



ENPM667(CONTROLS)

FORMAL REPORT

PROJECT

Non-linear Control

Author:

Akwasi A. Obeng

Eashwar Sathyamurthy

UID:

117000801

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

$$V_{m_1} = -m_1 * l_1 \cos(\theta_1(t)) \quad (1)$$

$$V_{m_2} = -m_2 * l_2 \cos(\theta_2(t)) \quad (2)$$

$$V_M = 0 \quad (3)$$

$$V = V_{m_1} + V_{m_2} + V_M \quad (4)$$

Kinetic Energy

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

$$T_{m_1} = \frac{1}{2}m_1\left(\frac{dx(t)}{dt}\right)^2 + \frac{1}{2}m_1l_1^2\left(\frac{d\theta_1}{dt}\right)^2 - m_1l_1 \cos \theta_1(t) \frac{d\theta_1}{dt} \frac{dx}{dt} \quad (5)$$

$$T_{m_2} = \frac{1}{2}m_2\left(\frac{dx(t)}{dt}\right)^2 + \frac{1}{2}m_2l_2^2\left(\frac{d\theta_2}{dt}\right)^2 - m_2l_2 \cos \theta_2(t) \frac{d\theta_2}{dt} \frac{dx}{dt} \quad (6)$$

$$T_M = \frac{1}{2}M\left(\frac{dx}{dt}\right)^2 \quad (7)$$

$$T = T_{m_1} + T_{m_2} + T_M \quad (8)$$

Lagrangian

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

$$L = T - V \quad (9)$$

$$\begin{aligned} &= \frac{1}{2}m_1\left(\frac{dx(t)}{dt}\right)^2 + \frac{1}{2}m_1l_1^2\left(\frac{d\theta_1}{dt}\right)^2 - m_1l_1 \cos \theta_1(t) \frac{d\theta_1}{dt} \frac{dx}{dt} + \\ &\quad \frac{1}{2}m_2\left(\frac{dx(t)}{dt}\right)^2 + \frac{1}{2}m_2l_2^2\left(\frac{d\theta_2}{dt}\right)^2 - m_2l_2 \cos \theta_2(t) \frac{d\theta_2}{dt} \frac{dx}{dt} + \\ &\quad \frac{1}{2}M\left(\frac{dx}{dt}\right)^2 + m_1 * l_1 \cos(\theta_1(t)) + m_2 * l_2 \cos(\theta_2(t)) \end{aligned} \quad (10)$$

1.2 Equation of Motion

$$0 = -m_1 l_1 \frac{d^2 x}{dt^2} \cos \theta_1(t) + m_1 l_1^2 \frac{d^2 \theta_1}{dt^2} + m g l_1 \sin \theta_1(t) \quad (11)$$

$$0 = -m_2 l_2 \frac{d^2 x}{dt^2} \cos \theta_2(t) + m_2 l_2^2 \frac{d^2 \theta_2}{dt^2} + m g l_2 \sin \theta_2(t) \quad (12)$$

$$F = \frac{d^2 x(t)}{dt^2} [M + m_1 + m_2] - m_1 l_1 \left[\frac{d^2 \theta_1(t)}{dt^2} \cos(\theta_1(t)) - \sin \theta_1(t) \left(\frac{d\theta_1}{dt} \right)^2 \right] \\ - m_2 l_2 \left[\frac{d^2 \theta_2}{dt^2} \cos \theta_2(t) - \sin \theta_2(t) \left(\frac{d\theta_2}{dt} \right)^2 \right]. \quad (13)$$

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

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

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

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

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

2 Linearization (Question B)

A non linear system, $F(X, U)$ with state space equation $\frac{dX}{dt} = F(X, U)$ can be linearized with the equation stated below.

$$\delta_x = A_F \frac{dx(t)}{dt} + B_F \frac{du(t)}{dt} \quad (17)$$

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.

$$X = \left[x \quad \frac{dx}{dt} \quad \theta_1 \quad \frac{d\theta_1}{dt} \quad \theta_2 \quad \frac{d\theta_2}{dt} \right]^T \quad (18)$$

Therefore, the state space equation can be written as

$$\frac{dX}{dt} = \frac{d}{dt} \begin{bmatrix} x \\ \frac{dx}{dt} \\ \theta_1 \\ \frac{d\theta_1}{dt} \\ \theta_2 \\ \frac{d\theta_2}{dt} \end{bmatrix} = \begin{bmatrix} f_1(X, U) \\ f_2(X, U) \\ f_3(X, U) \\ f_4(X, U) \\ f_5(X, U) \\ f_6(X, U) \end{bmatrix} = F \quad (19)$$

Based on equation above, the following can be noted

$$f_1(X, U) = \frac{dx}{dt} \quad (20)$$

$$f_2(X, U) = \frac{d^2x}{dt^2} \quad \text{Refer to Dynamics for equation} \quad (21)$$

$$f_3(X, U) = \frac{d\theta}{dt} \quad (22)$$

$$f_4(X, U) = \frac{d^2\theta_1}{dt^2} \quad \text{Refer to Dynamics for equation} \quad (23)$$

$$f_5(X, U) = \frac{d\theta_2}{dt} \quad (24)$$

$$f_6(x, U) = \frac{d^2\theta_2}{dt^2} \quad \text{Refer to Dynamics for equation} \quad (25)$$

$$(26)$$

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

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

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 1: lagrange Dynamics

3 Controllability (Question C)

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

$$\text{Det}(C) \neq 0 \quad (28)$$

Condition for Controllability

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

$$\text{Det}(C) = -\frac{g^6}{M^6 l_1^6 l_2^6} (l_1 - l_2)^2 \quad (30)$$

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

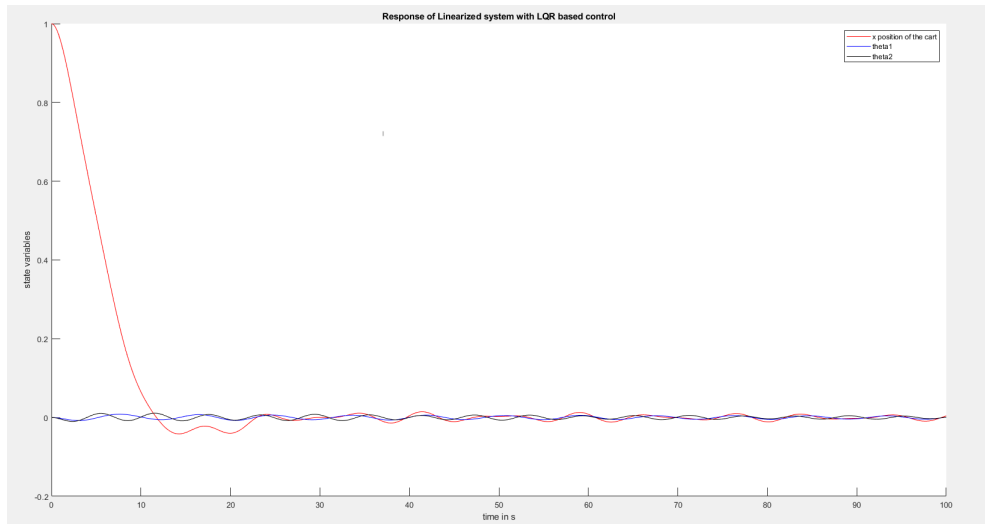


Figure 2: Linearized System with Feedback

LQR Non-Linear System

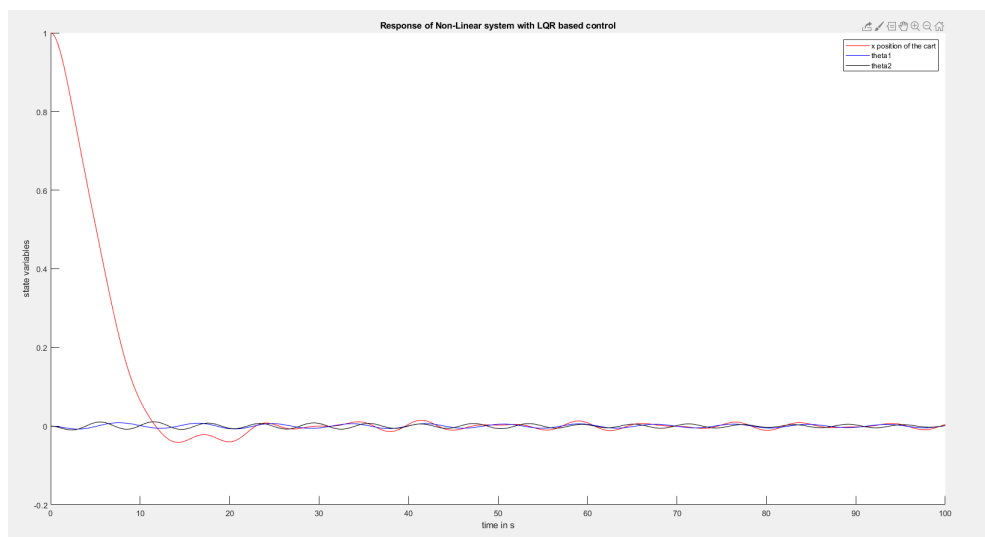


Figure 3: LQR Non linear System

Refer to code for information on how these were obtained

5 Observability (Question E)

Observability $x(t)$

```
ob1 =
    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    1.0000    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   -1.0000   -0.1000    0    0    0.6500
    0    1.0000    0    0   -1.0000   -1.1000    0    0    1.1500    1.2150    0
    0    0    0    1.0000    0    0   -1.0000   -1.1000    0    0    1.1500
```

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

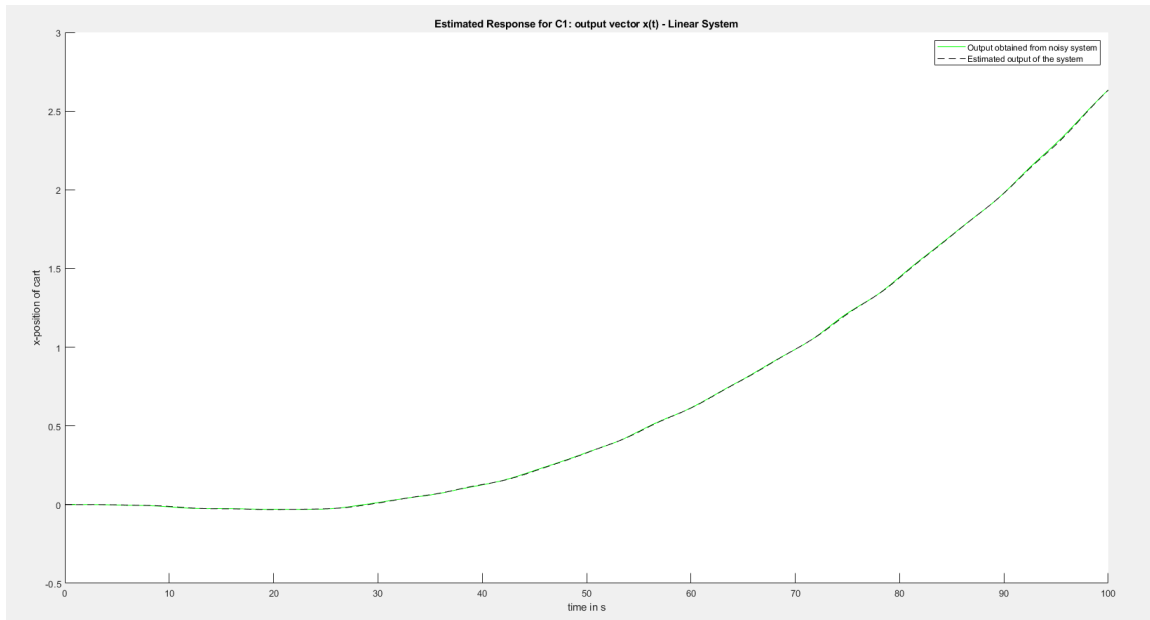
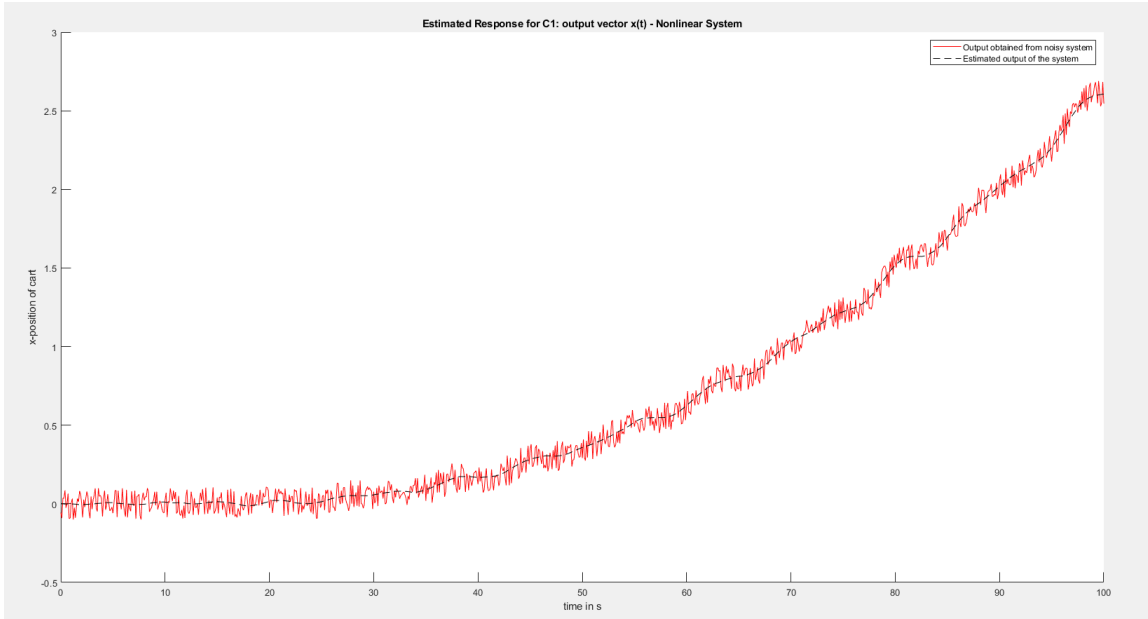
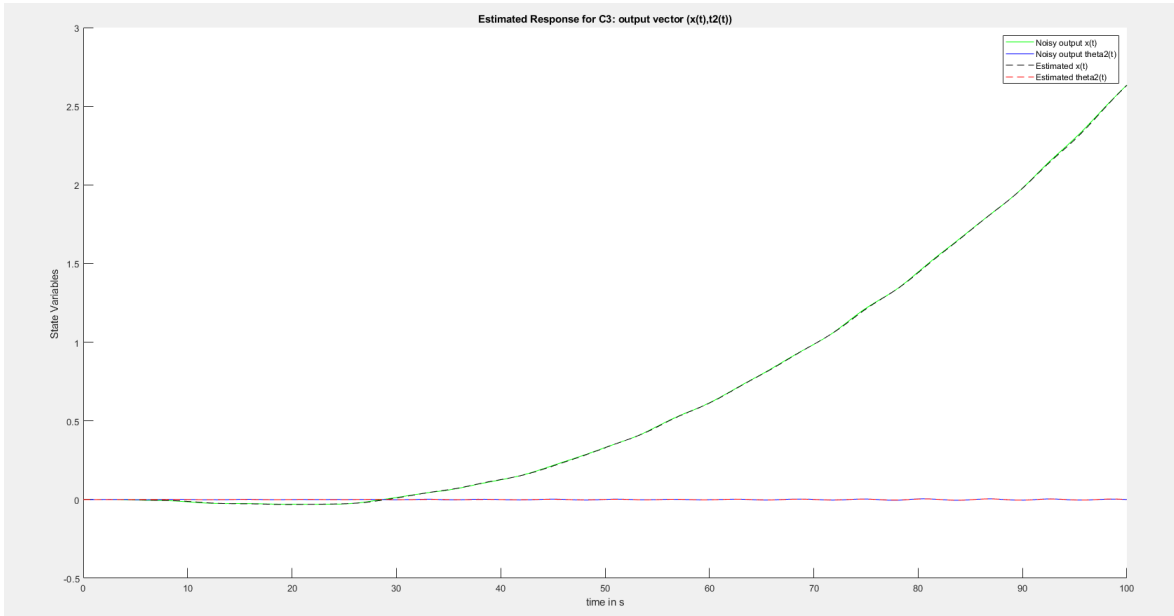
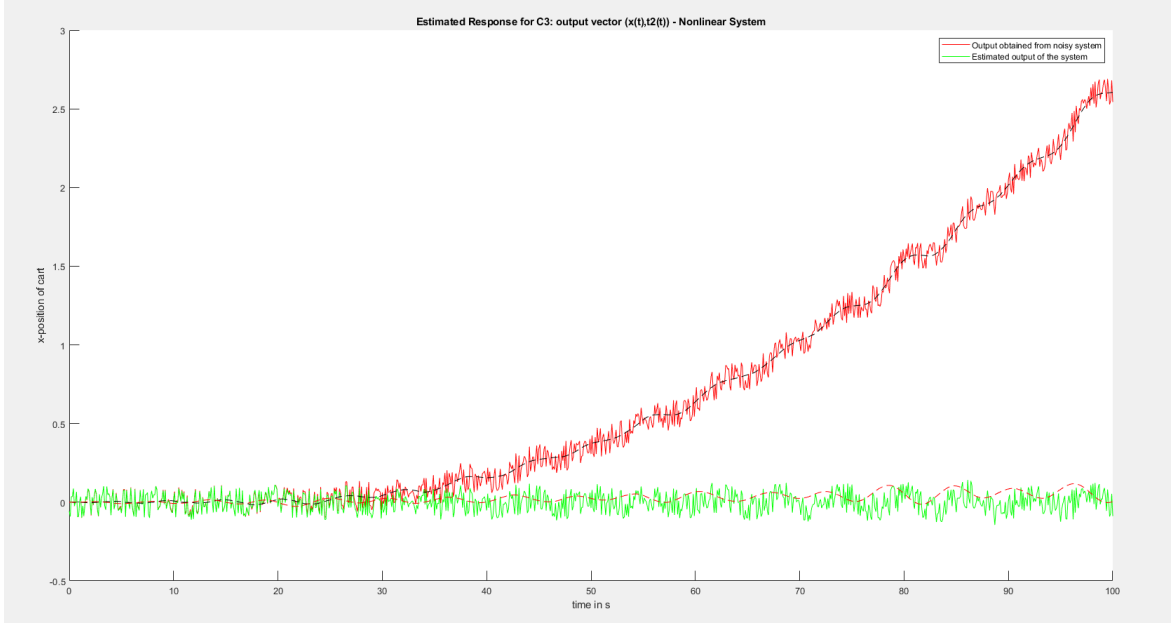
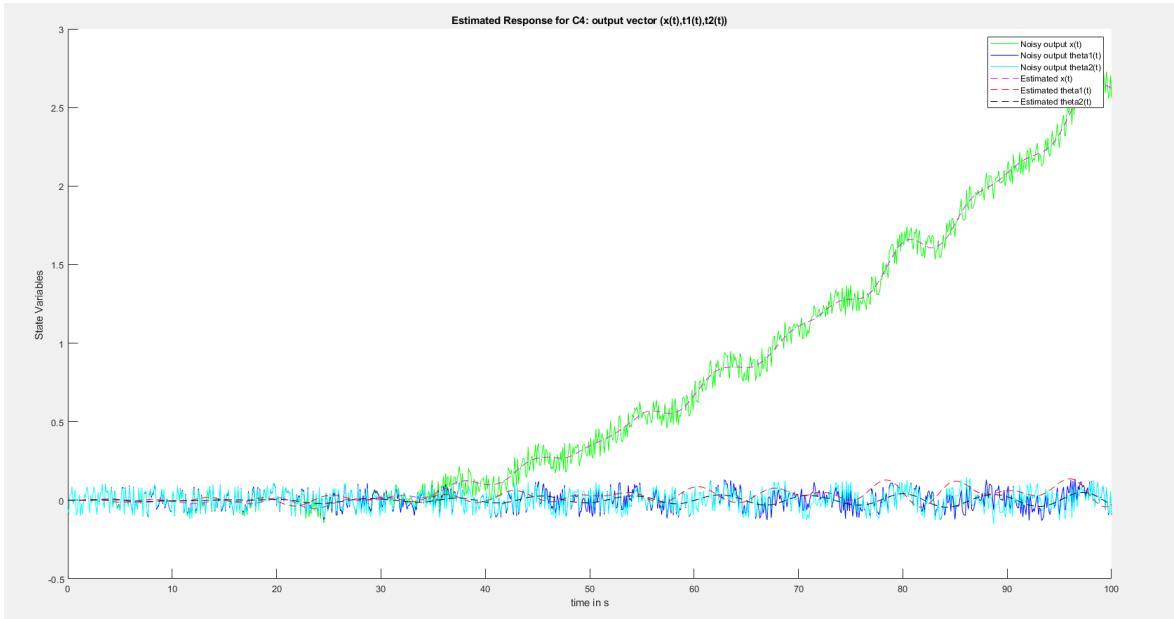
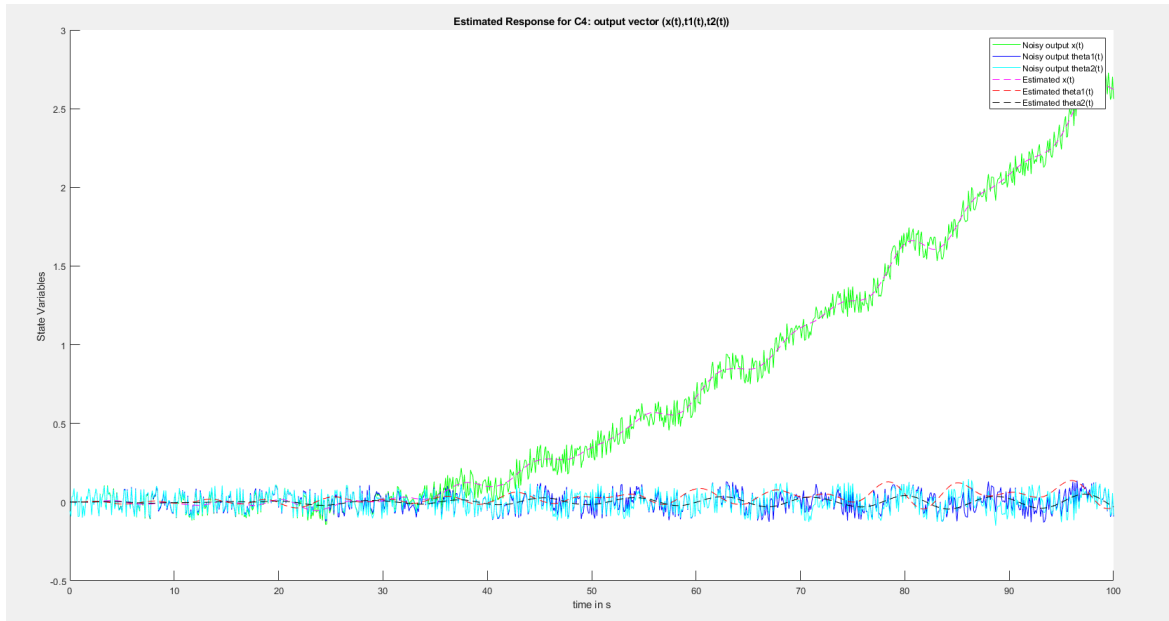


Figure 4: State $x(t)$ Linear Response

State $x(t)$ Non Linear ResponseFigure 5: State $x(t)$ Non Linear ResponseState $x(t)$ and $\theta_2(t)$ Linear ResponseFigure 6: State $x(t)$ and $\theta_2(t)$ Linear Response

State $x(t)$ and $\theta_2(t)$ Non-Linear ResponseFigure 7: State $x(t)$ and $\theta_2(t)$ Non-Linear Response**State $x(t), \theta_1(t)$ and $\theta_2(t)$ Linear Response**Figure 8: State $x(t), \theta_1(t)$ and $\theta_2(t)$ Linear Response

State $x(t), \theta_1(t)$ and $\theta_2(t)$ Non-Linear ResponseFigure 9: State $x(t), \theta_1(t)$ and $\theta_2(t)$ Non-Linear Response

Refer to code for information on how these were obtained

7 LQG Non linear

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

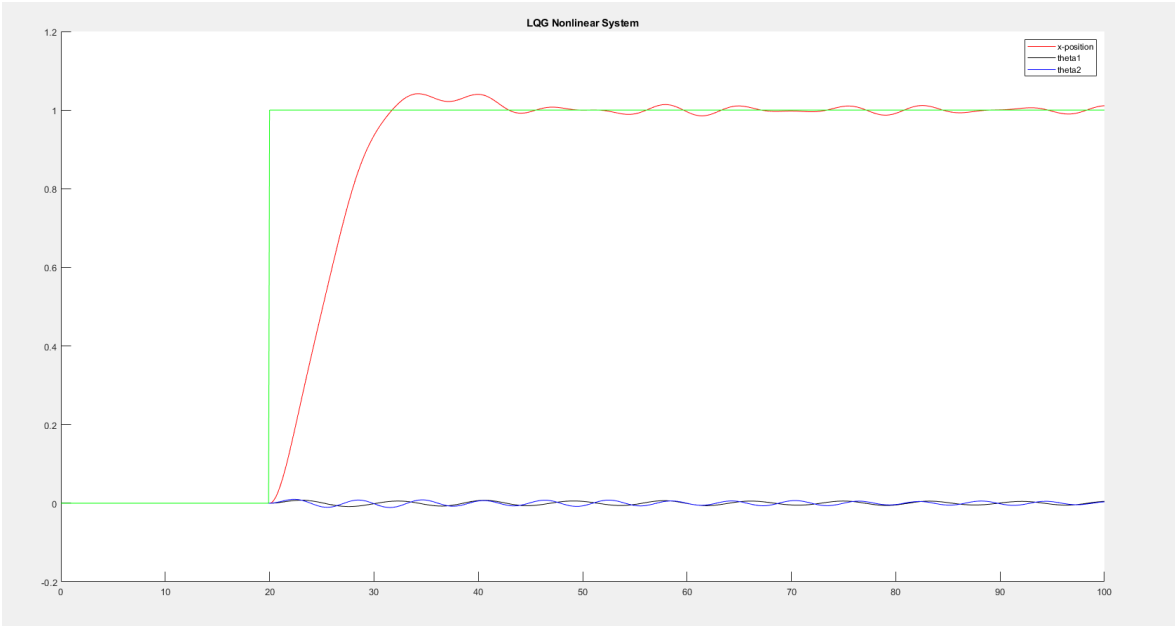


Figure 10: LQG Non linear

Refer to Code for information on how this was achieved

8 Appendix

8.1 Code for Dynamics Calculations

```
1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
3 %                                PROBLEM 3
4 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
5 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```

5 %
  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

6
7 % Note This approach doesn't account for Rotational Inertia '
8
9 syms I m1 m2 M g F1 F2 F3 F
10 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)
11 syms l1 l2
12 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
13 % Symbol definitions
14 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
15 % q(t) - Represents generalized coordinates
16     % q1 represents theta 1
17     % q2 represents theta 2
18     % q3 represents x(t)
19 % F - Represents generalized force or torque
20 % q_d(t) - Represents derivative of q wrt to t
21 % q_dd(t)- Represents second derivative of q wrt to t
22 % r - Position of COM wrt to inertial(world) frame
23 % a's - robot link parameters
24 % I - Rotational Inertia from links
25
26 %
  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

27 %                               Lagrangian
28 %
  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

29 r1 = [q3-l1*sin(q1)
30       -l1*cos(q1)];
31
32 r2 = [q3-l2*sin(q2)
33       -l2*cos(q2)];
34
35 r3 = q3;
36
37 % Getting velocities

```



```

38 v1 = diff(r1);
39 v2 = diff(r2);
40 v3 = diff(r3);
41
42 % Kinetic Energy
43 T1 = 0.5 * transpose(v1) * v1*m1;
44 T2 = 0.5 * transpose(v2) * v2*m2;
45 T3 = 0.5 * transpose(v3) * v3*M;
46
47 T = T1 + T2 + T3;
48 T=simplify(T);
49
50 % Potential Energy
51 V1 = -m1*g*l1*cos(q1);
52 V2 = -m2*g*l2*cos(q2);
53 V3 = 0;
54
55 V = V1 + V2 + V3;
56 V=simplify(V)
57
58 % Lagrangian
59 L = T-V;
60 L = simplify(L)
61
62 %
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
63 %
64 %
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

65
66 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
67 %      Derivative of L wrt to q
68 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
69
70 L=subs(L, diff(q1(t),t), q1_d);
71 L=subs(L, diff(q2(t),t), q2_d);
72 L=subs(L, diff(q3(t),t), q3_d);
73

```

```

74 dL_dq1 = functionalDerivative(L,q1);
75 dL_dq2 = functionalDerivative(L,q2);
76 dL_dq3 = functionalDerivative(L,q3);
77
78
79 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
80 %   Derivative of L wrt q_dot
81 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
82
83 % Replacing derivatives with symbols
84 % Matlab can't take functional derivative with
85 % respect to a derivative
86 dL_dq1_dot= functionalDerivative(L,q1_d);
87 dL_dq2_dot= functionalDerivative(L,q2_d);
88 dL_dq3_dot= functionalDerivative(L,q3_d);
89
90 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
91 %Derivative of dl/dq_dot wrt to t
92 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
93 %Taking derivative of dL_dq with respect to t
94 dt_dL_dq1_dot = diff(dL_dq1_dot,t);
95 dt_dL_dq2_dot = diff(dL_dq2_dot,t);
96 dt_dL_dq3_dot = diff(dL_dq3_dot,t);
97
98
99 %
100 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
101 %
102 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
103 %Repeating derivatives with symbols
104 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
105
106
107 %Replacing derivatives with symbols to perform substaction
108 dt_dL_dq1_dot=subs(dt_dL_dq1_dot,diff(q1_d(t),t),q1_dd);
109 dt_dL_dq1_dot=subs(dt_dL_dq1_dot,diff(q1(t),t),q1_d);

```

```

110 dt_dL_dq1_dot=subs(dt_dL_dq1_dot , diff(q2_d(t),t) , q2_dd);
111 dt_dL_dq1_dot=subs(dt_dL_dq1_dot , diff(q2(t),t) , q2_d);
112 dt_dL_dq1_dot=subs(dt_dL_dq1_dot , diff(q3_d(t),t) , q3_dd);
113 dt_dL_dq1_dot=subs(dt_dL_dq1_dot , diff(q3(t),t) , q3_d);
114
115 dt_dL_dq2_dot=subs(dt_dL_dq2_dot , diff(q1_d(t),t) , q1_dd);
116 dt_dL_dq2_dot=subs(dt_dL_dq2_dot , diff(q1(t),t) , q1_d);
117 dt_dL_dq2_dot=subs(dt_dL_dq2_dot , diff(q2_d(t),t) , q2_dd);
118 dt_dL_dq2_dot=subs(dt_dL_dq2_dot , diff(q2(t),t) , q2_d);
119 dt_dL_dq2_dot=subs(dt_dL_dq2_dot , diff(q3_d(t),t) , q3_dd);
120 dt_dL_dq2_dot=subs(dt_dL_dq2_dot , diff(q3(t),t) , q3_d);
121
122 dt_dL_dq3_dot=subs(dt_dL_dq3_dot , diff(q1_d(t),t) , q1_dd);
123 dt_dL_dq3_dot=subs(dt_dL_dq3_dot , diff(q1(t),t) , q1_d);
124 dt_dL_dq3_dot=subs(dt_dL_dq3_dot , diff(q2_d(t),t) , q2_dd);
125 dt_dL_dq3_dot=subs(dt_dL_dq3_dot , diff(q2(t),t) , q2_d);
126 dt_dL_dq3_dot=subs(dt_dL_dq3_dot , diff(q3_d(t),t) , q3_dd);
127 dt_dL_dq3_dot=subs(dt_dL_dq3_dot , diff(q3(t),t) , q3_d);
128
129
130 %Replacing derivatives with symbols to perform substaction
131
132
133 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
134 %           Finding Equations
135 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
136 eqn1 = simplify(dt_dL_dq1_dot - dL_dq1) == 0
137 eqn2 = simplify(dt_dL_dq2_dot - dL_dq2) == 0
138 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 11: lagrange Dynamics

8.2 Code for LQR,LQG and Observability

```

1 function c = compute(m1,m2,M,l1 ,l2 ,g)
2     syms t
3     %
4     %                                     LQR CONTROLLER
5     %
6     %[A,B] = mat(m1,m2,M,l1 ,l2 ,g);
7
8     A = [0,1.0000,0,0,0,0;
9          0,0,-1.0000,0,-1.0000,0;
10         0,0,0,1.0000,0,0;
11         0,0,-0.5500,0,-0.0500,0;
12         0,0,0,0,0,1.0000;
13         0,0,-0.1000,0,-1.1000,0];

```

```

14     B = [0;0.001;0;0.00005;0;0.0001];
15
16     Eigen = eig(A)
17     disp('The eigen values of A ')
18     disp('therefore Lyapunovs indirect method is
19         inconclusive for the system')
20     tspan = 0:0.1:100;
21     Q = 100*eye(6);
22     R = 0.01;
23     [K,P,E] = lqr(double(A),double(B),Q,R)
24     eigen = eig(A-B*K)
25     C1 = [1 0 0 0 0 0] %For output (x(t))
26     C2 = [0 0 1 0 0 0; %For output (theta1(t),
27         theta2(t))
28         0 0 1 0 1 0]
29     C3 = [1 0 0 0 0 0; %For output (x(t),theta2(t)
30         ))
31         0 0 0 0 1 0]
32     C4 = [1 0 0 0 0 0; %For output (x(t),theta1(
33         t),theta2(t))
34         0 0 1 0 0 0;
35         0 0 0 0 1 0]
36
37 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
38
39 %Checking Observability
40 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
41
42 ob1 = [C1;C1*A;C1*A^(2);C1*A^(3);C1*A^(4);C1*A^(5)]'
43 ob2 = [C2;C2*A;C2*A^(2);C2*A^(3);C2*A^(4);C2*A^(5)]'
44 ob3 = [C3;C3*A;C3*A^(2);C3*A^(3);C3*A^(4);C3*A^(5)]'
45 ob4 = [C4;C4*A;C4*A^(2);C4*A^(3);C4*A^(4);C4*A^(5)]'
46
47 observability_of_C1 = rank(ob1)
48 disp('Rank is 6. The system is observable for output x(t)
49     ')
50 observability_of_C2 = rank(ob2)
51 disp('Rank is 4. The system is not observable for output

```

```

        (t1(t),t2(t))')
45     observability_of_C3 = rank(ob3)
46     disp('The system is observable for output (x(t),t2(t))')
47     observability_of_C4 = rank(ob4)
48     disp('Rank is 6.The system is observable for output (x(t
        ),t1(t),t2(t))')
49     s0 = [1; 0; 0; 0; 0; 0];
50 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

51 %                               Obsevability of Linearized System
52 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

53     [t,y1] = ode45(@(t,y)(A-B*K)*y,tspan,s0);
54     figure;
55     hold on
56     plot(t,y1(:,1),'r')
57     plot(t,y1(:,3),'b')
58     plot(t,y1(:,5),'k')
59     ylabel('state variables')
60     xlabel('time in s')
61     title('Response of Linearized system with LQR based
        control')
62     legend('x position of the cart','theta1','theta2')
63
64 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

65 %                               Obsevability of Original Non-Linear System
66 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

67
68     [t1,y2] = ode45(@(t,y) nlinear(y,t,m1,m2,M,l1,l2,g,-K*y),
        tspan,s0);
69     figure;
70     hold on
71     plot(t1,y2(:,1),'r')
72     plot(t1,y2(:,3),'b')

```

```

73     plot(t1,y2(:,5),'k')
74     ylabel('state variables')
75     xlabel('time in s')
76     title('Response of Non-Linear system with LQR based
           control')
77     legend('x position of the cart','theta1','theta2')
78
79     % Noise and Disturbances in the system
80     Bd = 0.1*eye(6);           %input disturbance
           covarianve
81     Bn = 0.1;
82     Bn1 = 0;                   %output measurement noise
83     Bn3 = 0*[0,1;0,1];
84     Bn4 = 0*[0,0,1;0,0,1;0,0,1];
85
86     %
           %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
87     %   Obtaining "best" Luenberger observer for each one of the
           output vectors using Kalman Bucy Filter (Lqe))
88     %
           %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

89     % system
90     [L1,P,E] = lqe(A,Bd,C1,Bd,Bn);
91     [L3,P,E] = lqe(A,Bd,C3,Bd,Bn*eye(2));
92     [L4,P,E] = lqe(A,Bd,C4,Bd,Bn*eye(3));
93     %% Luenberger's Observer using Pole placement Method
94     %%
95     Ae=[(A-B*K)];
96     poles = eig(Ae)
97     P = [-2 -5 -6 -7 -8 -9];
98     L1p = place(A',C1',P)';
99     L3p = place(A',C3',P)';
100    C4p = [1,0,0,0,0,0;0,0,1,0,0,0;0,0,0,0,1,0]
101    L4p = place(A',C4',P)';
102
103
104    % Creating Augmented Matrices for Simulation
105    uD = randn(6,size(tspan,2));           %input for disturbance

```

```

106     uN = randn(size(tspan));           %input for noise
107     u = 0*tspan;
108     u(200:length(tspan)) = 1;         % Step input at t = 10
109     u1 = [u; Bd*Bd*uD; uN];
110
111     uDp = 0*randn(6, size(tspan,2));
112     uNp = 0*randn(size(tspan));
113     up = 0*tspan;
114     up(200:length(tspan)) = 1;         % Step input at t = 10
115
116     u1p = [up; Bd*Bd*uDp; uNp];
117
118     Be = [B,Bd, zeros(size(B))];
119
120     %
121     %% Luenberger Observer output when X(t) is the output vector
122     %
123     %%
124     sysLO1 = ss(A-L1*C1,[B L1],C1,zeros(1,2)); %State
125                                     Estimator system
126
127     %Obtaining Y values for a system simulated with noise
128     and disturbance.
129     De1 = [0,0,0,0,0,0,0,Bn1]; %
130     Augmented D matrix
131
132     sys1 = ss(A,Be,C1,De1)
133     [y1,t] = lsim(sys1,u1,tspan);
134
135     %Simulating the States of the output variables
136     [x1,t] = lsim(sysLO1,[u; y1'],tspan);
137
138     figure();
139     hold on
140     plot(t,y1(:,1),'g')
141     plot(t,x1(:,1),'k—')

```



```

139     ylabel('x-position of cart')
140     xlabel('time in s')
141     legend('Output obtained from noisy system','Estimated
           output of the system')
142     title('Estimated Response for C1: output vector x(t) -
           Linear System')
143     hold off
144
145     opt = simset('solver','ode45','SrcWorkspace','Current');
146     [tout2]=sim('nonlinerLO',tspan,opt);
147     figure();
148     hold on
149     plot(tout2,out1(:,1),'r')
150     plot(tout2,states1(:,1),'k—')
151     ylabel('x-position of cart')
152     xlabel('time in s')
153     legend('Output obtained from noisy system','Estimated
           output of the system')
154     title('Estimated Response for C1: output vector x(t) -
           Nonlinear System')
155     hold off
156
157 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
158 %% Luenberger Observer output when (X(t),theta2(t)) is the
           output vector
159 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
160     sysLO3 = ss(A-L3*C3,[B L3],C3,zeros(2,3))    %State
           Estimator system
161
162     %Obtaining Y values for a system simulated with noise
           and disturbance.
163     De3 = [zeros(size(C3)),Bn3];                  %
           Augmented D matrix
164
165     sys3 = ss(A,Be,C3,De3)
166     [y3,t] = lsim(sys3,u1,tspan);

```

```

167
168 %Simulating the States of the output variables
169 [x3,t] = lsim(sysLO3,[u; y3'],tspan);
170
171 figure();
172 hold on
173 plot(t,y3(:,1),'g')
174 plot(t,y3(:,2),'b')
175 plot(t,x3(:,1),'k—')
176 plot(t,x3(:,2),'r—')
177 ylabel('State Variables')
178 xlabel('time in s')
179 legend('Noisy output x(t)','Noisy output theta2(t)','
        Estimated x(t)','Estimated theta2(t)')
180 title('Estimated Response for C3: output vector (x(t),t2
        (t))')
181 hold off
182
183 opt = simset('solver','ode45','SrcWorkspace','Current');
184 [tout3]=sim('nonlinearLO3',tspan,opt);
185 figure();
186 hold on
187 plot(tout3,out3(:,1),'r')
188 plot(tout3,out3(:,2),'g')
189 plot(t,states3(:,1),'k—')
190 plot(t,states3(:,2),'r—')
191 ylabel('x-position of cart')
192 xlabel('time in s')
193 legend('Output obtained from noisy system','Estimated
        output of the system')
194 title('Estimated Response for C3: output vector (x(t),t2
        (t)) – Nonlinear System')
195 hold off
196
197 %
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
198 %% Luenberger Observer output when (X(t),theta1(t),
        theta2(t)) is the output vector
199 %

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
200     sysLO4 = ss(A-L4*C4,[B L4],C4,zeros(3,4))      %State
        Estimator system
201
202     %Obtaining Y values for a system simulated with noise
        and disturbance.
203     De4 = [zeros(3,5),Bn4];                        %Augmented D
        matrix
204
205     sys4 = ss(A,Be,C4,De4)
206     [y4,t] = lsim(sys4,u1,tspan);
207
208     %Simulating the States of the output variables
209     [x4,t] = lsim(sysLO4,[u;y4'],tspan);
210
211     figure();
212     hold on
213     plot(t,y4(:,1),'g')
214     plot(t,y4(:,2),'b')
215     plot(t,y4(:,3),'c')
216     plot(t,x4(:,1),'m—')
217     plot(t,x4(:,2),'r—')
218     plot(t,x4(:,3),'k—')
219     ylabel('State Variables ')
220     xlabel('time in s')
221     legend('Noisy output x(t)','Noisy output theta1(t)','
        Noisy output theta2(t)','Estimated x(t)','Estimated
        theta1(t)','Estimated theta2(t)')
222     title('Estimated Response for C4: output vector (x(t),t1
        (t),t2(t))')
223     hold off
224
225     opt = simset('solver','ode45','SrcWorkspace','Current');
226     [tout4]=sim('nonlinearLO4',tspan,opt);
227
228     figure();
229     hold on
230     plot(tout4,out4(:,1),'g')
231     plot(tout4,out4(:,2),'b')

```

```

232     plot(tout4,out4(:,3),'c')
233     plot(tout4,states4(:,1),'m—')
234     plot(tout4,states4(:,2),'r—')
235     plot(tout4,states4(:,3),'k—')
236     ylabel('State Variables')
237     xlabel('time in s')
238     legend('Noisy output x(t)', 'Noisy output theta1(t)', '
        Noisy output theta2(t)', 'Estimated x(t)', 'Estimated
        theta1(t)', 'Estimated theta2(t)')
239     title('Estimated Response for C4: output vector (x(t),t1
        (t),t2(t))')
240     hold off
241
242     %%
243     %
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

244     %% LQG Controller for smallest Output Vector C1 =
        [1,0,0,0,0,0]
245     %
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

246     Ac = A-L1p*C1;
247     Bc = [B L1p];
248     Cc = eye(6);
249     Dc = 0*[B L1p];
250
251     opt = simset('solver','ode45','SrcWorkspace','Current');
252     sim('nonlinearlqg',tspan,opt);
253     %% Simulation Results
254     %%
255     figure();
256     hold on
257     plot(tout,states(:,1),'r')
258     plot(tout,states(:,3),'k')
259     plot(tout,states(:,5),'b')
260     plot(tout,inputlqg(:,1),'g')
261
262     title('LQG Nonlinear System')
263     legend('x-position','theta1','theta2')

```

```

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

1 function [Af,Bf] = mat(a,b,c,d,e,f)
2     syms m1 m2 g M l1 l2 F t1 t2 t1_dot t2_dot x x_dot
3     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));
4     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));
5     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));
6     f1x1 = diff(A,x);
7     f1x1_dot = diff(A,x_dot);
8     f1t1 = diff(A,t1);
9     f1t1_dot = diff(A,t1_dot);
10    f1t2 = diff(A,t2);
11    f1t2_dot = diff(A,t2_dot);
12    f2x1 = diff(B,x);
13    f2x1_dot = diff(B,x_dot);
14    f2t1 = diff(B,t1);
15    f2t1_dot = diff(B,t1_dot);
16    f2t2 = diff(B,t2);
17    f2t2_dot = diff(B,t2_dot);
18    f3x1 = diff(C,x);
19    f3x1_dot = diff(C,x_dot);
20    f3t1 = diff(C,t1);
21    f3t1_dot = diff(B,t1_dot);
22    f3t2 = diff(C,t2);
23    f3t2_dot = diff(C,t2_dot);
24
25    A11 = subs(f1x1, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
26    A12 = subs(f1t1, {x,x_dot,t1,t1_dot,t2,t2_dot},
        {0,0,0,0,0,0});
27    A13 = subs(f1t2, {x,x_dot,t1,t1_dot,t2,t2_dot},

```

```

    {0,0,0,0,0,0});
28  A14 = subs(f1x1_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
    {0,0,0,0,0,0});
29  A15 = subs(f1t1_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
    {0,0,0,0,0,0});
30  A16 = subs(f1t2_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
    {0,0,0,0,0,0});
31  A21 = subs(f2x1, {x,x_dot,t1,t1_dot,t2,t2_dot},
    {0,0,0,0,0,0});
32  A22 = subs(f2t1, {x,x_dot,t1,t1_dot,t2,t2_dot},
    {0,0,0,0,0,0});
33  A23 = subs(f2t2, {x,x_dot,t1,t1_dot,t2,t2_dot},
    {0,0,0,0,0,0});
34  A24 = subs(f2x1_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
    {0,0,0,0,0,0});
35  A25 = subs(f2t1_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
    {0,0,0,0,0,0});
36  A26 = subs(f2t2_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
    {0,0,0,0,0,0});
37  A31 = subs(f3x1, {x,x_dot,t1,t1_dot,t2,t2_dot},
    {0,0,0,0,0,0});
38  A32 = subs(f3t1, {x,x_dot,t1,t1_dot,t2,t2_dot},
    {0,0,0,0,0,0});
39  A33 = subs(f3t2, {x,x_dot,t1,t1_dot,t2,t2_dot},
    {0,0,0,0,0,0});
40  A34 = subs(f3x1_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
    {0,0,0,0,0,0});
41  A35 = subs(f3t1_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
    {0,0,0,0,0,0});
42  A36 = subs(f3t2_dot, {x,x_dot,t1,t1_dot,t2,t2_dot},
    {0,0,0,0,0,0});
43
44  f2F = diff(A,F);
45  f4F = diff(B,F);
46  f6F = diff(C,F);
47  B11 = subs(f2F, {x,x_dot,t1,t1_dot,t2,t2_dot},
    {0,0,0,0,0,0});
48  B12 = subs(f4F, {x,x_dot,t1,t1_dot,t2,t2_dot},
    {0,0,0,0,0,0});
49  B13 = subs(f6F, {x,x_dot,t1,t1_dot,t2,t2_dot},

```

```

    {0,0,0,0,0,0});
50 B1 = simplify([0;B11;0;B12;0;B13])
51 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])
52 c = simplify(simplify([B1 A1*B1 A1^2*B1 A1^3*B1 A1^4*B1
    A1^5*B1]))
53 B1
54 A1*B1
55 A1^2*B1
56 A1^3*B1
57 A1^4*B1
58 A1^5*B1
59 det(c)
60 %Bf = (subs(B1, {m1,m2,M,l1,l2,g},{a,b,c,d,e,f}))
61 Bf = B1
62 %Af = (subs(A1, {m1,m2,M,l1,l2,g},{a,b,c,d,e,f}))
63 Af = A1
64 end

```