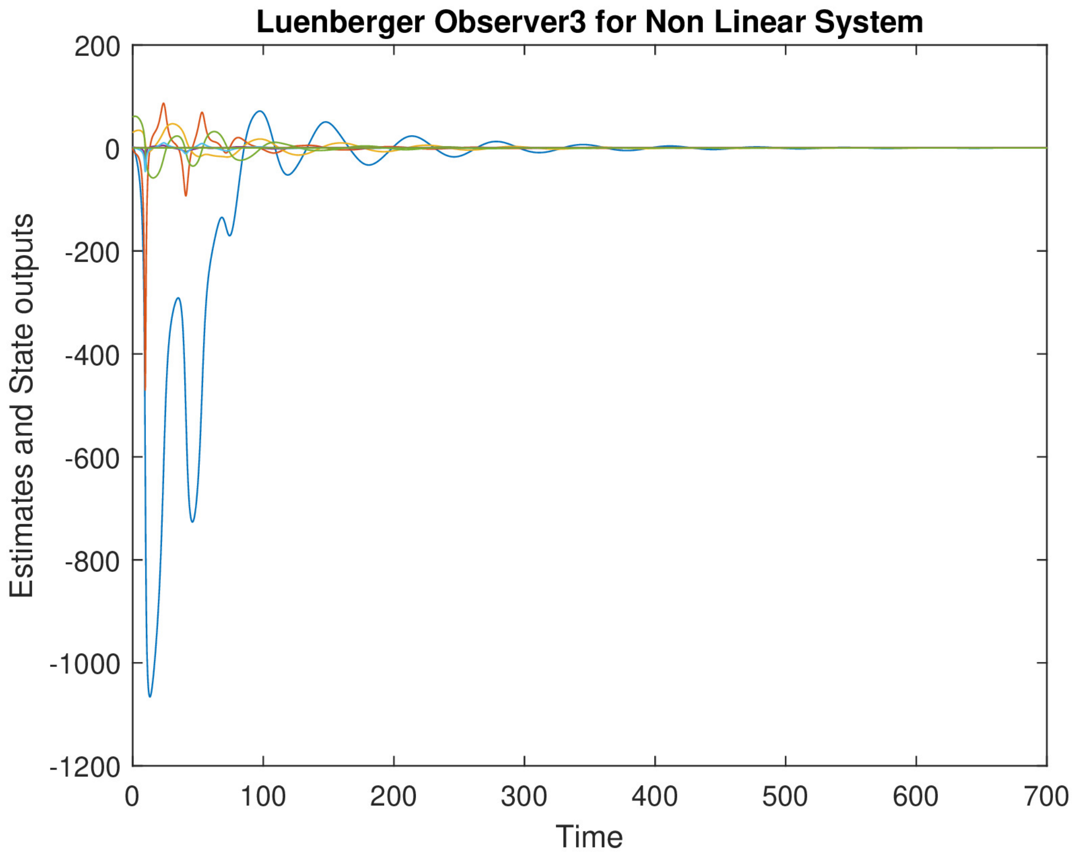


7.2 Luenberger observer for Non Linear systems

```
%Luenberger Observer for Non Linear System Model:
%plotting the estimates. Not plotting the error

simulation_time = 0:0.1:700;
[time,out] = ode45(@luenberger_func,simulation_time,initial_state);
figure
plot(time,out)
title('Luenberger Observer1 for Non Linear System')
xlabel('Time')
ylabel('Estimates and State outputs')
```




```
[time,out] = ode45(@luenberger_func3,simulation_time,initial_state);
figure
plot(time,out)
title('Luenberger Observer3 for Non Linear System')
xlabel('Time')
ylabel('Estimates and State outputs')
```


Luenberger Observer3 for Non Linear System
Estimates and State outputs
Time
200
0
-200
-400
-600
-800
-1000
-1200
0
100
200
300
400
500
600
700

8 LQG Controller Design

8.1 Linear Part

G. LQG Controller

```
QXU = eye(7)
```

```
QXU = 7x7
     1     0     0     0     0     0     0
     0     1     0     0     0     0     0
     0     0     1     0     0     0     0
     0     0     0     1     0     0     0
     0     0     0     0     1     0     0
     0     0     0     0     0     1     0
     0     0     0     0     0     0     1
```

```
% process noise
%wgn is in-buit MATLAB to generate gaussian process noise
w = wgn(6,1,4)
```

```
w = 6x1
    1.5414
   -0.8277
    0.2799
    1.5385
   -0.6561
   -0.6946
```

```
% generating measurement noise
v = wgn(1,1,4)
```

```
v = 3.1752
```

```
QWV = [w;v]*[w' v']
```

```
QWV = 7x7
    2.3760   -1.2758    0.4314    2.3715   -1.0113   -1.0707    4.8942
   -1.2758    0.6851   -0.2316   -1.2734    0.5431    0.5749   -2.6281
    0.4314   -0.2316    0.0783    0.4306   -0.1836   -0.1944    0.8886
    2.3715   -1.2734    0.4306    2.3670   -1.0094   -1.0687    4.8850
   -1.0113    0.5431   -0.1836   -1.0094    0.4305    0.4557   -2.0832
   -1.0707    0.5749   -0.1944   -1.0687    0.4557    0.4825   -2.2055
    4.8942   -2.6281    0.8886    4.8850   -2.0832   -2.2055   10.0816
```

```
%state space representation of the closed loop system with LQG controller
sys5 = ss(As,Bs,C1,D)
```

```
sys5 =

  A =
             x1      x2      x3      x4      x5      x6
   x1         0       1       0       0       0       0
   x2         0       0   -0.98       0   -0.98       0
   x3         0       0       0       1       0       0
   x4         0       0  -0.539       0  -0.049       0
   x5         0       0       0       0       0       1
   x6         0       0  -0.098       0  -1.078       0

  B =
             u1
   x1         0
   x2     0.001
   x3         0
   x4     5e-05
   x5         0
   x6    0.0001

  C =
         x1  x2  x3  x4  x5  x6
   y1     1   0   0   0   0   0

  D =
         u1
   y1     0

Continuous-time state-space model.
```

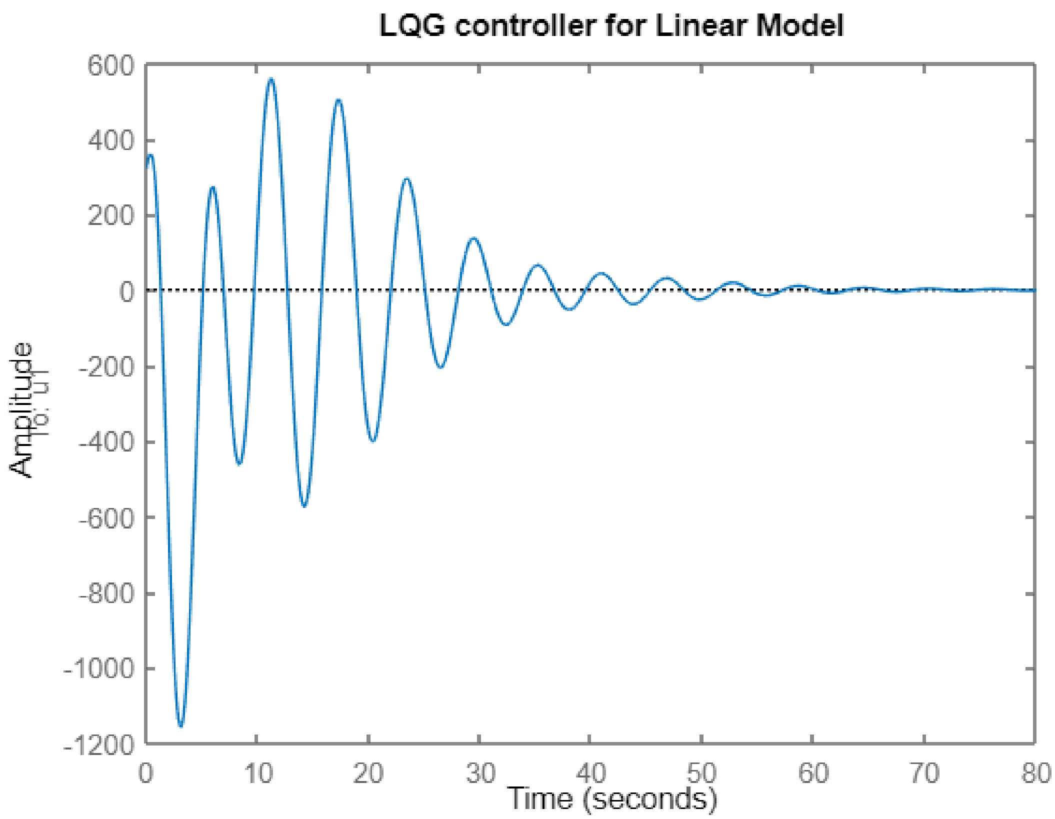
```
sys_LQG = lqg(sys5,QXU,QWV);
initial_state_2 = [1;2;30;3;30;3]
```

```
initial_state_2 = 6x1
     1
     2
    30
     3
    30
     3
```

```
figure
disp("Response at initial conditions")
```

```
Response at initial conditions
```

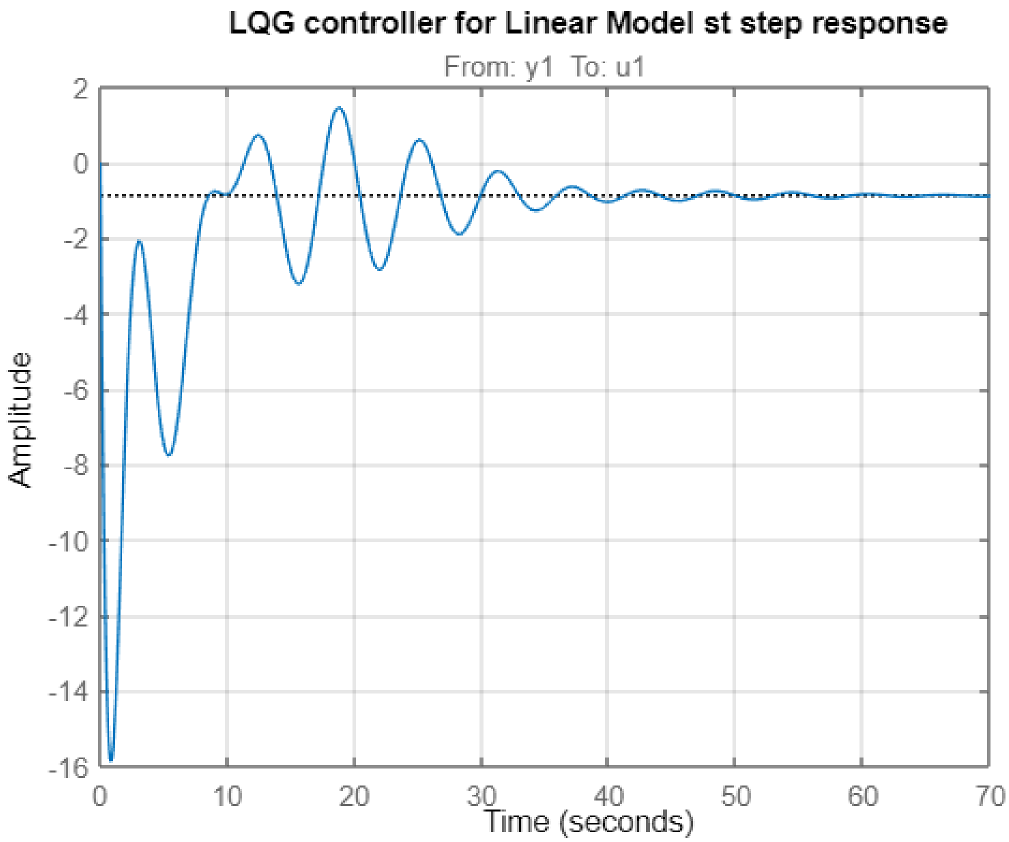
```
figure
initial(sys_LQG,initial_state_2)
title("LQG controller for Linear Model")
```




```
disp("Response at step LQR control")
```

```
Response at step LQR control
```

```
figure
step(sys_LQG)
title("LQG controller for Linear Model st step response")
grid on
```

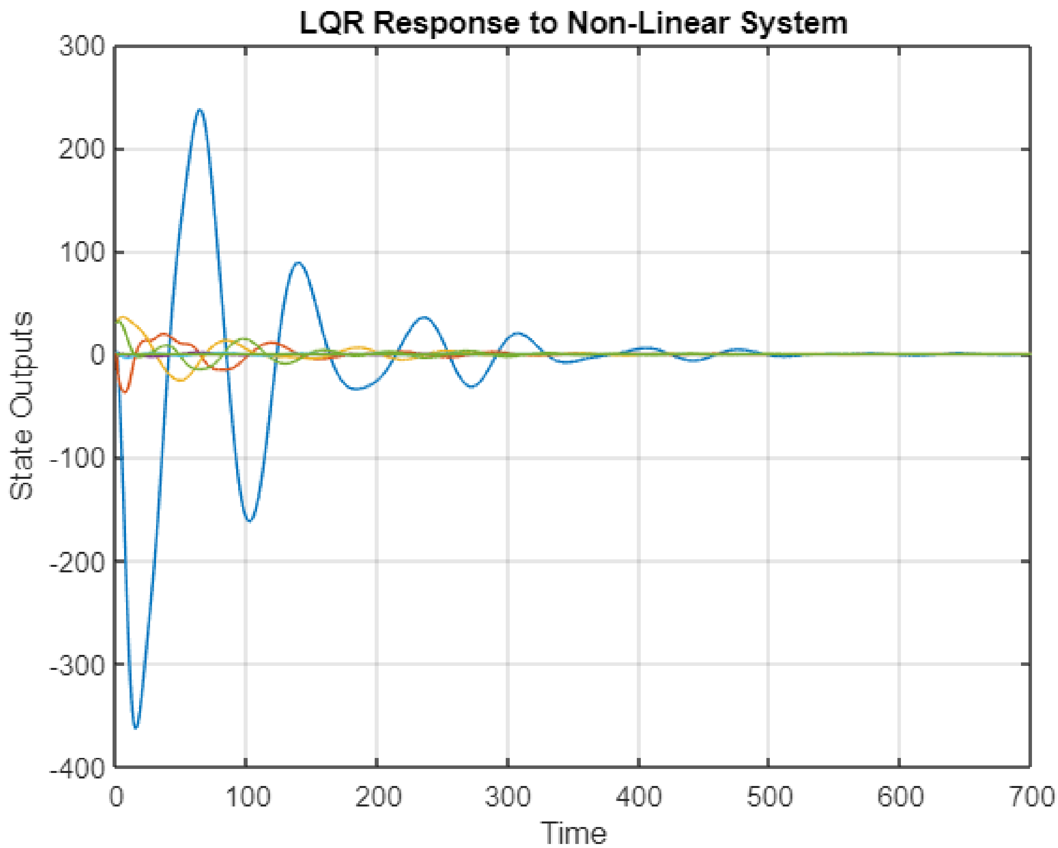

LQG controller for Linear Model st step response
From: y1 To: u1
Amplitude
Time (seconds)
2
0
-2
-4
-6
-8
-10
-12
-14
-16
0
10
20
30
40
50
60
70

8.2 Non Linear Part of LQG

```
time_samples = 0:0.1:700
initial_state_Lqg=[1;2;30;3;30;3;zeros(6,1)]
```

```
initial_state_Lqg = 12x1
     1
     2
    30
     3
    30
     3
     0
     0
     0
     0
```

```
[t2,y] = ode45(@lqg_func,time_samples,initial_state_Lqg);
%plotting output of the function as 2D graph
plot(time_samples,y)
title('LQR Response to Non-Linear System')
xlabel('Time')
ylabel('State Outputs')
grid on
```




```
%% Function doublepend_lqg for non linear LQG control
```

```
%%LQG Controller Function
function yfunc = lqg_func(t2,y)
M=1000;
m1=100;
m2=100;
l1 = 20;
l2 = 10;
g = 9.8;
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=[0; 1/M; 0; 1/(M*l1); 0; 1/(M*l2)];
Q=[5 0 0 0 0 0;
    0 0 0 0 0 0;
    0 0 5000 0 0 0;
    0 0 0 0 0 0;
```

```
    0 0 0 0 5000 0;
    0 0 0 0 0 0];
R = 0.01;
C1 = [1 0 0 0 0 0];
K = lqr(A,B,Q,R);
F = -K*y(1:6);
yfunc = zeros(12,1);
yfunc(1) = y(2);
yfunc(2)=(F-(g/2)*(m1*sind(2*y(3))+m2*sind(2*y(5)))-(m1*l1*(y(4)^2)*sind(y(3))) ...
-(m2*l2*(y(6)^2)*sind(y(5))))/(M+m1*((sind(y(3)))^2)+m2*((sind(y(5)))^2));%X_DD
yfunc(3)= y(4);
yfunc(4)= (yfunc(2)*cosd(y(3))-g*(sind(y(3))))/l1';
yfunc(5)= y(6);
yfunc(6)= (yfunc(2)*cosd(y(5))-g*(sind(y(5))))/l2;
p_noise = eye(6);
m_noise = 1;
k_gain = lqr(A',C1',p_noise,m_noise)';
est = (A-k_gain*C1)*y(7:12);
yfunc(7) = est(1);
yfunc(8) = est(2);
yfunc(9) = est(3);
yfunc(10) = est(4);
yfunc(11) = est(5);
yfunc(12) = est(6);
end
```

- To asymptotically track a constant reference on x we re-configure our controller: For the most optimal Reference Tracking, our aim is to minimize the following cost function:

$$\int_0^\infty (X(t) - X(d))^T(t)Q(X(t) - X(d)) + (Uk - U\infty)^T R(Uk - U\infty)dt$$

- To reconfigure a controller to asymptotically track a constant reference, you would need to adjust the controller's gains (i.e., the coefficients for the proportional, integral, and derivative terms) until the system's response is satisfactory. This process is known as "tuning" the controller.

- Yes, this design will accommodate constant force disturbances applied on the Cart. Under the assumption that the force disturbances are Gaussian in nature, the controller will account for these disturbances.