UNIVERSITY OF MARYLAND

ENPM667(Controls)

Formal Report

Project

Non-linear Control

Author:	*UID:*
Akwasi A. Obeng	117000801
Eashwar Sathyamurthy	116946148

December 08, 2019

Contents

1 Dynamics(Question A)

1.1 Lagrangian

Potential Energy

The potential energy associated with m_1,M and m_2 are written obtained as follows. Setting a reference point of U= 0 at the origin we get

$$V_{m_1} = -m_1 * l_1 cos(\theta_1(t)) \tag{1}$$
$$V_{m_2} = -m_2 * l_2 cos(\theta_2(t)) \tag{2}$$
$$V_M = 0 \tag{3}$$
$$V = V_{m_1} + V_{m_2} + V_M \tag{4}$$

Kinetic Energy

The Kinetic energy associated with m_1,M and m_2 are written obtained as follows

$$T_{m_1} = \frac{1}{2}m_1(\frac{\mathrm{d}x(t)}{\mathrm{d}t})^2 + \frac{1}{2}m_1 l_1^2(\frac{\mathrm{d}\theta_1}{\mathrm{d}t})^2 - m_1 l_1 \cos\theta_1(t)\frac{\mathrm{d}\theta_1}{\mathrm{d}t}\frac{\mathrm{d}x}{\mathrm{d}t} \tag{5}$$
$$T_{m_2} = \frac{1}{2}m_2(\frac{\mathrm{d}x(t)}{\mathrm{d}t})^2 + \frac{1}{2}m_2 l_2^2(\frac{\mathrm{d}\theta_2}{\mathrm{d}t})^2 - m_2 l_2 \cos\theta_2(t)\frac{\mathrm{d}\theta_2}{\mathrm{d}t}\frac{\mathrm{d}x}{\mathrm{d}t} \tag{6}$$
$$T_M = \frac{1}{2}M(\frac{\mathrm{d}x}{\mathrm{d}t})^2 \tag{7}$$
$$T = T_{m_1} + T_{m_2} + T_M \tag{8}$$

Lagrangian

The lagrangian is calculated as $T - V$ and is obtained as follows

$$L = T - V \tag{9}$$
$$\begin{aligned}&= \frac{1}{2}m_1(\frac{\mathrm{d}x(t)}{\mathrm{d}t})^2 + \frac{1}{2}m_1 l_1^2(\frac{\mathrm{d}\theta_1}{\mathrm{d}t})^2 - m_1 l_1 \cos\theta_1(t)\frac{\mathrm{d}\theta_1}{\mathrm{d}t}\frac{\mathrm{d}x}{\mathrm{d}t}+ \\ &\frac{1}{2}m_2(\frac{\mathrm{d}x(t)}{\mathrm{d}t})^2 + \frac{1}{2}m_2 l_2^2(\frac{\mathrm{d}\theta_2}{\mathrm{d}t})^2 - m_2 l_2 \cos\theta_2(t)\frac{\mathrm{d}\theta_2}{\mathrm{d}t}\frac{\mathrm{d}x}{\mathrm{d}t}+ \\ &\frac{1}{2}M(\frac{\mathrm{d}x}{\mathrm{d}t})^2 + m_1 * l_1 cos(\theta_1(t)) + m_2 * l_2 cos(\theta_2(t))\end{aligned} \tag{10}$$

1.2 Equation of Motion

$$0 = -m_1 l_1 \frac{\mathrm{d}^2 x}{\mathrm{d}t^2} \cos\theta_1(t) + m_1 l_1^2 \frac{\mathrm{d}^2\theta_1}{\mathrm{d}t^2} + mgl_1 \sin\theta_1(t) \tag{11}$$

$$0 = -m_2 l_2 \frac{\mathrm{d}^2 x}{\mathrm{d}t^2} \cos\theta_2(t) + m_2 l_2^2 \frac{\mathrm{d}^2\theta_2}{\mathrm{d}t^2} + mgl_2 2 \sin\theta_2(t) \tag{12}$$

$$\begin{aligned} F = {} & \frac{\mathrm{d}^2 x(t)}{\mathrm{d}t^2}[M + m_1 + m_2] - m_1 l_1 [\frac{\mathrm{d}^2\theta_1(t)}{\mathrm{d}t^2} cos(\theta_1(t)) - \sin\theta_1(t)(\frac{\mathrm{d}\theta_1}{\mathrm{d}t})^2] \\ & - m_2 l_2 [\frac{\mathrm{d}^2\theta_2}{\mathrm{d}t^2} \cos\theta_2(t) - \sin\theta_2(t)(\frac{\mathrm{d}\theta_2}{\mathrm{d}t})^2]. \end{aligned} \tag{13}$$

NB.Refer to Appendix(Code for Dynamics) on how this was obtained

The equations obtained above can be further simplified to obtain the equation stated below

Simplification

Make $\frac{\mathrm{d}^2\theta_2(t)}{dt^2}$ and $\frac{\mathrm{d}^2\theta_1(t)}{dt^2}$ the subject of equations 11 and 12 respectively. Then substitute into equation 13 to get the equation below.

$$\frac{\mathrm{d}^2 x}{\mathrm{d}t^2} = \frac{-\frac{g}{2}[m_1 \sin(2\theta_1(t)) + m_2 sin(2\theta_2(t))] - m_1 l_1 \sin(\theta_1(t))(\frac{\mathrm{d}\theta_1(t)}{\mathrm{d}t})^2 - m_2 l_2 \sin(\theta_2(t))(\frac{\mathrm{d}\theta_2}{\mathrm{d}t})^2 + F}{M + m_1 \sin^2(\theta_1(t)) + m_2 \sin^2(\theta_2(t))} \tag{14}$$

Resubstitue the equation above into equation 11 and 12 to get the following.

$$\frac{\mathrm{d}^2\theta_1}{\mathrm{d}t^2} = \frac{\frac{cos(\theta_1)}{l_1}(F - \frac{m_1 g \sin(2\theta_1)}{2} - \frac{m_2 g \sin(2\theta_2)}{2} - m_1 l_1 \sin\theta_1(\frac{\mathrm{d}\theta_1}{\mathrm{d}t})^2 - m_2 l_2 \sin\theta_2(\frac{\mathrm{d}\theta_2}{\mathrm{d}t})^2)}{M + m_1 \sin^2(\theta_1(t)) + m_2 \sin^2(\theta_2(t))} \tag{15}$$

$$\frac{\mathrm{d}^2\theta_2}{\mathrm{d}t^2} = \frac{\frac{cos(\theta_2)}{l_2}(F - \frac{m_1 g \sin(2\theta_1)}{2} - \frac{m_2 g \sin(2\theta_2)}{2} - m_1 l_1 \sin\theta_1(\frac{\mathrm{d}\theta_1}{\mathrm{d}t})^2 - m_2 l_2 \sin\theta_2(\frac{\mathrm{d}\theta_2}{\mathrm{d}t})^2)}{M + m_1 \sin^2(\theta_1(t)) + m_2 \sin^2(\theta_2(t))} \tag{16}$$

2 Linearization (Question B)

A non linear system,$F(X,U)$ with state space euation $\dfrac{\mathrm{d}X}{\mathrm{d}t} = F(X,U)$ can be linearized with the equation stated below.

$$\delta_x = A_F \frac{\mathrm{d}x(t)}{\mathrm{d}t} + B_F \frac{\mathrm{d}x(t)}{\mathrm{d}t} \tag{17}$$

where $A_F = \nabla_x F$,$B_F = \nabla_u F$, U(input) and X(State)

Given the equation above, the following state space is chosen for the Dynamic problem.

$$X = \begin{bmatrix} x & \frac{\mathrm{d}x}{\mathrm{d}t} & \theta_1 & \frac{\mathrm{d}\theta_1}{\mathrm{d}t} & \theta_2 & \frac{\mathrm{d}\theta_2}{\mathrm{d}t} \end{bmatrix}^T \tag{18}$$

Therefore, the state space equation can be written as

$$\frac{\mathrm{d}X}{\mathrm{d}t} = \frac{\mathrm{d}}{\mathrm{d}t} \begin{bmatrix} x \\ \frac{\mathrm{d}x}{\mathrm{d}t} \\ \theta_1 \\ \frac{\mathrm{d}\theta_1}{\mathrm{d}t} \\ \theta_2 \\ \frac{\mathrm{d}\theta_2}{\mathrm{d}t} \end{bmatrix} = \begin{bmatrix} f_1(X,U) \\ f_2(X,U) \\ f_3(X,U) \\ f_4(X,U) \\ f_5(X,U) \\ f_6(X,U) \end{bmatrix} = F \tag{19}$$

Based on equation above, the following can be noted

$$f_1(X,U) = \frac{\mathrm{d}x}{\mathrm{d}t} \tag{20}$$
$$f_2(X,U) = \frac{\mathrm{d}^2x}{\mathrm{d}t^2} \quad \text{Refer to Dynamics for equation} \tag{21}$$
$$f_3(X,U) = \frac{\mathrm{d}\theta}{\mathrm{d}t} \tag{22}$$
$$f_4(X,U) = \frac{\mathrm{d}^2\theta_1}{\mathrm{d}t^2} \quad \text{Refer to Dynamics for equation} \tag{23}$$
$$f_5(X,U) = \frac{\mathrm{d}\theta_2}{\mathrm{d}t} \tag{24}$$
$$f_6(x,U) = \frac{\mathrm{d}^2\theta_2}{\mathrm{d}t^2} \quad \text{Refer to Dynamics for equation} \tag{25}$$
$$\tag{26}$$

From $A_F = \nabla_x F = \frac{\partial F}{\partial X}$ and $B_F = \nabla_u F = \frac{\partial F}{\partial U}$ we get

$$A_F = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{m_1 g}{M} & 0 & -\frac{m_2 g}{M} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{g}{l_1}(1+\frac{m_1}{M}) & 0 & \frac{-m_2 g}{l_1 M} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{-m_1 g}{l_2 M} & 0 & -\frac{g}{l_1}(1+\frac{m_2}{M}) & 0 \end{bmatrix} \quad B_F = \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ \frac{1}{Ml_1} \\ 0 \\ \frac{1}{Ml_2} \end{bmatrix} \tag{27}$$

3 Controllability (Question C)

The controllability matrix,C, is a 6x6 matrix and is controllable when

$$Det(C) \neq 0 \tag{28}$$

Condition for Controllability

$$Det(C) = -\frac{(g^6 l_1^2 - 2g^6 l_1 l_2 + g^6 l_2^2)}{M^6 l_1^6 l_2^6} \tag{29}$$
$$Det(C) = -\frac{g^6}{M^6 l_1^6 l_2^6}(l_1 - l_2)^2 \tag{30}$$

Therefore, Linearized system is controllable when $l_1 \neq l_2$
Refer to code for information on these where obtained

4 FeedBack and Linear Quadratic Regulator (Question D)

Linearized System with Feedback

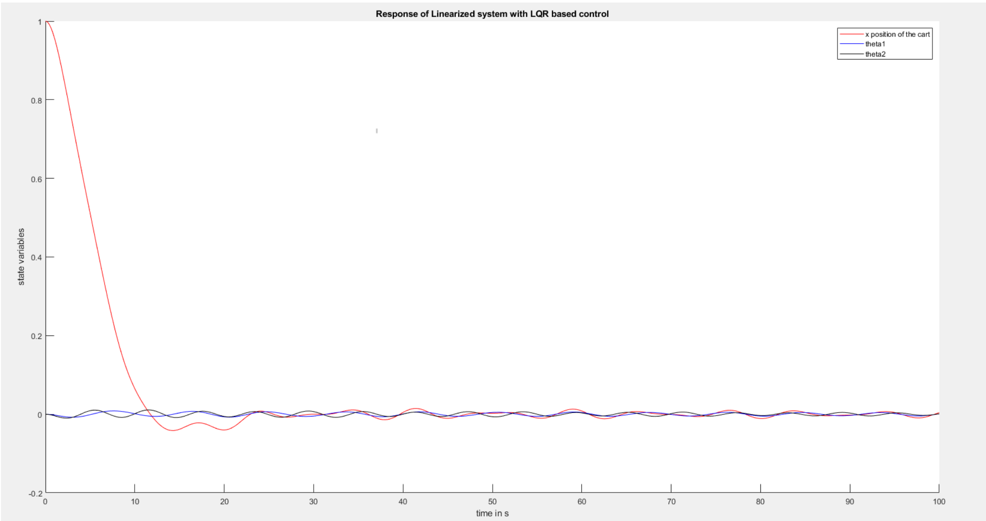

Figure 1: Linearized System with Feedback

LQR Non-Linear System

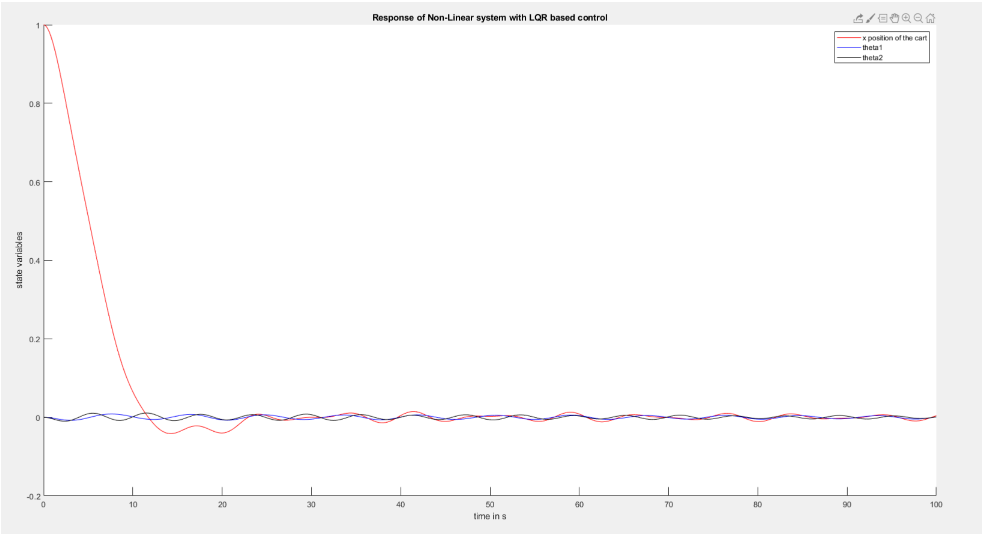

Figure 2: LQR Non linear System

Refer to code for information on how these where obtained

5 Observability (Question E)

Observability x(t)

```
obl =

    1.0000         0         0         0         0         0
         0    1.0000         0         0         0         0
         0         0   -1.0000         0    0.6500         0
         0         0         0   -1.0000         0    0.6500
         0         0   -1.0000         0    1.1500         0
         0         0         0   -1.0000         0    1.1500
```

$\theta_2(t)$ and $\theta_1(t)$

ob2 =

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
1.0000	1.0000	0	0	-0.5500	-0.6500	0	0	0.3075
0	0	1.0000	1.0000	0	0	-0.5500	-0.6500	0
0	1.0000	0	0	-0.0500	-1.1500	0	0	0.0825
0	0	0	1.0000	0	0	-0.0500	-1.1500	0

Observability x(t) and $\theta_2(t)$

ob3 =

1.0000	0	0	0	0	0	0	0	0	0	0	0
0	0	1.0000	0	0	0	0	0	0	0	0	0
0	0	0	0	-1.0000	-0.1000	0	0	0.6500	0.1650	0	0
0	0	0	0	0	0	-1.0000	-0.1000	0	0	0.6500	0.1650
0	1.0000	0	0	-1.0000	-1.1000	0	0	1.1500	1.2150	0	0
0	0	0	1.0000	0	0	-1.0000	-1.1000	0	0	1.1500	1.2150

Observability x(t), $\theta_2(t)$ and $\theta_1(t)$

ob4 =

1.0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1.0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1.0000	0	0	0	0	-1.0000	-0.5500	-0.1000	0	0	0	0.6500	0.3075	0.1650	0	0	0
0	0	0	0	1.0000	0	0	0	0	-1.0000	-0.5500	-0.1000	0	0	0	0.6500	0.3075	0.1650
0	0	1.0000	0	0	0	-1.0000	-0.0500	-1.1000	0	0	0	1.1500	0.0825	1.2150	0	0	0
0	0	0	0	0	1.0000	0	0	0	-1.0000	-0.0500	-1.1000	0	0	0	1.1500	0.0825	1.2150

Observability x(t), $\theta_2(t)$ and $\theta_1(t)$

```
observability_of_C1 =

     6

Rank is 6. The system is observable for output x(t)

observability_of_C2 =

     4

Rank is 4. The system is not observable for output (t1(t),t2(t))

observability_of_C3 =

     6

The system is observable for output (x(t),t2(t))

observability_of_C4 =

     6

Rank is 6.The system is observable for output (x(t),t1(t),t2(t))
```

Refer to code for information on these where generated

6 Luenberger Observer (Question F)

State x(t) Linear Response

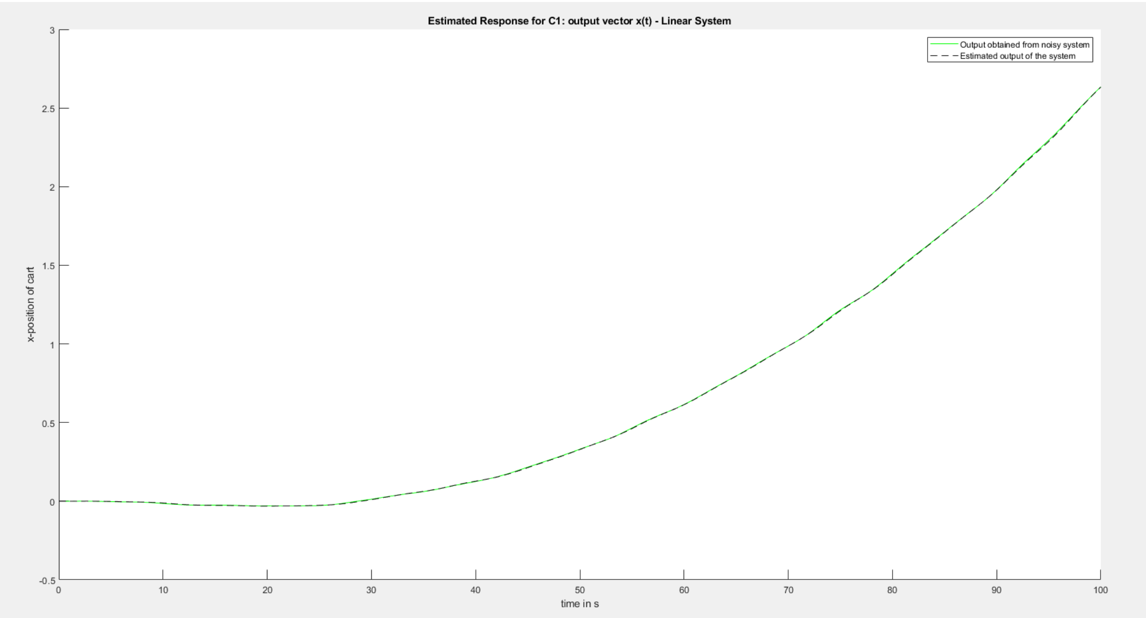

Figure 3: State x(t) Linear Response

State x(t) Non Linear Response

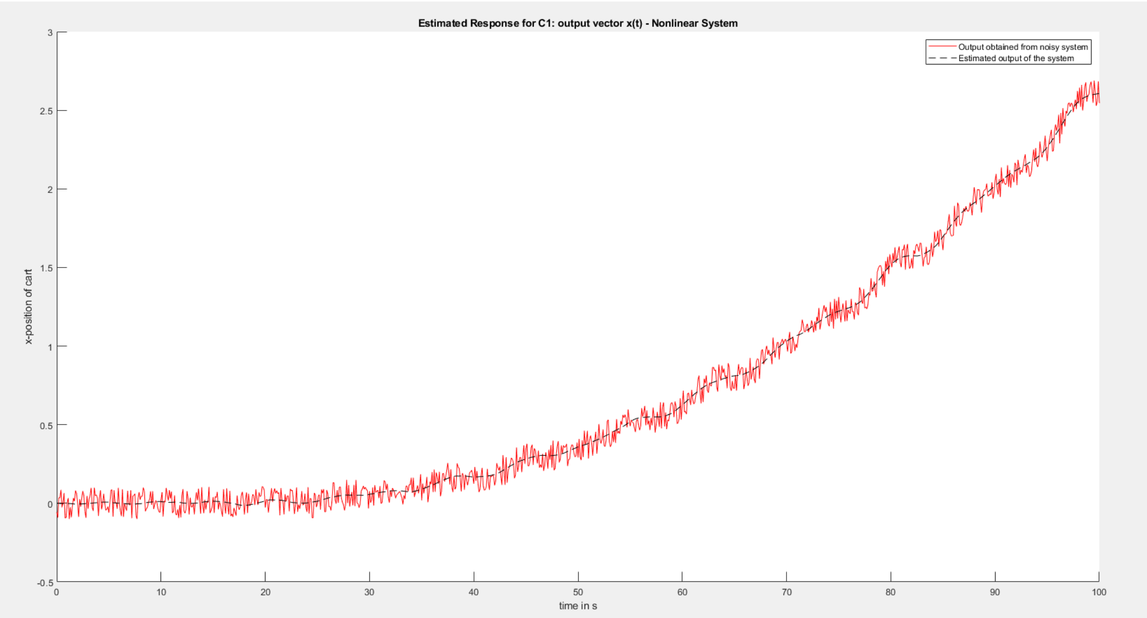

Figure 4: State x(t) Non Linear Response

State x(t) and $\theta_2(t)$ Linear Response

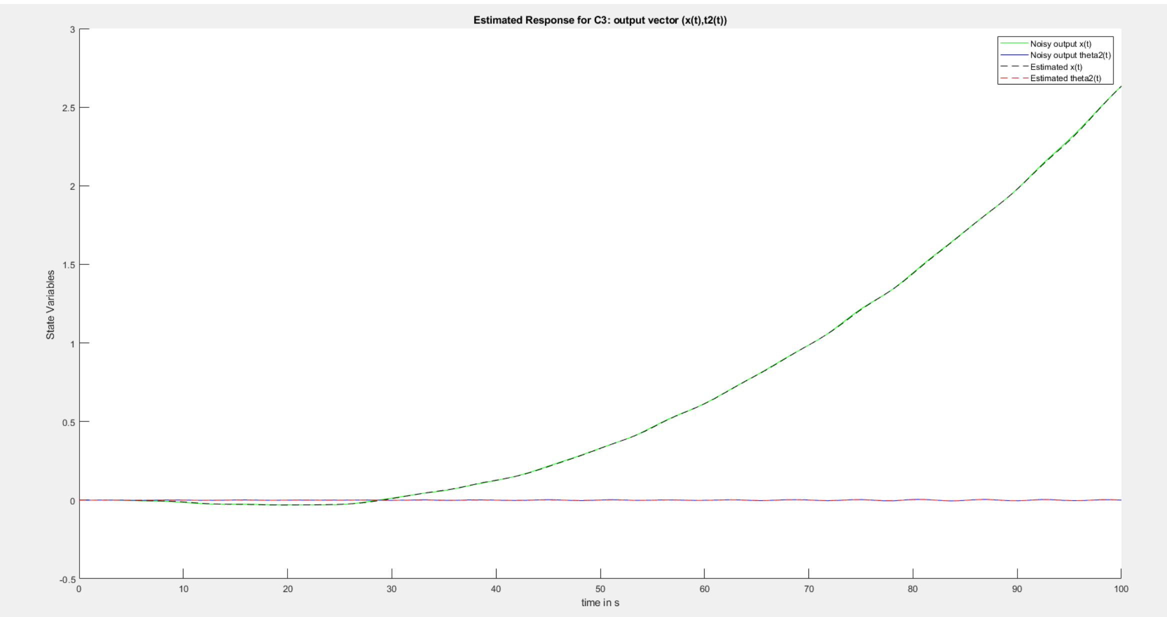

Figure 5: State x(t) and $\theta_2(t)$ Linear Response

State x(t) and $\theta_2(t)$ Non-Linear Response

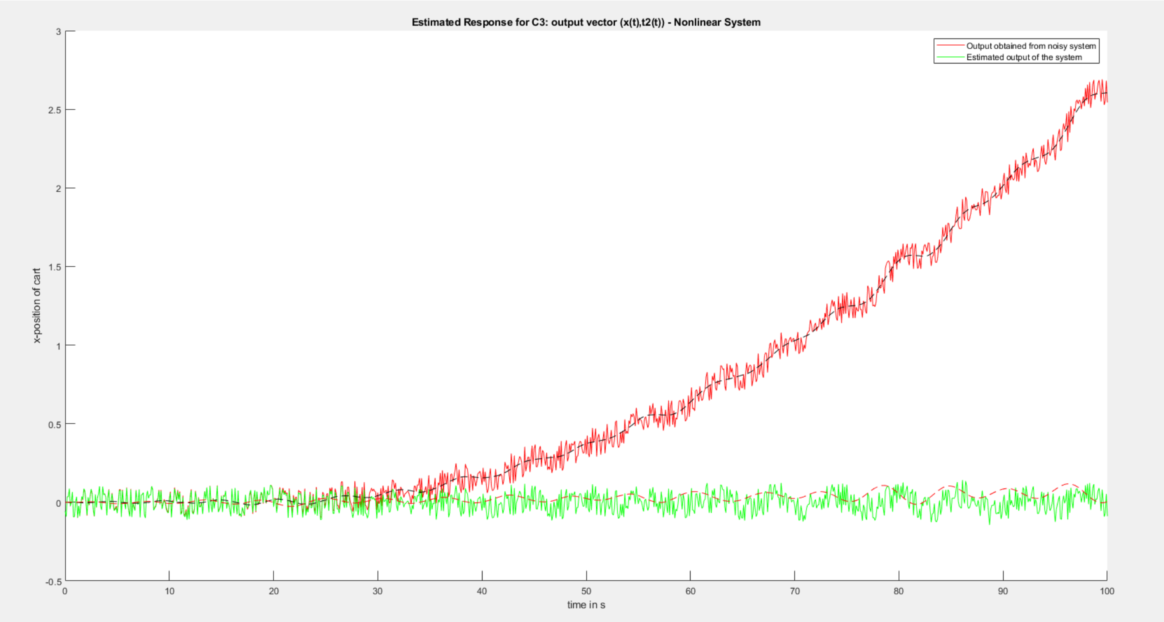

Figure 6: State x(t) and $\theta_2(t)$ Non-Linear Response

State x(t),$\theta_1(t)$ and $\theta_2(t)$ Linear Response

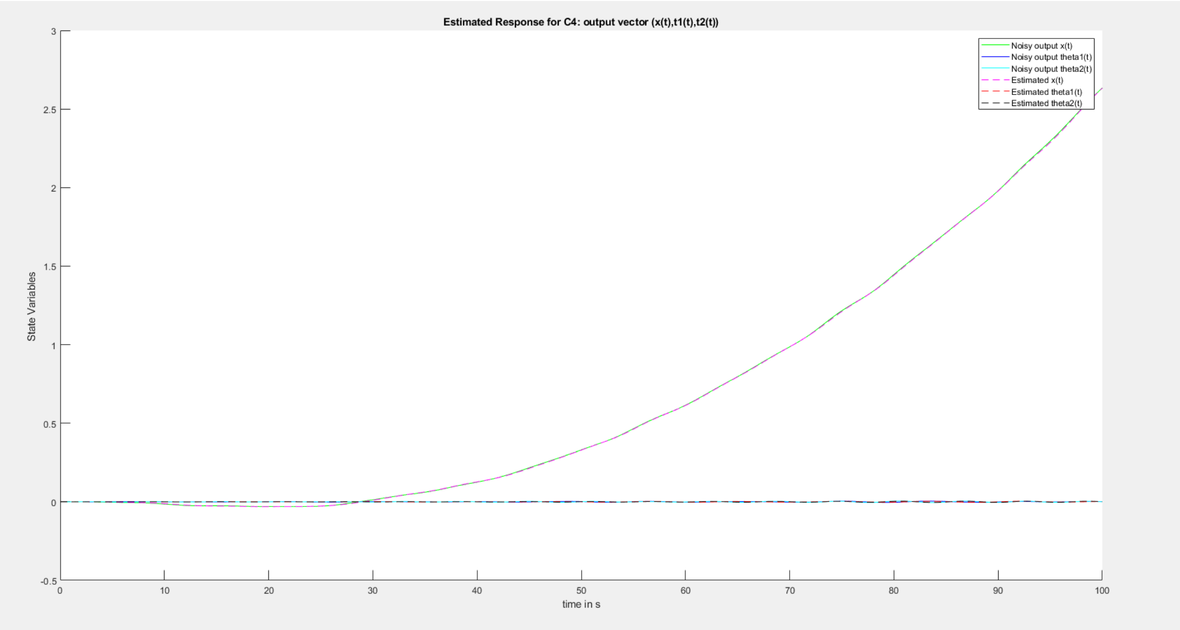

Figure 7: State x(t),$\theta_1 t()$ and $\theta_2(t)$ Linear Response

State x(t),$\theta_1(t)$ and $\theta_2(t)$ Non-Linear Response

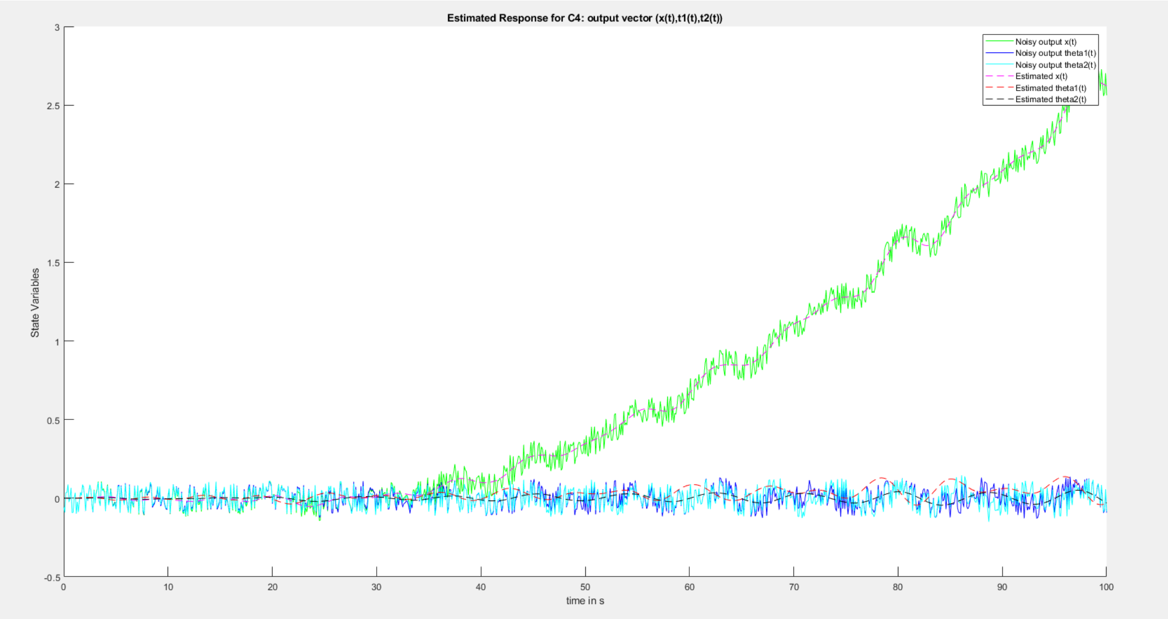

Figure 8: State x(t),$\theta_1 t()$ and $\theta_2(t)$ Non-Linear Response

Refer to code for information on how these where obtained

7 LQG Non linear

State x(t),$\theta_1(t)$ and $\theta_2(t)$ Non-Linear Response

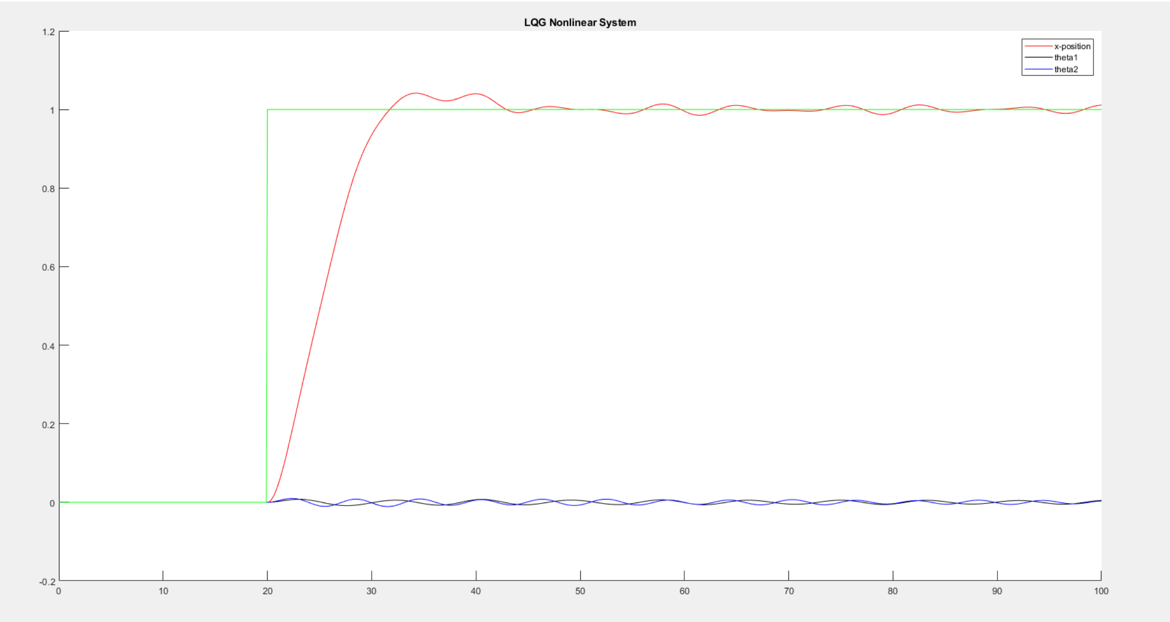

Figure 9: LQG Non linear

Refer to Code for information on how this was achieved

8 Appendix

8.1 Code for Dynamics Calculations

```
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                        PROBLEM 3
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Note This approach doesn't account for Rotational Inertia'

syms I  m1 m2 M g F1 F2 F3 F
syms q1(t) q2(t) q3(t) q1_d(t) q1_dd(t) q2_d(t) q2_dd(t)
    q3_d(t) q3_dd(t)
syms l1 l2
%%%%%%%%%%%%%%%%%%%%%%
% Symbol definitions
%%%%%%%%%%%%%%%%%%%%%%
% q(t) - Represents generalized coordinates
      % q1 represents theta 1
      % q2 represents theta 2
      % q3 represents x(t)
% F - Represents generalized force or torque
% q_d(t) - Represents derivative of q wrt to t
% q_dd(t)- Represents second derivative of q wrt to t
% r - Position of COM wrt to inertial(world) frame
% a's - robot link parameters
% I - Rotational Inertia from links

%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                          Lagrangian
%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
r1 = [q3-l1*sin(q1)
      -l1*cos(q1)];

r2 = [q3-l2*sin(q2)
      -l2*cos(q2)];

r3 = q3;

% Getting velocities
```

```
v1 = diff(r1);
v2 = diff(r2);
v3 = diff(r3);

% Kinetic Energy
T1 = 0.5 * transpose(v1) * v1*m1;
T2 = 0.5 * transpose(v2) * v2*m2;
T3 = 0.5 * transpose(v3) * v3*M;

T = T1 + T2 + T3;
T=simplify(T);

% Potential Energy
V1 = -m1*g*l1*cos(q1);
V2 = -m2*g*l2*cos(q2);
V3 = 0;

V = V1 + V2 + V3;
V=simplify(V)

% Lagrangian
L = T-V;
L = simplify(L)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                              Taking derivatives
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%       Derivative of L wrt to q
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

L=subs(L,diff(q1(t),t),q1_d);
L=subs(L,diff(q2(t),t),q2_d);
L=subs(L,diff(q3(t),t),q3_d);

```

```
dL_dq1 = functionalDerivative(L,q1);
dL_dq2 = functionalDerivative(L,q2);
dL_dq3 = functionalDerivative(L,q3);


 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%    Derivative of L wrt q_dot
 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Replacing derivatives with symbols
% Matlab can't take functional derivative with
% respect to a derivative
dL_dq1_dot= functionalDerivative(L,q1_d);
dL_dq2_dot= functionalDerivative(L,q2_d);
dL_dq3_dot= functionalDerivative(L,q3_d);

 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
  %Derivative of dl/dq_dot wrt to t
 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Taking derivative of dL_dq with respect to t
dt_dL_dq1_dot = diff(dL_dq1_dot,t);
dt_dL_dq2_dot = diff(dL_dq2_dot,t);
dt_dL_dq3_dot = diff(dL_dq3_dot,t);


 %
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                    %Equation of Motion
 %
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 %Replacing derivatives with symbols
 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

 %Replacing derivatives with symbols to perform substaction
dt_dL_dq1_dot=subs(dt_dL_dq1_dot,diff(q1_d(t),t),q1_dd);
dt_dL_dq1_dot=subs(dt_dL_dq1_dot,diff(q1(t),t),q1_d);
```

```
dt_dL_dq1_dot=subs(dt_dL_dq1_dot, diff(q2_d(t),t),q2_dd);
dt_dL_dq1_dot=subs(dt_dL_dq1_dot, diff(q2(t),t),q2_d);
dt_dL_dq1_dot=subs(dt_dL_dq1_dot, diff(q3_d(t),t),q3_dd);
dt_dL_dq1_dot=subs(dt_dL_dq1_dot, diff(q3(t),t),q3_d);

dt_dL_dq2_dot=subs(dt_dL_dq2_dot, diff(q1_d(t),t),q1_dd);
dt_dL_dq2_dot=subs(dt_dL_dq2_dot, diff(q1(t),t),q1_d);
dt_dL_dq2_dot=subs(dt_dL_dq2_dot, diff(q2_d(t),t),q2_dd);
dt_dL_dq2_dot=subs(dt_dL_dq2_dot, diff(q2(t),t),q2_d);
dt_dL_dq2_dot=subs(dt_dL_dq2_dot, diff(q3_d(t),t),q3_dd);
dt_dL_dq2_dot=subs(dt_dL_dq2_dot, diff(q3(t),t),q3_d);

dt_dL_dq3_dot=subs(dt_dL_dq3_dot, diff(q1_d(t),t),q1_dd);
dt_dL_dq3_dot=subs(dt_dL_dq3_dot, diff(q1(t),t),q1_d);
dt_dL_dq3_dot=subs(dt_dL_dq3_dot, diff(q2_d(t),t),q2_dd);
dt_dL_dq3_dot=subs(dt_dL_dq3_dot, diff(q2(t),t),q2_d);
dt_dL_dq3_dot=subs(dt_dL_dq3_dot, diff(q3_d(t),t),q3_dd);
dt_dL_dq3_dot=subs(dt_dL_dq3_dot, diff(q3(t),t),q3_d);


 %Replacing derivatives with symbols to perform substaction


 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 %           Finding Equations
 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
eqn1 = simplify(dt_dL_dq1_dot - dL_dq1) == 0
eqn2 = simplify(dt_dL_dq2_dot - dL_dq2) == 0
eqn3 = simplify(dt_dL_dq3_dot - dL_dq3) == F
```

Output

Output of langrange Dynamics

```
>> proj

V(t) =

- g*l1*m1*cos(q1(t)) - g*l2*m2*cos(q2(t))


L(t) =

(M*diff(q3(t), t)^2)/2 + (m1*diff(q3(t), t)^2)/2 + (m2*diff(q3(t), t)^2)/2 + (l1^2*m1*diff(q1(t), t)^2)/2 +
(l2^2*m2*diff(q2(t), t)^2)/2 + g*l1*m1*cos(q1(t)) + g*l2*m2*cos(q2(t)) - l1*m1*cos(q1(t))*diff(q1(t), t)*diff(q3(t),
t) - l2*m2*cos(q2(t))*diff(q2(t), t)*diff(q3(t), t)


eqn1(t) =

l1^2*m1*q1_dd(t) - l1*m1*cos(q1(t))*q3_dd(t) + g*l1*m1*sin(q1(t)) == 0


eqn2(t) =

l2^2*m2*q2_dd(t) - l2*m2*cos(q2(t))*q3_dd(t) + g*l2*m2*sin(q2(t)) == 0


eqn3(t) =

l1*m1*sin(q1(t))*q1_d(t)^2 + l2*m2*sin(q2(t))*q2_d(t)^2 + M*q3_dd(t) + m1*q3_dd(t) + m2*q3_dd(t) -
l1*m1*cos(q1(t))*q1_dd(t) - l2*m2*cos(q2(t))*q2_dd(t) == F
```

Figure 10: lagrange Dynamics

8.2 Code for LQR,LQG and Observability

```
function c = compute(m1,m2,M,l1,l2,g)
    syms t
%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                         LQR CONTROLLER
%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %[A,B] = mat(m1,m2,M,l1,l2,g);

A = [0,1.0000,0,0,0,0;
         0,0,-1.0000,0,-1.0000,0;
         0,0,0,1.0000,0,0;
         0,0,-0.5500,0,-0.0500,0;
         0,0,0,0,0,1.0000;
         0,0,-0.1000,0,-1.1000,0];
```

```
    B = [0;0.001;0;0.00005;0;0.0001];

    Eigen = eig(A)
    disp('The eigen values of A ')
    disp('therefore Lyapunovs indirect method is inconclusive for the system')
    tspan = 0:0.1:100;
    Q = 100*eye(6);
    R= 0.01;
    [K,P,E] = lqr(double(A),double(B),Q,R)
    eigen = eig(A-B*K)
    C1 = [1 0 0 0 0 0]              %For output (x(t))
    C2 = [0 0 1 0 0 0;           %For output (theta1(t),theta2(t))
          0 0 1 0 1 0]
    C3 = [1 0 0 0 0 0;           %For output (x(t),theta2(t))
          0 0 0 0 1 0]
    C4 = [1 0 0 0 0 0;            %For output (x(t),theta1(t),theta2(t))
          0 0 1 0 0 0;
          0 0 0 0 1 0]

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
   %Checking Observability
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    ob1 = [C1;C1*A;C1*A^(2);C1*A^(3);C1*A^(4);C1*A^(5)]'
    ob2 = [C2;C2*A;C2*A^(2);C2*A^(3);C2*A^(4);C2*A^(5)]'
    ob3 = [C3;C3*A;C3*A^(2);C3*A^(3);C3*A^(4);C3*A^(5)]'
    ob4 = [C4;C4*A;C4*A^(2);C4*A^(3);C4*A^(4);C4*A^(5)]'

    observability_of_C1 = rank(ob1)
    disp('Rank is 6. The system is observable for output x(t)')
    observability_of_C2 = rank(ob2)
    disp('Rank is 4. The system is not observable for output
```

```
(t1(t),t2(t))')
observability_of_C3 = rank(ob3)
disp('The system is observable for output (x(t),t2(t))')
observability_of_C4 = rank(ob4)
disp('Rank is 6.The system is observable for output (x(t),t1(t),t2(t))')
s0 = [1; 0; 0; 0; 0; 0];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                    Obsevability of Linearized System
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[t,y1] = ode45(@(t,y)(A-B*K)*y,tspan,s0);
figure;
hold on
plot(t,y1(:,1),'r')
plot(t,y1(:,3),'b')
plot(t,y1(:,5),'k')
ylabel('state variables')
xlabel('time in s')
title('Response of Linearized system with LQR based control')
legend('x position of the cart','theta1','theta2')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                    Obsevability of Original Non-Linear System
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

[t1,y2] = ode45(@(t,y)nlinear(y,t,m1,m2,M,l1,l2,g,-K*y),tspan,s0);
figure;
hold on
plot(t1,y2(:,1),'r')
plot(t1,y2(:,3),'b')
```

```
    plot(t1,y2(:,5),'k')
    ylabel('state variables')
    xlabel('time in s')
    title('Response of Non-Linear system with LQR based control')
    legend('x position of the cart','theta1','theta2')

    % Noise and Disturbances in the system
    Bd = 0.1*eye(6);                %input disturbance covarianve
    Bn = 0.1;
    Bn1 = 0;                        %output measurement noise
    Bn3 = 0*[0,1;0,1];
    Bn4 = 0*[0,0,1;0,0,1;0,0,1];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   Obtaining "best" Luenberger observer for each one of the output vectors using Kalman Bucy Filter (Lqe))
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% system
[L1,P,E] = lqe(A,Bd,C1,Bd,Bn);
[L3,P,E] = lqe(A,Bd,C3,Bd,Bn*eye(2));
[L4,P,E] = lqe(A,Bd,C4,Bd,Bn*eye(3));
%% Luenberger's Observer using Pole placement Method
%%
    Ae=[(A-B*K)];
    poles = eig(Ae)
    P = [-2 -5 -6 -7 -8 -9];
    L1p = place(A',C1',P)'
    L3p = place(A',C3',P)'
    C4p = [1,0,0,0,0,0;0,0,1,0,0,0;0,0,0,0,1,0]
    L4p = place(A',C4',P)';


    % Creating Augmented Matrices for Simulation
    uD = randn(6,size(tspan,2));        %input for disturbance
```

```
    uN = randn(size(tspan));            %input for noise
    u = 0*tspan;
    u(200:length(tspan)) = 1;        % Step input at t = 10
    u1 = [u; Bd*Bd*uD; uN];

    uDp = 0*randn(6,size(tspan,2));
    uNp = 0*randn(size(tspan));
    up = 0*tspan;
    up(200:length(tspan)) = 1;       % Step input at t = 10

    u1p = [up; Bd*Bd*uDp; uNp];

    Be = [B,Bd,zeros(size(B))];

%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Luenberger Observer output when X(t) is the output vector
%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%
    sysLO1 = ss(A-L1*C1,[B L1],C1,zeros(1,2));      %State
        Estimator system

    %Obtaining Y values for a system simulated with noise
        and disturbance.
    De1 = [0,0,0,0,0,0,0,Bn1];                              %
        Augmented D matrix

    sys1 = ss(A,Be,C1,De1)
    [y1,t] = lsim(sys1,u1,tspan);

    %Simulating the States of the output variables
    [x1,t] = lsim(sysLO1,[u; y1'],tspan);

    figure();
    hold on
    plot(t,y1(:,1),'g')
    plot(t,x1(:,1),'k--')
```

```
    ylabel('x-position of cart')
    xlabel('time in s')
    legend('Output obtained from noisy system','Estimated output of the system')
    title('Estimated Response for C1: output vector x(t) - Linear System')
    hold off

    opt = simset('solver','ode45','SrcWorkspace','Current');
    [tout2]=sim('nonlinerLO',tspan,opt);
    figure();
    hold on
    plot(tout2,out1(:,1),'r')
    plot(tout2,states1(:,1),'k--')
    ylabel('x-position of cart')
    xlabel('time in s')
    legend('Output obtained from noisy system','Estimated output of the system')
    title('Estimated Response for C1: output vector x(t) - Nonlinear System')
    hold off

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Luenberger Observer output when (X(t),theta2(t)) is the output vector
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    sysLO3 = ss(A-L3*C3,[B L3],C3,zeros(2,3))      %State Estimator system

   %Obtaining Y values for a system simulated with noise and disturbance.
    De3 = [zeros(size(C3)),Bn3];                     %Augmented D matrix

    sys3 = ss(A,Be,C3,De3)
    [y3,t] = lsim(sys3,u1,tspan);
```

```

%Simulating the States of the output variables
[x3,t] = lsim(sysLO3,[u; y3'],tspan);

figure();
hold on
plot(t,y3(:,1),'g')
plot(t,y3(:,2),'b')
plot(t,x3(:,1),'k--')
plot(t,x3(:,2),'r--')
ylabel('State Variables ')
xlabel('time in s')
legend('Noisy output x(t)','Noisy output theta2(t)','Estimated x(t)','Estimated theta2(t)')
title('Estimated Response for C3: output vector (x(t),t2(t))')
hold off

opt = simset('solver','ode45','SrcWorkspace','Current');
[tout3]=sim('nonlinearLO3',tspan,opt);
figure();
hold on
plot(tout3,out3(:,1),'r')
plot(tout3,out3(:,2),'g')
plot(t,states3(:,1),'k--')
plot(t,states3(:,2),'r--')
ylabel('x-position of cart')
xlabel('time in s')
legend('Output obtained from noisy system','Estimated output of the system')
title('Estimated Response for C3: output vector (x(t),t2(t)) - Nonlinear System')
hold off

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Luenberger Observer output when (X(t),theta1(t),theta2(t)) is the output vector
%
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
sysLO4 = ss(A-L4*C4,[B L4],C4,zeros(3,4))      %State Estimator system

%Obtaining Y values for a system simulated with noise and disturbance.
De4 = [zeros(3,5),Bn4];                         %Augmented D matrix

sys4 = ss(A,Be,C4,De4)
[y4,t] = lsim(sys4,u1,tspan);

%Simulating the States of the output variables
[x4,t] = lsim(sysLO4,[u;y4'],tspan);

figure();
hold on
plot(t,y4(:,1),'g')
plot(t,y4(:,2),'b')
plot(t,y4(:,3),'c')
plot(t,x4(:,1),'m--')
plot(t,x4(:,2),'r--')
plot(t,x4(:,3),'k--')
ylabel('State Variables ')
xlabel('time in s')
legend('Noisy output x(t)','Noisy output theta1(t)','Noisy output theta2(t)','Estimated x(t)','Estimated theta1(t)','Estimated theta2(t)')
title('Estimated Response for C4: output vector (x(t),t1(t),t2(t))')
hold off

opt = simset('solver','ode45','SrcWorkspace','Current');
[tout4]=sim('nonlinearLO4',tspan,opt);

figure();
hold on
plot(tout4,out4(:,1),'g')
plot(tout4,out4(:,2),'b')
```

```
    plot(tout4,out4(:,3),'c')
    plot(tout4,states4(:,1),'m--')
    plot(tout4,states4(:,2),'r--')
    plot(tout4,states4(:,3),'k--')
    ylabel('State Variables ')
    xlabel('time in s')
    legend('Noisy output x(t)','Noisy output theta1(t)','Noisy output theta2(t)','Estimated x(t)','Estimated theta1(t)','Estimated theta2(t)')
    title('Estimated Response for C4: output vector (x(t),t1(t),t2(t))')
    hold off

%%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% LQG Controller for smallest Output Vector C1 = [1,0,0,0,0,0]
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    Ac = A-L1p*C1;
    Bc = [B L1p];
    Cc = eye(6);
    Dc = 0*[B L1p];

    opt = simset('solver','ode45','SrcWorkspace','Current');
    sim('nonlinearlqg',tspan,opt);
    %% Simulation Results
    %%
    figure();
    hold on
    plot(tout,states(:,1),'r')
    plot(tout,states(:,3),'k')
    plot(tout,states(:,5),'b')
    plot(tout,inputlqg(:,1),'g')

    title('LQG Nonlinear System')
    legend('x-position','theta1','theta2')
```

```
    hold off
        c = [B A*B A^2*B A^3*B A^4*B A^5*B];
    end

function [Af,Bf] = mat(a,b,c,d,e,f)
    syms m1 m2 g M l1 l2 F t1 t2 t1_dot t2_dot x x_dot
    A = (F - m1*g*sin(2*t1)/2 - m2*g*sin(2*t2)/2 - m1*l1*sin
        (t1)*(t1_dot)^2 - m2*l2*sin(t2)*(t2_dot)^2)/(M + m1*(
        sin(t1)^2) + m2*(sin(t2)^2));
    B = (1/l1)*(cos(t1)*(F - m1*g*sin(2*t1)/2 - m2*g*sin(2*
        t2)/2 - m1*l1*sin(t1)*(t1_dot)^2 - m2*l2*sin(t2)*(
        t2_dot)^2)/(M + m1*(sin(t1)^2) + m2*(sin(t2)^2)) - g*
        sin(t1));
    C = (1/l2)*(cos(t2)*(F - m1*g*sin(2*t1)/2 - m2*g*sin(2*
        t2)/2  - m1*l1*sin(t1)*(t1_dot)^2 - m2*l2*sin(t2)*(
        t2_dot)^2)/(M + m1*(sin(t1)^2) + m2*(sin(t2)^2)) - g*
        sin(t2));
    f1x1 = diff(A,x);
    f1x1_dot = diff(A,x_dot);
    f1t1 = diff(A,t1);
    f1t1_dot = diff(A,t1_dot);
    f1t2 = diff(A,t2);
    f1t2_dot = diff(A,t2_dot);
    f2x1 = diff(B,x);
    f2x1_dot = diff(B,x_dot);
    f2t1 = diff(B,t1);
    f2t1_dot = diff(B,t1_dot);
    f2t2 = diff(B,t2);
    f2t2_dot = diff(B,t2_dot);
    f3x1 = diff(C,x);
    f3x1_dot = diff(C,x_dot);
    f3t1 = diff(C,t1);
    f3t1_dot = diff(B,t1_dot);
    f3t2 = diff(C,t2);
    f3t2_dot = diff(C,t2_dot);

    A11 = subs(f1x1, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    A12 = subs(f1t1, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    A13 = subs(f1t2, {x,x_dot,t1,t1_dot,t2,t2_dot},
```

```
    {0,0,0,0,0,0});
    A14 = subs(f1x1_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    A15 = subs(f1t1_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    A16 = subs(f1t2_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    A21 = subs(f2x1, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    A22 = subs(f2t1, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    A23 = subs(f2t2, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    A24 = subs(f2x1_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    A25 = subs(f2t1_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    A26 = subs(f2t2_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    A31 = subs(f3x1, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    A32 = subs(f3t1, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    A33 = subs(f3t2, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    A34 = subs(f3x1_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    A35 = subs(f3t1_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    A36 = subs(f3t2_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});

    f2F = diff(A,F);
    f4F = diff(B,F);
    f6F = diff(C,F);
    B11 = subs(f2F, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    B12 = subs(f4F, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
    B13 = subs(f6F, {x,x_dot,t1,t1_dot,t2,t2_dot},
```

```
        {0,0,0,0,0,0});
    B1 = simplify([0;B11;0;B12;0;B13])
    A1 = simplify([0 1 0 0 0 0; A11 A14 A12 A15 A13 A16;0 0
        0 1 0 0; A21 A24 A22 A25 A23 A26;0 0 0 0 0 1;A31 A34
        A32 A35 A33 A36])
    c =  simplify(simplify([B1 A1*B1 A1^2*B1 A1^3*B1 A1^4*B1
         A1^5*B1]))
    B1
    A1*B1
    A1^2*B1
    A1^3*B1
    A1^4*B1
    A1^5*B1
    det(c)
    %Bf = (subs(B1, {m1,m2,M,l1,l2,g},{a,b,c,d,e,f}))
    Bf = B1
    %Af = (subs(A1, {m1,m2,M,l1,l2,g},{a,b,c,d,e,f}))
    Af = A1
end
```