General: The goal of this INDIVIDUAL project is to analyze a particular physical system, the so-called "broom balancing problem", from a linearization point of view. That is, you will design a controller for a nonlinear system using linear feedback design methods. This will give you a chance to use some of what you learned in EECS 560, plus maybe a few other techniques as well. More importantly, it will illustrate a very common design technique for nonlinear systems: linearize, then hope for the best! We will learn more analytical methods during the term and these will be illustrated in HW on this model. To complete the project, you may have to read a few things in the textbook before we cover them in class. Please be assured that students in previous terms have accomplished this successfully and you will too.

Honor Code: You are to do your own work. Discussing the project with a friend is fine. Sharing MATLAB code is not allowed.

Getting the model, Step 1: Consider the inverted pendulum positioning system described in the attached scanned-sheets. Assuming that friction is negligible, but not assuming that the mass of the rod is small in comparison to the mass of the cart, equations (1-15)-(1-18) of the attached sheets can be shown to yield:

$$\begin{aligned}
\frac{d^2\phi}{dt^2} &= \frac{(M+m)mgL\sin(\phi) - (mL\dot{\phi})^2\sin(\phi)\cos(\phi) - (mL\cos(\phi))\mu}{(M+m)(J+mL^2) - (mL)^2\cos^2(\phi)} \\
\frac{d^2 s}{dt^2} &= \frac{(J+mL^2)mL\dot{\phi}^2\sin(\phi) - (mL)^2 g\sin(\phi)\cos(\phi) + (J+mL^2)\mu}{(M+m)(J+mL^2) - (mL)^2\cos^2(\phi)}
\end{aligned} \tag{1}$$

where, here, the dots represent differentiation with respect to the physical time t. Indeed, you may wish to do the tedious derivation for your own pleasure. But this is NOT required. The method of Lagrange can be used to get the model in a much easier and quicker way. I have done this symbolically and verified the above equations. So, you can assume that they are correct and free of typos. [see the file: symb_model_inv_pend_cart.m in the Project folder.]

Getting the model, Step 2: The concept of dimensionless variables is very important in engineering practice because it often leads to models that are numerically better conditioned, and, in addition, it often leads to fewer unknown parameters (as is the case in our example below). It is becoming a lost art in engineering schools. I learned about the subject in Thermodynamics; how about you? Introduce the dimensionless variables:

$$\begin{array}{lll}
\bar{s} = (\frac{M}{m}+1)(\frac{s}{L}) & \bar{\mu} = \frac{1}{(M+m)g}\mu & d = 1 + c \\
\bar{t} = \frac{t}{T}, & \text{where} & T^2 = (\frac{J}{mL^2} + \frac{M}{m+M})\frac{L}{g} \\
b = (\frac{M}{m}+1)(\frac{J}{mL^2}+1) & c = \frac{m^2L^2}{J(m+M)+mML^2} &
\end{array} \tag{2}$$

Using the dimensionless variables defined above, the model dynamics from (1) become:

$$\begin{aligned} \frac{d^2\phi}{d\bar{t}^2} &= \frac{-c\dot{\phi}^2\sin(\phi)\cos(\phi)}{1+c\sin^2(\phi)} + \frac{\sin(\phi)}{1+c\sin^2(\phi)} - \frac{\cos(\phi)}{1+c\sin^2(\phi)}\bar{\mu} \\ \frac{d^2\bar{s}}{d\bar{t}^2} &= \frac{d\cdot\dot{\phi}^2\sin(\phi)}{1+c\sin^2(\phi)} - \frac{\cos(\phi)\sin(\phi)}{1+c\sin^2(\phi)} + \frac{b}{1+c\sin^2(\phi)}\bar{\mu} \end{aligned} \tag{3}$$

where the dots now represent differentiation with respect to the normalized time $\bar{t}$.

Remark: You may wish to verify that the denominator in the model never vanishes, so that the equations are well defined. Do not include this in your solutions.

Problem 1: Verify the Dimensionless Model (10 points): Verify that the equation for ϕ in (3) indeed follows from (1) after substitution of the given dimensionless quantities and proper application of the chain rule. (NOTE: Doing the same for s is similar and is not required here.)

Hint: First note that $\bar{t} = \frac{t}{T}$ implies that $t = T\bar{t}$. Hence, by the chain rule

$$\frac{d\phi(\bar{t})}{d\bar{t}} = \left.\frac{d\phi(t)}{dt}\right|_{t=T\bar{t}} \frac{dt}{d\bar{t}},$$

where of course, $\frac{dt}{d\bar{t}} = T$. You have to figure out the second derivative on your own. No help on this part in office hours!

Problem 2: Linear State-Variable Model (10 points): Construct a state-variable representation of (3) and linearize the equations about the upright equilibrium point. Define the state vector $x \triangleq [x_1 \ \ x_2 \ \ x_3 \ \ x_4]^T$ in the following manner: $x_1 = \phi$, $x_2 = \dot{\phi}$, $x_3 = \bar{s}$, $x_4 = \dot{\bar{s}}$. For the remainder of this project, the following data is assumed:

$$\begin{aligned} &M = 25\ (kg) \qquad m = 20\ (kg) \quad L = 9.81\ (m) \\ &g = 9.81\ (m/s^2) \quad J = \tfrac{1}{3}mL^2 \end{aligned} \tag{4}$$

1. Write down the dynamics for state-vector x in the form $\dot{x} = f(x, \bar{\mu})$.
2. Linearizing the system in part 1 about the origin should yield the lineraized system $\dot{x} = Ax + B\bar{\mu}$. Find matrices A and B.

Problem 3: Stabilizing Feedback (10 points): Verify that the linearized system is completely controllable and then determine a state variable feedback control law $\bar{\mu} = Kx$

that will place the closed-loop eigenvalues at $(-3, -2, -0.7 \pm j0.2)$. You may use the `place` command in MATLAB.

Problem 4: Stability with Full State Feedback (10 points): Apply the control law $\bar{\mu} = Kx$ to both the linearized system and nonlinear system. This gives you two closed-loop systems:

$$\dot{x} = (A + BK)x \tag{5}$$

and

$$\dot{x} = f(x, Kx) \tag{6}$$

Verify that $A + BK = \frac{\partial}{\partial x} f(x, Kx)|_{x=0}$, that is (5) is the linearization of (6) about the origin. What can you conclude about the stability properties of the origin for the nonlinear closed-loop system? See Theorem 4.7 on page 139 of Khalil. See also Definition 4.1 on page 112 of our textbook.

Problem 5: Simulation with Full State Feedback (10 points): Simulate both closed-loop systems for several initial conditions, comparing the results of the linearized model to the nonlinear model. With $x_2(0) = 0$, $x_3(0) = 0$, and $x_4(0) = 0$, explore how large you can make $x_1(0)$ before the nonlinear model goes unstable.

Problem 6: Luenberger Observer Design (15 points): Suppose now that you can only measure the outputs $y_1 = \phi$ and $y_2 = \bar{s}$. Write the output equation $y = Cx$. Verify that the pair (A, C) is completely observable and then design an observer to implement the control law synthesized in Problem 3. This means that you have to design an observer-based design control law using the following steps:

1. Design an asymptotically stabvle observer. You may use pole-placement to design the observer. Your linear observer should have the form:

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu + L(y - \hat{y}) \\ \hat{y} &= C\hat{x} \\ u &= K\hat{x} \end{aligned} \tag{7}$$

   You recognize the observer as being a copy of the linearized model plus an "output injection" term $L(y = \hat{y})$, which adjusts the state estimates as a function of errors in the estimate of the output.

2. Use the control law $\bar{\mu} = K\hat{x}$, where K is the control gain designed in Problem 3.

Note that this is a more difficult pole placement problem than the one you did in Problem 3 because there are two outputs instead of just one; hence there are many ways of achieving the same pole positions. If you have had EECS 565, you may use methods from that course to design the observer gain; otherwise, use once again the `place` command.

Problem 7: Simulation Results for Observer Based Compensator (15 Points): Simulate the observer-based control law (also known as observer-based compensator or dynamic controller) as obtained in Problem 6 on both the linearized and nonlinear system models for a variety of initial conditions. Plot the outputs $y_1(t)$ and $y_2(t)$. How do the responses between the static and observer-based compensator(also known as dynamic controller) compare? Always take the initial conditions of the observer to be $[x_1(0), 0, 0, 0]$. Explore how large you can make $x_1(0)$ (the initial angle of the pendulum) before the nonlinear model goes unstable. Generally speaking, it should be smaller than in Problem 5, though in some cases, one may find the opposite. Do not worry about this.

Problem 8: Observer Based Compensator using Nonlinear Model (20 points) For the nonlinear plant, consider implementing the observer as

$$\begin{aligned} \dot{\hat{x}} &= f(\hat{x}, u) + L(y - \hat{y}) \\ \hat{y} &= C\hat{x} \\ u &= K\hat{x} \end{aligned} \tag{8}$$

which you recognize as being a copy of the nonlinear model plus linear output injection. See if this observer provides better estimates than (**??**) and hence better performance, for the nonlinear system. Why should the origin of the closed-loop nonlinear system still be asymptotically stable? **Hint:** linearize the closed-loop system (8) about the origin and compare to (7).

Contents

Question:

(A) Using linear state variable feedback and with the nonlinear model initialized at $x_2(0) = 0$, $x_3(0) = 0$, $x_4(0) = 0$, the largest I could make $x_1(0)$ before the closed-loop system went unstable was 0.65898 (radians). The linear state variable feedback that I used was $K = [19.300, 22.975, 1.590, 5.525]$ for (A+BK).

(B) Using linear state variable feedback plus an observer, with the observer initialized at the origin and the nonlinear model initialized at $x_2(0) = 0$, $x_3(0) = 0$, $x_4(0) = 0$, the largest I could make $x_1(0)$ before the closed-loop system went unstable was 0.585 (radians). The observer that I used to obtain the best response was nonlinear (see the simulation in problem 8) and the observer gain was $L = \begin{bmatrix} 22 & 0 \\ 121 & 0 \\ 0 & 22 \\ -1 & 120 \end{bmatrix}$, which places poles at $[-10, -10, -12, -12]$.

1 Problem 1: Verify the Dimensionless Model

Problem 1 refers to the verification of the dimensionless model by substituting the provided quantities and apply derivative chain rules. Given that:

$$\frac{d^2\phi(t)}{dt^2} = \frac{(M+m)\cdot m\cdot g\cdot L\cdot \sin(\phi) - \left(m\cdot L\cdot \dot{\phi}\right)^2\cdot \sin(\phi)\cdot\cos(\phi) - (m\cdot L\cdot\cos(\phi))\cdot u}{(M+m)\cdot(J+m\cdot L^2) - (m\cdot L)^2\cdot\cos^2(\phi)} \tag{1}$$

The parameters used in equation (1) are defined as follows:

$$\begin{aligned}
\bar{u} &= \frac{1}{(M+m)\cdot g}\cdot u\\
\bar{t} &= \frac{t}{T}\\
c &= \frac{m^2\cdot L^2}{J\cdot(m+M)+m\cdot M\cdot L^2}\\
T^2 &= \left(\frac{J}{mL^2}+\frac{M}{m+M}\right)\cdot\frac{L}{g} = \frac{J\cdot(m+M)+M\cdot mL^2}{mL\cdot(M+m)\cdot g}\\
u &= (M+m)\cdot g\cdot\bar{u}
\end{aligned} \tag{2}$$

The first derivative with respect to $\bar{t}$:

$$\frac{d\phi(\bar{t})}{d\bar{t}} = \left.\frac{d\phi(t)}{dt}\right|_{t=T\bar{t}}\cdot\frac{dt}{d\bar{t}} = \left.\frac{d\phi(t)}{dt}\right|_{t=T\bar{t}}\cdot\frac{d}{d\bar{t}}(T\cdot d\bar{t}) = T\cdot\left.\frac{d\phi(t)}{dt}\right|_{t=T\bar{t}} \tag{3}$$

The second derivative can be derived by chain rule:

$$\begin{aligned}
\frac{d^2\phi(\bar{t})}{d\bar{t}^2} &= \frac{d}{dt}\left[\left.\frac{d\phi(t)}{dt}\right|_{t=T\bar{t}}\cdot\frac{dt}{d\bar{t}}\right]\Bigg|_{t=T\bar{t}}\cdot\frac{dt}{d\bar{t}} = T^2\cdot\left.\frac{d^2\phi(t)}{dt^2}\right|_{t=T\bar{t}}\\
&= T^2\cdot\frac{(M+m)\cdot m\cdot g\cdot L\cdot\sin(\phi) - \left(m\cdot L\cdot\dot{\phi}\right)^2\cdot\sin(\phi)\cdot\cos(\phi) - (m\cdot L\cdot\cos(\phi))\cdot(M+m)\cdot g\cdot\bar{u}}{(M+m)\cdot(J+m\cdot L^2)-(m\cdot L)^2\cdot\cos^2(\phi)}\\
&= \frac{\frac{J\cdot(m+M)+M\cdot mL^2}{mL\cdot(M+m)\cdot g}\cdot(M+m)\cdot m\cdot g\cdot L\cdot\sin(\phi)}{(M+m)\cdot(J+m\cdot L^2)-(m\cdot L)^2\cdot\cos^2(\phi)} - \frac{\frac{J\cdot(m+M)+M\cdot mL^2}{mL\cdot(M+m)\cdot g}\cdot\left(m\cdot L\cdot\dot{\phi}\right)^2\cdot\sin(\phi)\cdot\cos(\phi)}{(M+m)\cdot(J+m\cdot L^2)-(m\cdot L)^2\cdot\cos^2(\phi)} -\\
&\quad \frac{\frac{J\cdot(m+M)+M\cdot mL^2}{mL\cdot(M+m)\cdot g}\cdot(m\cdot L\cdot\cos(\phi))\cdot(M+m)\cdot g\cdot\bar{u}}{(M+m)\cdot(J+m\cdot L^2)-(m\cdot L)^2\cdot\cos^2(\phi)}\\
&= \frac{\frac{1}{mL\cdot(M+m)\cdot g}\cdot(M+m)\cdot m\cdot g\cdot L\cdot\sin(\phi)}{\frac{(M+m)\cdot(J+m\cdot L^2)-(m\cdot L)^2\cdot\cos^2(\phi)}{J\cdot(m+M)+M\cdot mL^2}} - \frac{\frac{1}{mL\cdot(M+m)\cdot g}\cdot\left(m\cdot L\cdot\dot{\phi}\right)^2\cdot\sin(\phi)\cdot\cos(\phi)}{\frac{(M+m)\cdot(J+m\cdot L^2)-(m\cdot L)^2\cdot\cos^2(\phi)}{J\cdot(m+M)+M\cdot mL^2}} -\\
&\quad \frac{\frac{1}{mL\cdot(M+m)\cdot g}\cdot(m\cdot L\cdot\cos(\phi))\cdot(M+m)\cdot g\cdot\bar{u}}{\frac{(M+m)\cdot(J+m\cdot L^2)-(m\cdot L)^2\cdot\cos^2(\phi)}{J\cdot(m+M)+M\cdot mL^2}}\\
&= \frac{\sin(\phi)}{1+c\sin^2(\phi)} - \frac{c\cdot\dot{\phi}^2\cdot\sin(\phi)\cdot\cos(\phi)}{1+c\sin^2(\phi)} - \frac{\cos(\phi)\cdot\bar{u}}{1+c\sin^2(\phi)}
\end{aligned} \tag{4}$$

2 Problem 2: Linear State-Variable Model

2.1 Problem 2.1: State-Vector dynamics

Construct a state-variable representation and linearize the equations about the upright equilibrium point. As defined in problem statement, the state vector x is chosen as $x = [x_1\ x_2\ x_3\ x_4]^T = \left[\phi\ \dot{\phi}\ \bar{s}\ \dot{\bar{s}}\right]^T$. The state equations contain unknown parameters J, c, b, and d, which can be numerically determined by implementing the following codes.

```
%% Problem 1 (1) Parameters Setting
M = 25;
m =20;
L = 9.81;
g = 9.81;

J = (1/3)*m*L^2;
c = (m^2*L^2)/(J*(m+M)+m*M*L^2);
b = (M/m + 1)*(J/(m*L^2) + 1);
d = 1 + c;
```

The parameters can be calculated as $J = 6.41574$, $c = 0.5$, $b = 3$, and $d = 1.5$. Therefore, the corresponding state-vector form of nonlinear equations $\dot{x} = f(x, \bar{u})$ can be expressed as equations (5).

$$
\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \frac{-c \cdot x_2^2 \cdot \sin(x_1)\cos(x_1)}{1 + c\sin^2(x_1)} + \frac{\sin(x_1)}{1 + c\sin^2(x_1)} - \frac{\cos(x_1)}{1 + c\sin^2(x_1)} \cdot \bar{u} \\
\dot{x}_3 &= x_4 \\
\dot{x}_4 &= \frac{d \cdot x_2^2 \cdot \sin(x_1)}{1 + c\sin^2(x_1)} - \frac{\cos(x_1)\sin(x_1)}{1 + c\sin^2(x_1)} + \frac{b}{1 + c\sin^2(x_1)} \cdot \bar{u}
\end{aligned}
\tag{5}
$$

2.2 Problem 2.2: Linearized System $\dot{x} = Ax + B\bar{u}$

Jacobian linearization is essential to determined the linearized system. Equation (6) linearizes the state equations and simplifies most of the unimportant terms (i.e. most partial derivatives result in 0 and 1 terms and they have been processed and simplified here for better illustration).

$$
\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{\partial f_4}{\partial x_1} & \frac{\partial f_4}{\partial x_2} & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{\partial f_2}{\partial \bar{u}} \\ 0 \\ \frac{\partial f_4}{\partial \bar{u}} \end{bmatrix} \cdot \bar{u}
\tag{6}
$$

$$\left.\frac{\partial f(x,Kx)}{\partial x}\right|_{x=0} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{\partial f_2}{\partial x_1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{\partial f_4}{\partial x_1} & \frac{\partial f_4}{\partial x_2} & \frac{\partial f_4}{\partial x_3} & \frac{\partial f_4}{\partial x_4} \end{bmatrix} \tag{7}$$

The remaining 6 partial derivatives are related to the corresponding state equations, and these derivatives can be determined at origin (x=0) as follows:

$$\left.\frac{\partial f_2}{\partial x_1}\right|_{x=0} = \frac{-c\cdot\cos^2(x_1)\cdot x_2^2 + \cos(x_1) + c\cdot\sin^2(x_1)\cdot x_2^2 + \bar{u}\cdot\sin(x_1)}{c\cdot\sin^2(x_1)+1} + \left.\frac{2\cdot c\cdot\cos(x_1)\cdot\sin(x_1)\cdot\left[\cos(x_1)\cdot\bar{u} - \sin(x_1) + c\cdot\cos(x_1)\cdot\sin(x_1)\cdot x_2^2\right]}{\left(c\cdot\sin^2(x_1)+1\right)^2}\right|_{x_1=0,x_2=0} = 1$$

$$\left.\frac{\partial f_2}{\partial x_2}\right|_{x=0} = \left.\frac{-2\cdot c\cdot\cos(x_1)\cdot\sin(x_1)\cdot x_2}{c\cdot\sin^2(x_1)+1}\right|_{x_1=0,x_2=0} = 0$$

$$\left.\frac{\partial f_4}{\partial x_1}\right|_{x=0} = \frac{\sin^2(x_1) - \cos^2(x_1) + d\cdot\cos(x_1)\cdot x_2^2}{c\cdot\sin^2(x_1)+1} - \left.\frac{2\cdot c\cdot\cos(x_1)\cdot\sin(x_1)\cdot\left[d\cdot\sin(x_1)\cdot x_2^2 + b\cdot\bar{u} - \cos(x_1)\cdot\sin(x_1)\right]}{\left(c\cdot\sin^2(x_1)+1\right)^2}\right|_{x_1=0,x_2=0} = -1$$

$$\left.\frac{\partial f_4}{\partial x_2}\right|_{x=0} = \left.\frac{2\cdot d\cdot\sin(x_1)\cdot x_2}{c\cdot\sin^2(x_1)+1}\right|_{x_1=0,x_2=0} = 0$$

$$\left.\frac{\partial f_2}{\partial \bar{u}}\right|_{x=0} = -\left.\frac{\cos(x_1)}{1+c\sin^2(x_1)}\right|_{x_1=0,x_2=0} = 1$$

$$\left.\frac{\partial f_4}{\partial \bar{u}}\right|_{x=0} = \left.\frac{b}{1+c\sin^2(x_1)}\cdot\bar{u}\right|_{x_1=0,x_2=0} = 3$$

Then the linearized model matrixes A and B can be determined as:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \end{bmatrix} \textbf{and} B = \begin{bmatrix} 0 \\ -1 \\ 0 \\ 3 \end{bmatrix} \tag{8}$$

3 Problem 3: Stabilizing Feedback

This problem refers to verifying the controllability of the linearized system and determining the state feedback control law via pole placement. To determine the controllability of the linearized system, the corresponding controllability matrix can be determined as follows:

$$CO = \left[B \ AB \ A^2B \ A^3B\right] = \begin{bmatrix} 0 & -1 & 0 & -1 \\ -1 & 0 & -1 & 0 \\ 0 & 0 & 3 & 1 \\ 3 & 3 & 1 & 0 \end{bmatrix} \tag{9}$$

Then, due to the fact that $rank\,(CO) = 4$, it can be concluded that the linearized system is completely controllable. By apply the control feedback to the linearied system, the following equation can be derived:

$$\dot{x} = Ax + B\bar{u} = Ax + B\,(Kx) = (A + BK)\,x \tag{10}$$

To stabilize the system, feedback control gain K will be designed to place eigenvalues at (-3,-2,-0.7+j0.2,-0.7-j0.2). The following MATLAB codes can be used to determine feedback control gain:

```
%% Problem 3: Stabilizing Feedback
% Define system matrix
A = [0 1 0 0;1 0 0 0;0 0 0 1;-1 0 0 0];
B = [0; -1; 0; 3];
% Define desired poles
P = [-3,-2,-0.7+0.2i,-0.7-0.2i];
% Determine feedback gain
K = -place(A,B,P)

% Check controllability
ContMatrix = [B A*B A^2*B A^3*B]
rank(ContMatrix)
% Check the pole placement
eig(A+B*K) % Pay attention of the sign of control law
```

The corresponding feedback control gain is determined as $K = [19.300, 22.975, 1.590, 5.525]$.

4 Problem 4: Stability with Full-State Feedback

4.1 Feedback Control to Linearized System $\dot{x} = Ax + B\bar{u}$

Given that $\bar{u} = Kx$, the corresponding system can be derived as $\dot{x} = Ax + B(Kx) = (A + BK)x$.

$$A+BK = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \\ 0 \\ 3 \end{bmatrix} \begin{bmatrix} 19.3 & 22.975 & 1.59 & 5.525 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -18.3 & -22.975 & -1.59 & -5.525 \\ 0 & 0 & 0 & 1 \\ 56.9 & 68.925 & 4.77 & 16.575 \end{bmatrix}$$

4.2 Closed-loop Linearization of $\dot{x} = f(x, Kx)$

As determined in the previous question, $\bar{u}$ introduces feedback of all states, including x_1, x_2, x_3, and x_4, as shown below:

$$\bar{u} = Kx = 19.3 \cdot x_1 + 22.975 \cdot x_2 + 1.59 \cdot x_3 + 5.525 \cdot x_4 \tag{11}$$

By plugging the control input into system equations (12), the following equation can be obtained:

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \frac{-c \cdot x_2^2 \cdot \sin(x_1)\cos(x_1)}{1 + c\sin^2(x_1)} + \frac{\sin(x_1)}{1 + c\sin^2(x_1)} - \frac{\cos(x_1)}{1 + c\sin^2(x_1)} \cdot (19.3x_1 + 22.975x_2 + 1.59x_3 + 5.525x_4) \\
\dot{x}_3 &= x_4 \\
\dot{x}_4 &= \frac{d \cdot x_2^2 \cdot \sin(x_1)}{1 + c\sin^2(x_1)} - \frac{\cos(x_1)\sin(x_1)}{1 + c\sin^2(x_1)} + \frac{b}{1 + c\sin^2(x_1)} \cdot (19.3x_1 + 22.975x_2 + 1.59x_3 + 5.525x_4)
\end{aligned} \tag{12}$$

Then, apply Jacobian linearization to the feedback system. The corresponding calculation illustrates the essential terms that require MATLAB calculations.

$$\left.\frac{\partial f(x, Kx)}{\partial x}\right|_{x=0} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \frac{\partial f_2}{\partial x_3} & \frac{\partial f_2}{\partial x_4} \\ 0 & 0 & 0 & 1 \\ \frac{\partial f_4}{\partial x_1} & \frac{\partial f_4}{\partial x_2} & \frac{\partial f_4}{\partial x_3} & \frac{\partial f_4}{\partial x_4} \end{bmatrix}$$

These calculations of partial derivatives are intensive; hence, MATLAB is used to determine the partial derivates and correspondingly plug in the initial conditions.

```
 %% Partial Derivative with Feedback Control Law
% Define the symbolic terms
syms x1(t) x2(t) u(t) x3(t) x4(t) x5(t) x6(t)
```

```
syms c b d
% Define the expression of control law
u = 19.3*x1 + 22.975*x2 + 1.59*x3 + 5.525*x4;
% Define the state-vector equations with feedback
x1_dot = x2;
x2_dot = (-c*x2^2*sin(x1)*cos(x1) + sin(x1) - cos(x1)* u )/...
    (1+c*(sin(x1))^2);
x3_dot = x4;
x4_dot = (d*x2^2*sin(x1) - cos(x1)*sin(x1) + b*u)/...
    (1+c*(sin(x1))^2);

% Taking partial derivative symbolically
dx2_dot_dx1 = diff(x2_dot,x1);
dx2_dot_dx2 = diff(x2_dot,x2);
dx2_dot_dx3 = diff(x2_dot,x3);
dx2_dot_dx4 = diff(x2_dot,x4);
dx4_dot_dx1 = diff(x4_dot,x1);
dx4_dot_dx2 = diff(x4_dot,x2);
dx4_dot_dx3 = diff(x4_dot,x3);
dx4_dot_dx4 = diff(x4_dot,x4);

% Pluging the corresponding symbolic values
f21 = subs(dx2_dot_dx1(t), [x1(t),x2(t),x3(t),x4(t),c,b,d], [0, 0, 0, 0, 0.5,3,1.5])
f22 = subs(dx2_dot_dx2(t), [x1(t),x2(t),x3(t),x4(t),c,b,d], [0, 0, 0, 0, 0.5,3,1.5])
f23 = subs(dx2_dot_dx3(t), [x1(t),x2(t),x3(t),x4(t),c,b,d], [0, 0, 0, 0, 0.5,3,1.5])
f24 = subs(dx2_dot_dx4(t), [x1(t),x2(t),x3(t),x4(t),c,b,d], [0, 0, 0, 0, 0.5,3,1.5])

f41 = subs(dx4_dot_dx1(t), [x1(t),x2(t),x3(t),x4(t),c,b,d], [0, 0, 0, 0, 0.5,3,1.5])
f42 = subs(dx4_dot_dx2(t), [x1(t),x2(t),x3(t),x4(t),c,b,d], [0, 0, 0, 0, 0.5,3,1.5])
f43 = subs(dx4_dot_dx3(t), [x1(t),x2(t),x3(t),x4(t),c,b,d], [0, 0, 0, 0, 0.5,3,1.5])
f44 = subs(dx2_dot_dx4(t), [x1(t),x2(t),x3(t),x4(t),c,b,d], [0, 0, 0, 0, 0.5,3,1.5])

% Print out each term
fline2 = [f21 f22 f23 f24]
fline4 = [f41 f42 f43 f44]
```

The results can be directly outputted as:

```
fline2 =

[-183/10, -919/40, -159/100, -221/40]


fline4 =

[569/10, 2757/40, 477/100, -221/40]

fx >>
```

Therefore, the linearization of nonlinear system about upright position can be determined:

$$\left.\frac{\partial f\left(x,Kx\right)}{\partial x}\right|_{x=0} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -18.3 & -22.975 & -1.59 & -5.525 \\ 0 & 0 & 0 & 1 \\ 56.9 & 68.925 & 4.77 & 16.575 \end{bmatrix} \tag{13}$$

Theorem 4.7 states the sufficient condition of the linearized nonlinear system for origin as the equilibrium point (i.e. $x_{eq} = 0$). If all eigenvalues of the linearized system matrix A are in left hand plane (LHP), the origin is asymptotically stable. **The eigenvalues of equation (13) are [-3, -2, -0.7+0.2i, -0.7-0.2i], which implies that the system is asymptotically stable.**

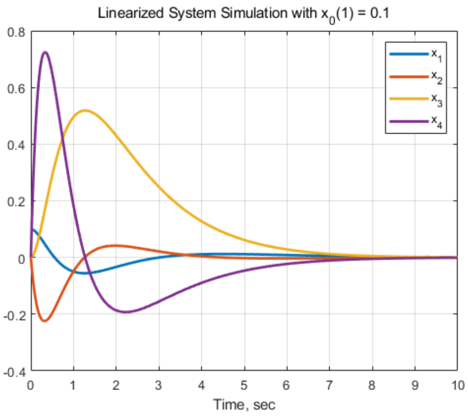
5 Problem 5: Simulation with Full-State Feedback

Simulate both closed-loop systems for several initial conditions, comparing the results of the linearized model to the nonlinear model.

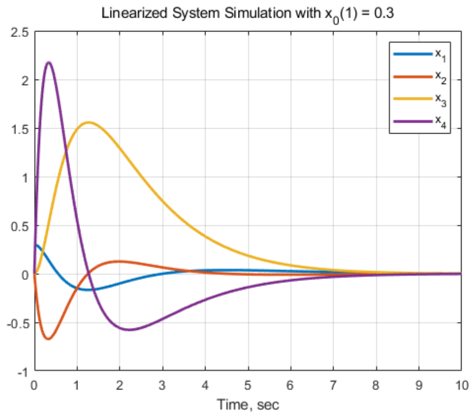
5.1 Simulation of Closed-loop Linearized System

```
%% Problem 5
%% Simulation for Closed-loop Linearized Model
A = [0 1 0 0;1 0 0 0;0 0 0 1;-1 0 0 0];
B = [0; -1; 0; 3];
Acl = [A + B*K]; % Closed-loop feedback system matrix
Bcl = [B];
Ccl = [1 0 0 0; 0 1 0 0; 0 0 1 0; 0 0 0 1];% Output each state
SScl = ss(Acl,Bcl,Ccl,0);

x0 = [0.65898; 0; 0; 0];        % Initial Condition
T = 0:0.01:10;                  % Simulation time = 10 seconds
U = zeros(size(T));
[Y, Tsim, X] = lsim(SScl,U,T,x0);  % Simulate
plot(Tsim,Y(:,1),'Linewidth',2);                 % plot the output vs. time
hold on
plot(Tsim,Y(:,2),'Linewidth',2);                   % plot the output vs. time
hold on
plot(Tsim,Y(:,3),'Linewidth',2);
hold on
plot(Tsim,Y(:,4),'Linewidth',2);
xlabel('Time, sec');
grid on
```

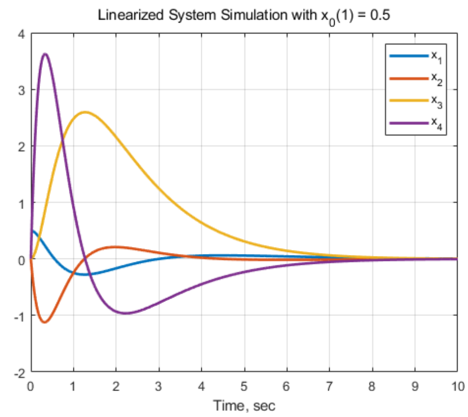
```
title('Linearized System Simulation with x_0(1) = 0.65898')
legend('x_1','x_2','x_3','x_4')
```



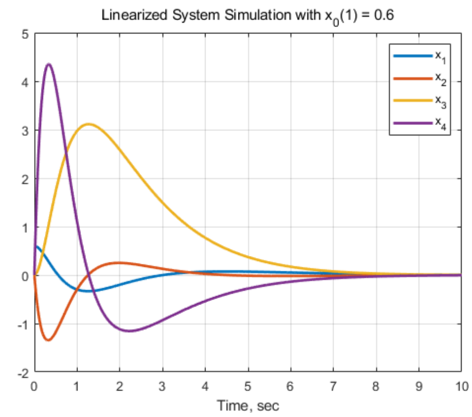
(a) Simulation of Full-state Feedback Linearized System for $x_0 = 0.1$



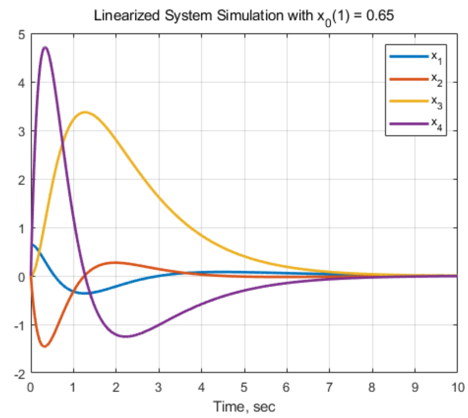
(b) Simulation of Full-state Feedback Linearized System for $x_0 = 0.3$



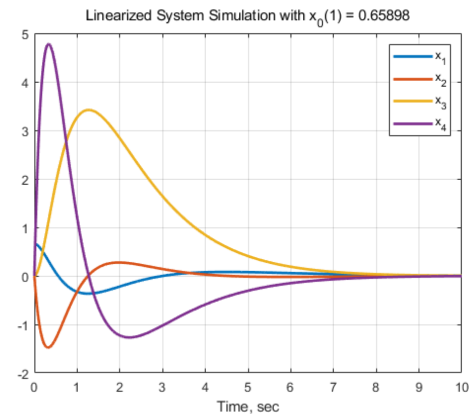
(c) Simulation of Full-state Feedback Linearized System for $x_0 = 0.5$



(d) Simulation of Full-state Feedback Linearized System for $x_0 = 0.6$



(e) Simulation of Full-state Feedback Linearized System for $x_0 = 0.65$



(f) Simulation of Full-state Feedback Linearized System for $x_0 = 0.65898$

Figure 1: Simulation Results of Full-state Feedback Linearized System

5.2 Simulation of Closed-loop Nonlinear System

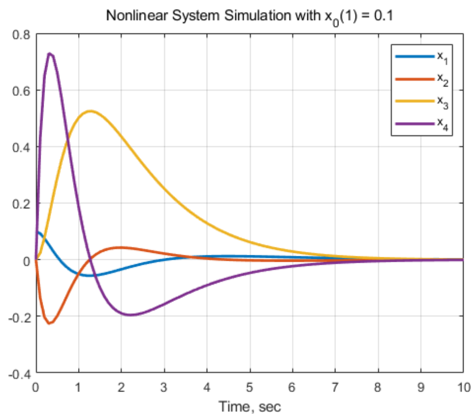
To simulate nonlinear system, it is of great importance to define the state-space equations before simulating in *ode*45(). The following MATLAB script demonstrates the equation setting for simulation.

$$
\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \frac{-c \cdot x_2^2 \cdot \sin(x_1)\cos(x_1)}{1 + c\sin^2(x_1)} + \frac{\sin(x_1)}{1 + c\sin^2(x_1)} - \frac{\cos(x_1)}{1 + c\sin^2(x_1)} \cdot (19.3x_1 + 22.975x_2 + 1.59x_3 + 5.525x_4) \\
\dot{x}_3 &= x_4 \\
\dot{x}_4 &= \frac{d \cdot x_2^2 \cdot \sin(x_1)}{1 + c\sin^2(x_1)} - \frac{\cos(x_1)\sin(x_1)}{1 + c\sin^2(x_1)} + \frac{b}{1 + c\sin^2(x_1)} \cdot (19.3x_1 + 22.975x_2 + 1.59x_3 + 5.525x_4)
\end{aligned}
\tag{14}
$$

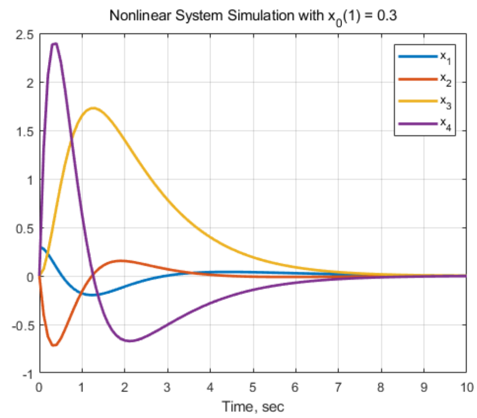
```matlab
function dxdt = P5ODE(x,c,b,d)
% This script is used to simulate nonlinear system in problem 5
  dxdt = zeros(4,1);
  dxdt(1) = x(2);
  dxdt(2) = (-c*x(2)^2*sin(x(1))*cos(x(1)) + sin(x(1)) - cos(x(1))* (19.3*x(1) + ...
      22.975*x(2) + 1.59*x(3) + 5.525*x(4)) )/...
    (1+c*(sin(x(1)))^2);
  dxdt(3) = x(4);
  dxdt(4) = (d*x(2)^2*sin(x(1)) - cos(x(1))*sin(x(1)) + b*(19.3*x(1) + ...
      22.975*x(2) + 1.59*x(3) + 5.525*x(4)))/...
    (1+c*(sin(x(1)))^2);
end
```

Then, the following MATLAB codes can be implemented to simulate nonlinear system response:

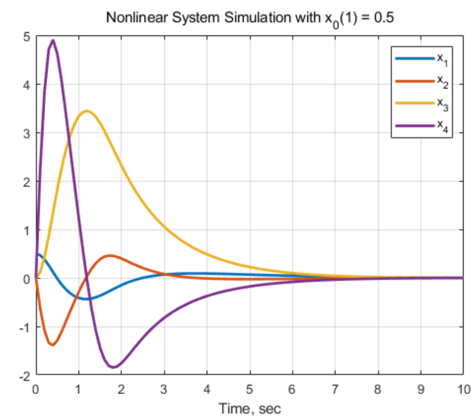
```matlab
%% Simulation for Closed-loop Linearized Nonlinear System
tspan = [0:0.1:10];
x0 = [0.65898 0 0 0];
[t,x] = ode45(@(t,x) P5ODE(x,c,b,d), tspan, x0);
plot(t,x(:,1),'Linewidth',2);
hold on
plot(t,x(:,2),'Linewidth',2);
hold on
plot(t,x(:,3),'Linewidth',2);
hold on
plot(t,x(:,4),'Linewidth',2);
xlabel('Time, sec');
grid on
title('Nonlinear System Simulation with x_0(1) = 0.65898')
legend('x_1','x_2','x_3','x_4')
```



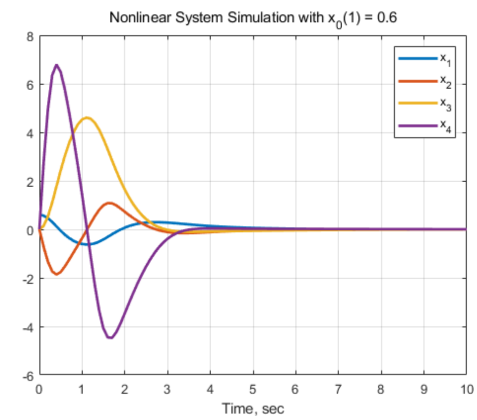
(a) Simulation of Full-state Feedback Nonlinear System for $x_0 = 0.1$



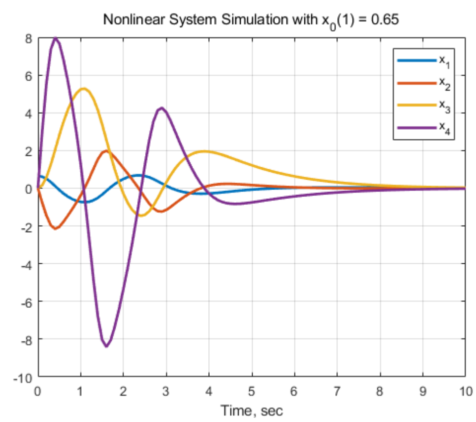
(b) Simulation of Full-state Feedback Nonlinear System for $x_0 = 0.3$



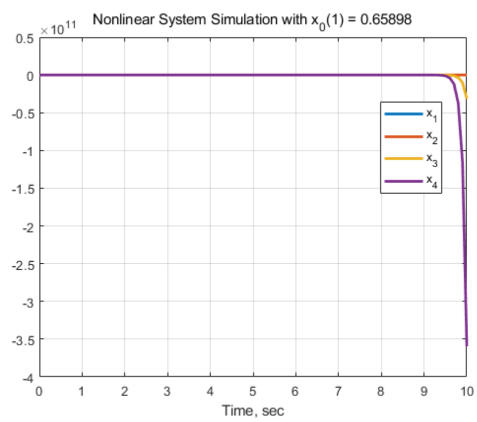
(c) Simulation of Full-state Feedback Nonlinear System for $x_0 = 0.5$



(d) Simulation of Full-state Feedback Nonlinear System for $x_0 = 0.6$



(e) Simulation of Full-state Feedback Nonlinear System for $x_0 = 0.65$



(f) Simulation of Full-state Feedback Nonlinear System for $x_0 = 0.65898$

Figure 2: Simulation Results of Full-state Feedback Nonlinear System

Based on the simulation, **the nonlinear system becomes unstable when $x_0 = 0.65898$.**

6 Problem 6: Luenberger Observer Design

6.1 Asymptotically Stable Observer Design via Pole Placement

To verify the observability of the system, the following observability matrix is calculated, and **the $rank\,(OB) = 4$ can be directly observed, which implies that the system is completely observable.**

$$OB = \begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix}$$

Assume linear observer has the form of the following equation (15).

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu + L\,(y - \hat{y}) \\ \hat{y} &= C\hat{x} \\ u &= K\hat{x} \end{aligned} \tag{15}$$

Assume the corresponding real linearized system is expressed as (16).

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{16}$$

Define error as $e_x = x - \hat{x}$, and then the error dynamics is $\dot{e}_x = \dot{x} - \dot{\hat{x}}$. The error dynamics $\dot{e}_x$ can be expressed as follows:

$$\begin{aligned} \dot{e}_x &= \dot{x} - \dot{\hat{x}} \\ &= (Ax + Bu) - A\hat{x} + Bu + L\,(y - \hat{y}) \\ &= (A - LC)\,x - (A - LC)\,\hat{x} \\ &= (A - LC)\,e_x \end{aligned} \tag{17}$$

Equation (15) implies that, by stabilizing $(A - LC)$ to be asymptotically stable, the error dynamics $\dot{e}_x$ will eventually approach zero, which means that the estimated state $\hat{x}$ will approach x and hence forms an excellent observer. **To choose observer gain L, the rule of thumb is that the eigenvalues of $A - LC$ should be sufficiently faster than that of $A - BK$** so that the estimated state $\hat{x}$ can be observed before performing feedback. In this project, the eigenvalues are placed at $[-10, -10, -12, -12]$, which is faster than controller poles. The corresponding MATLAB implementation has been demonstrated:

```
%% Problem 6 Luenberger Observer Design
%% Subquestion (1) Observer pole placement
A = [0 1 0 0;1 0 0 0;0 0 0 1;-1 0 0 0];
C = [1 0 0 0; 0 0 1 0];
ObsMatrix = [C; C*A; C*A^2; C*A^3]
rank(ObsMatrix)
P = [-10, -10, -12, -12];
% Transpose each matrix due to duality theorem
L = place(A',C',P)'
eig(A-L*C)
```

By applying pole placement, the corresponding L can be determined as:

$$L = \begin{bmatrix} 22 & 0 \\ 121 & 0 \\ 0 & 22 \\ -1 & 120 \end{bmatrix} \tag{18}$$

The feedback control law $\bar{u} = K\hat{x}$ utilize the estimated states from observer and hence can be expressed as:

$$\begin{aligned} \bar{u} = K\hat{x} &= 19.3\hat{x}_1 + 22.975\hat{x}_2 + 1.59\hat{x}_3 + 5.525\hat{x}_4 \\ &= 19.3\,(x_1 - e_{x1}) + 22.975\,(x_2 - e_{x2}) + 1.59\,(x_3 - e_{x3}) + 5.525\,(x_4 - e_{x4}) \end{aligned} \tag{19}$$

7 Problem 7: Simulation for Observer-Based Compensator

7.1 Simulation of Linearized Observer-based Compensator

After defining the error dynamics, the augmented system can be designed as equation (20) by applying control law $u = K\hat{x}$.

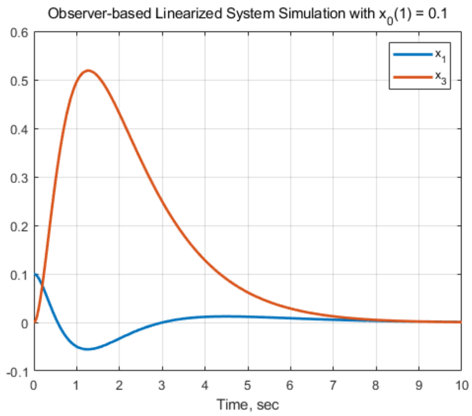
$$\begin{aligned} \dot{x} &= Ax + Bu = Ax + BK\hat{x} \\ &= Ax + BK\left(x - e_x\right) \\ &= \left(A + BK\right)x - BKe_x \\ y &= Cx \end{aligned} \tag{20}$$

Therefore, the corresponding stata-space model can be obtained:

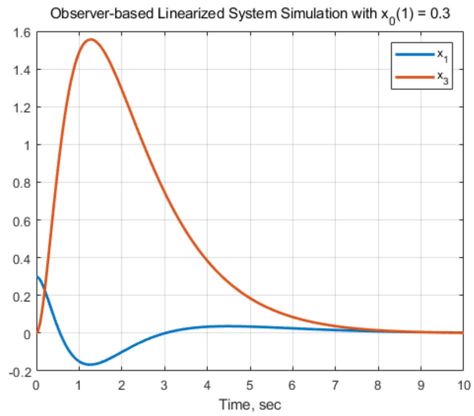
$$\begin{aligned} \begin{bmatrix} \dot{x} \\ \dot{e}_x \end{bmatrix} &= \begin{bmatrix} A + BK & -BK \\ O_{4\times4} & A - LC \end{bmatrix} \begin{bmatrix} x \\ e_x \end{bmatrix} \\ y &= [C \; O_{2\times4}] \cdot \begin{bmatrix} x \\ e_x \end{bmatrix} \end{aligned} \tag{21}$$

State Space model (21) illustrates the complete observer-based feedback design, which output $x_1 = \phi$ and $x_3 = \bar{s}$ for observation.

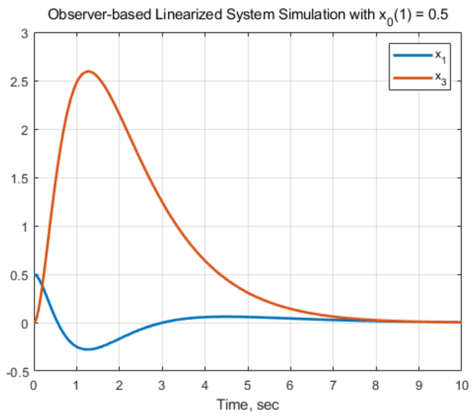
```
%% Problem 7 Observer Based Compensator using Nonlinear Model
%% Observer-based Linearized System Simulation
O_44 = zeros(4,4);
C = [1 0 0 0;0 0 1 0];
Aobs = [A + B*K -B*K;
        O_44 A-L*C];
Bobs = [zeros(4,1);zeros(4,1)];
O_24 = zeros(2,4);
Cobs = [C O_24];

SysObs = ss(Aobs,Bobs,Cobs,0);

x0 = [10 0 0 0 0 0 0 0];
T = 0:0.01:10;                 % simulation time = 10 seconds
U = zeros(size(T));
[Y, Tsim, X] = lsim(SysObs,U,T,x0);  % simulate
plot(Tsim,Y(:,1))                    % plot the output vs. time
hold on
plot(Tsim,Y(:,2))                    % plot the output vs. time
```



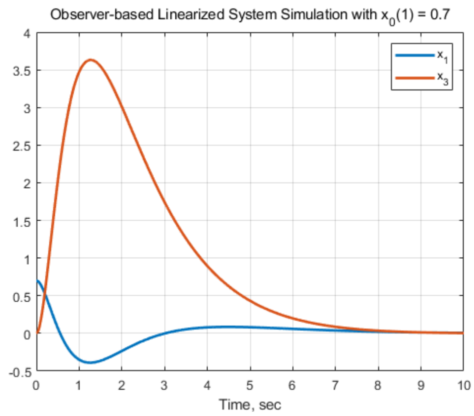
(a) Simulation of Observed-based Linearized System for $x_0 = 0.1$



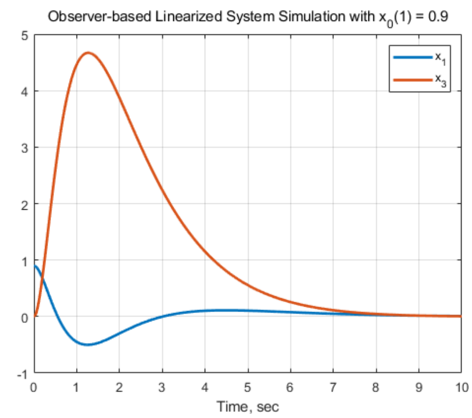
(b) Simulation of Observed-based Linearized System for $x_0 = 0.3$



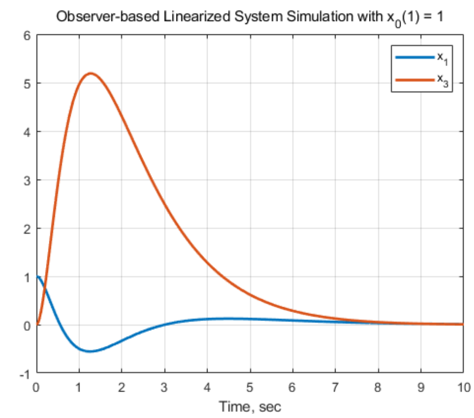
(c) Simulation of Observed-based Linearized System for $x_0 = 0.5$



(d) Simulation of Observed-based Linearized System for $x_0 = 0.7$



(e) Simulation of Observed-based Linearized System for $x_0 = 0.9$



(f) Simulation of Observed-based Linearized System for $x_0 = 1$

Figure 3: Simulation Results of Observed-based Linearized System

7.2 Simulation of Nonlinear Observer-based Compensator

Applying the observer-based control law (23) to the linearized system, the following state space model can be derived in terms of state x and error e_x defined as $[x_1\ x_2\ x_3\ x_4\ x_5\ x_6\ x_7\ x_8]^T = \left[\phi\ \dot{\phi}\ \bar{s}\ \dot{\bar{s}}\ e_{x1}\ \dot{e}_{x1}\ e_{x2}\ \dot{e}_{x2}\right]^T = \left[\phi\ \dot{\phi}\ \bar{s}\ \dot{\bar{s}}\ e_{x1}\ e_{x2}\ e_{x3}\ e_{x4}\right]^T$.

Based on the definition of error dynamics and the corresponding state variables, the equation set (22) can be used to further derive the model of (26).

$$\begin{cases} \hat{x}_1 = x_1 - e_{x1} = x_1 - x_5 \\ \hat{x}_2 = x_2 - e_{x2} = x_2 - x_6 \\ \hat{x}_3 = x_3 - e_{x3} = x_3 - x_7 \\ \hat{x}_4 = x_4 - e_{x4} = x_4 - x_8 \end{cases} \tag{22}$$

Therefore, the feedback control law $\bar{u} = K\hat{x}$ utilizes the estimated states from the observer and hence can be expressed as:

$$\begin{aligned} \bar{u} = K\hat{x} &= 19.3\hat{x}_1 + 22.975\hat{x}_2 + 1.59\hat{x}_3 + 5.525\hat{x}_4 \\ &= 19.3\left(x_1 - e_{x1}\right) + 22.975\left(x_2 - \dot{e}_{x2}\right) + 1.59\left(x_3 - e_{x3}\right) + 5.525\left(x_4 - e_{x4}\right) \\ &= 19.3\left(x_1 - x_5\right) + 22.975\left(x_2 - x_6\right) + 1.59\left(x_3 - x_7\right) + 5.525\left(x_4 - x_8\right) \end{aligned} \tag{23}$$

To construct the augmented state space model in (26), the observer state space is contructed as follows:

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + B\left(K\hat{x}\right) + L\left(y - \hat{y}\right) \\ &= \left(A + BK\right)\hat{x} + LC\left(x - \hat{x}\right) \\ &= \left(A + BK\right)\hat{x} + LC \cdot e_x \end{aligned} \tag{24}$$

Substitute the feedback gain K and observer gain L into equation (24). The following system can be derived, which is essential to numerically construct the state-space equation (26).

$$\begin{bmatrix} \dot{\hat{x}}_1 \\ \dot{\hat{x}}_2 \\ \dot{\hat{x}}_3 \\ \dot{\hat{x}}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -18.3 & -22.975 & -1.59 & -5.525 \\ 0 & 0 & 0 & 1 \\ 56.9 & 68.925 & 4.77 & 16.575 \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \\ \hat{x}_4 \end{bmatrix} + \begin{bmatrix} 22 & 0 & 0 & 0 \\ 121 & 0 & 0 & 0 \\ 0 & 0 & 22 & 0 \\ -1 & 0 & 120 & 0 \end{bmatrix} \begin{bmatrix} e_{x1} \\ e_{x2} \\ e_{x3} \\ e_{x4} \end{bmatrix} \tag{25}$$

Therefore, based on the results of equation (22), (23), and (25). The state-space model is constructed as follows:

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = \frac{-c \cdot x_2^2 \cdot \sin(x_1)\cos(x_1)}{1 + c\sin^2(x_1)} + \frac{\sin(x_1)}{1 + c\sin^2(x_1)} - \frac{\cos(x_1)}{1 + c\sin^2(x_1)} \cdot [19.3\,(x_1 - x_5) + 22.975\,(x_2 - x_6) + 1.59\,(x_3 - x_7) + 5.525\,(x_4 - x_8)$$

$$\dot{x}_3 = x_4$$

$$\dot{x}_4 = \frac{d \cdot x_2^2 \cdot \sin(x_1)}{1 + c\sin^2(x_1)} - \frac{\cos(x_1)\sin(x_1)}{1 + c\sin^2(x_1)} + \frac{b}{1 + c\sin^2(x_1)} \cdot [19.3\,(x_1 - x_5) + 22.975\,(x_2 - x_6) + 1.59\,(x_3 - x_7) + 5.525\,(x_4 - x_8)$$

$$\dot{x}_5 = \dot{x}_1 - \dot{\hat{x}}_1 = x_2 - (\hat{x}_2 + 22x_5) = x_2 - (-x_6 + x_2) - 22x_5 = x_6 - 22x_5$$

$$\dot{x}_6 = \dot{x}_2 - \dot{\hat{x}}_2 = \frac{-c \cdot x_2^2 \cdot \sin(x_1)\cos(x_1)}{1 + c\sin^2(x_1)} + \frac{\sin(x_1)}{1 + c\sin^2(x_1)} - (-18.3\hat{x}_1 - 22.975\hat{x}_2 - 1.59\hat{x}_3 - 5.525\hat{x}_4 + 121x_5)$$

$$\dot{x}_7 = \dot{x}_3 - \dot{\hat{x}}_3 = x_4 - (\hat{x}_4 + 22x_7) = x_8 - 22x_7$$

$$x_8 = \ddot{x}_3 - \dot{\hat{x}}_4 = \frac{d \cdot x_2^2 \cdot \sin(x_1)}{1 + c\sin^2(x_1)} - \frac{\cos(x_1)\sin(x_1)}{1 + c\sin^2(x_1)} - (56.9\hat{x}_1 + 68.925\hat{x}_2 + 4.77\hat{x}_3 + 16.575\hat{x}_4 - x_5 + 120x_7) \tag{26}$$

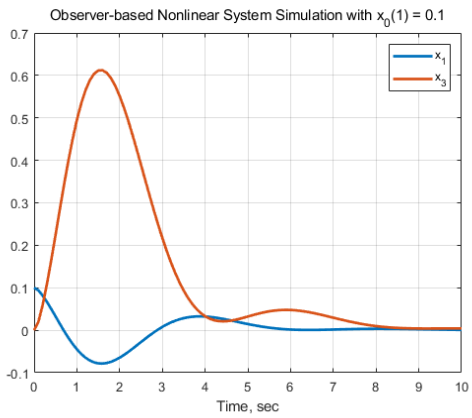
The implementation of this observed-based state-space model is demonstrated below:

```
function dxdt = MYODE7(x,c,b,d)
% Nonlinear Observer-based Compensator State Space Model Implementation
L = [22 ,0;121,0;0,22;-1,120];
C = [1 0 0 0; 0 0 1 0];
  dxdt = zeros(8,1);
K = [19.3,22.975,1.59,5.525];
  u = (K*[x(1)-x(5);x(2)-x(6);x(3)-x(7);x(4)-x(8)]);

  % Estimated state in terms of real state variables
  x1hat = -x(5) + x(1);
  x2hat = -x(6) + x(2);
  x3hat = -x(7) + x(3);
  x4hat = -x(8) + x(4);

  % System state space equations
  dxdt(1) = x(2);
  dxdt(2) = (-c*x(2)^2*sin(x(1))*cos(x(1))+sin(x(1))-cos(x(1))...
 *u)/(1+c*(sin(x(1)))^2);
  dxdt(3) = x(4);
  dxdt(4) = (d*x(2)^2*sin(x(1))-cos(x(1))*sin(x(1))+b*u)/(1+c*(sin(x(1)))^2);
  dxdt(5) = x(6) - 22*x(5);
  dxdt(6) = (-c*x(2)^2*sin(x(1))*cos(x(1))+sin(x(1))-cos(x(1))*(0))/(1 + ...
      c*(sin(x(1)))^2) -...
```

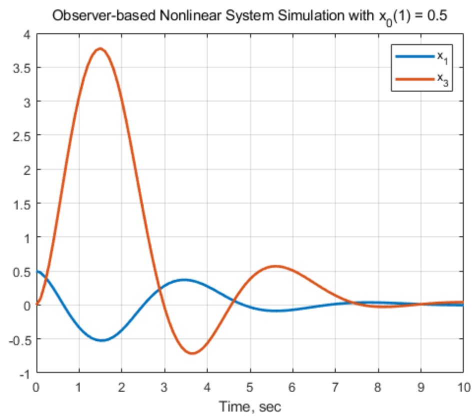
```
      (-18.3*x1hat - 22.975*x2hat - 1.59*x3hat - 5.525*x4hat + 121*x(5));
  dxdt(7) = x(8) - 22*x(7);
  dxdt(8) = (d*x(2)^2*sin(x(1))-cos(x(1))*sin(x(1))+b*(0))/(1+c*(sin(x(1)))^2)  -...
      (56.9*x1hat + 68.925*x2hat + 4.77*x3hat + 16.575*x4hat -1*x(5) + 120*x(7));
end
```

The following MATLAB codes can be used to simulate the response:

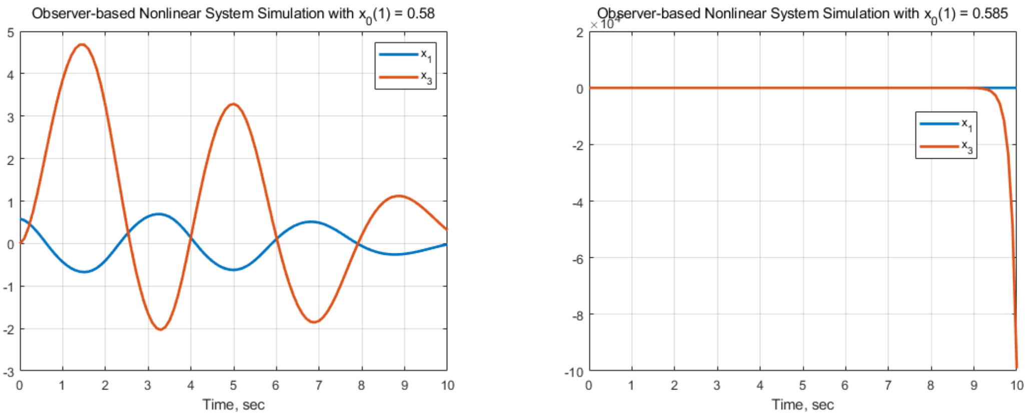
```
%% Observer-based Nonlinear System Simulation
tspan = [0:0.1:10];
% x0 = [0.958 0 0 0 0 0 0 0];
x0 = [0.958 0 0 0 0 0 0 0];
[t,x] = ode45(@(t,x) MYODE7(x,c,b,d), tspan, x0);

plot(t,x(:,1),'Linewidth',2);
hold on
plot(t,x(:,2),'Linewidth',2);
hold on
plot(t,x(:,3),'Linewidth',2);
hold on
plot(t,x(:,4),'Linewidth',2);
xlabel('Time, sec');
grid on
title('Observer-based Nonlinear System Simulation with x_0(1) = 0.958')
legend('x_1','x_2','x_3','x_4')
```



(a) Simulation of Observed-based Nonlinear System for $x_0 = 0.1$



(b) Simulation of Observed-based Nonlinear System for $x_0 = 0.5$



(c) Simulation of Observed-based Nonlinear System for $x_0 = 0.58$

(d) Simulation of Observed-based Nonlinear System for $x_0 = 0.585$

Figure 4: Simulation Results of Observed-based Nonlinear System

Based on the simulation, **the nonlinear system becomes unstable when $x_0 = 0.585$.**

8 Problem 8: Simulation for Observer-Based Compensator

For nonlinear system and nonlinear observer, the system can be integrated as following general form in equation sets (28).

$$
\begin{aligned}
\dot{x} &= f(x, \bar{u}) \\
y &= g(x, \bar{u}) \\
\dot{\hat{x}} &= \hat{f}(\hat{x}, \bar{u}) + L(y - \hat{y}) = \hat{f}(\hat{x}, \bar{u}) + L(g(x, \bar{u}) - g(\hat{x}, \bar{u})) \\
\hat{y} &= g(\hat{x}, \bar{u}) \\
\dot{e}_x &= \dot{x} - \dot{\hat{x}} = f(x, \bar{u}) - \hat{f}(\hat{x}, \bar{u}) - L(g(x, \bar{u}) - g(\hat{x}, \bar{u})) \\
\bar{u} &= K\hat{x}
\end{aligned}
\tag{27}
$$

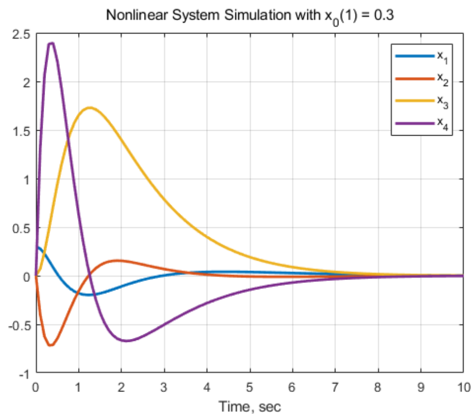
Then, the following state space model can be derived based on equations (28).;

$$
\begin{aligned}
&\dot{x}_1 = x_2 \\
&\dot{x}_2 = \frac{-c \cdot x_2^2 \cdot \sin(x_1)\cos(x_1)}{1 + c\sin^2(x_1)} + \frac{\sin(x_1)}{1 + c\sin^2(x_1)} - \frac{\cos(x_1)}{1 + c\sin^2(x_1)} \cdot [19.3(x_1 - x_5) + 22.975(x_2 - x_6) + \\
&\quad 1.59(x_3 - x_7) + 5.525(x_4 - x_8) \\
&\dot{x}_3 = x_4 \\
&\dot{x}_4 = \frac{d \cdot x_2^2 \cdot \sin(x_1)}{1 + c\sin^2(x_1)} - \frac{\cos(x_1)\sin(x_1)}{1 + c\sin^2(x_1)} + \frac{b}{1 + c\sin^2(x_1)} \cdot [19.3(x_1 - x_5) + 22.975(x_2 - x_6) + 1.59(x_3 - x_7) + \\
&\quad 5.525(x_4 - x_8) \\
&\dot{x}_5 = \dot{x}_1 - \dot{\hat{x}}_1 = x_2 - (\hat{x}_2 + 22x_5) = x_2 - (-x_6 + x_2) - 22x_5 = x_6 - 22x_5 \\
&\dot{x}_6 = \dot{x}_2 - \dot{\hat{x}}_2 = \frac{-c \cdot x_2^2 \cdot \sin(x_1)\cos(x_1)}{1 + c\sin^2(x_1)} + \frac{\sin(x_1)}{1 + c\sin^2(x_1)} - \frac{-c \cdot \hat{x}_2^2 \cdot \sin(\hat{x}_1)\cos(\hat{x}_1)}{1 + c\sin^2(\hat{x}_1)} - \frac{\sin(\hat{x}_1)}{1 + c\sin^2(\hat{x}_1)} \\
&\quad = \frac{-c \cdot x_2^2 \cdot \sin(x_1)\cos(x_1)}{1 + c\sin^2(x_1)} + \frac{\sin(x_1)}{1 + c\sin^2(x_1)} - \frac{-c \cdot (x_2 - x_6)^2 \cdot \sin(x_1 - x_5)\cos(x_1 - x_5)}{1 + c\sin^2(x_1 - x_5)} - \frac{\sin(x_1 - x_5)}{1 + c\sin^2(x_1 - x_5)} \\
&\dot{x}_7 = \dot{x}_3 - \dot{\hat{x}}_3 = x_4 - (\hat{x}_4 + 22x_7) = x_8 - 22x_7 \\
&x_8 = \ddot{x}_3 - \dot{\hat{x}}_4 = \frac{d \cdot x_2^2 \cdot \sin(x_1)}{1 + c\sin^2(x_1)} - \frac{\cos(x_1)\sin(x_1)}{1 + c\sin^2(x_1)} - \frac{d \cdot \hat{x}_2^2 \cdot \sin(\hat{x}_1)}{1 + c\sin^2(\hat{x}_1)} + \frac{\cos(\hat{x}_1)\sin(\hat{x}_1)}{1 + c\sin^2(\hat{x}_1)} \\
&\quad = \frac{d \cdot x_2^2 \cdot \sin(x_1)}{1 + c\sin^2(x_1)} - \frac{\cos(x_1)\sin(x_1)}{1 + c\sin^2(x_1)} - \frac{d \cdot (x_2 - x_6)^2 \cdot \sin(x_1 - x_5)}{1 + c\sin^2(x_1 - x_5)} + \frac{\cos(x_1 - x_5)\sin(x_1 - x_5)}{1 + c\sin^2(x_1 - x_5)}
\end{aligned}
\tag{28}
$$

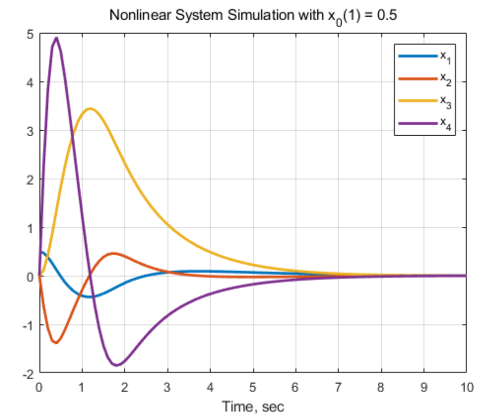
The corresponding MATLAB implementation is demonstrated as follows:

```
function dxdt = P8ODE(x,c,b,d)
% Problem 8 Nonlinear plant plus Observer
  L = [22 ,0;121,0;0,22;-1,120];
  C = [1 0 0 0; 0 0 1 0];
  dxdt = zeros(8,1);
```

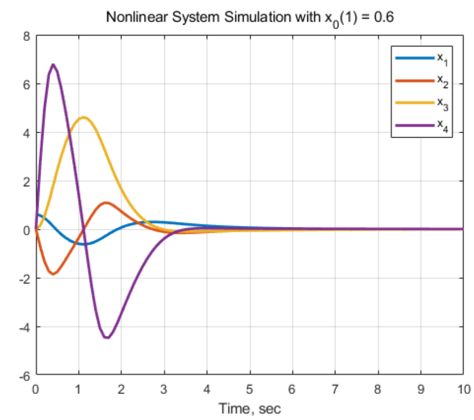
```
    K = [19.3,22.975,1.59,5.525];
    u = (K*[x(1)-x(5);x(2)-x(6);x(3)-x(7);x(4)-x(8)]);
    % System state space equations
    dxdt(1) = x(2);
    dxdt(2) = (-c*x(2)^2*sin(x(1))*cos(x(1))+sin(x(1))-cos(x(1))...
   *u)/(1+c*(sin(x(1)))^2);
    dxdt(3) = x(4);
    dxdt(4) = (d*x(2)^2*sin(x(1))-cos(x(1))*sin(x(1))+b*u)/(1+c*(sin(x(1)))^2);
    dxdt(5) = x(6) - 22*x(5);
    dxdt(6) = (-c*x(2)^2*sin(x(1))*cos(x(1))+sin(x(1))-cos(x(1))*(0))/(1 +...
        c*(sin(x(1)))^2) -(-c*(x(2)-x(6))^2*sin(x(1)-x(5))*cos(x(1)-x(5))+...
        sin(x(1)-x(5))-cos(x(1)-x(5))*(0))/(1+ c*(sin(x(1)-x(5)))^2);
    dxdt(7) = x(8) - 22*x(7);
    dxdt(8) = (d*x(2)^2*sin(x(1))-cos(x(1))*sin(x(1))+b*(0))/(1+c*(sin(x(1)))^2)...
        -(d*(x(2)-x(6))^2*sin(x(1)-x(5))-cos(x(1)-x(5))*sin(x(1)-x(5))+b*(0))/(1...
    +c*(sin(x(1)-x(5)))^2);
  end
```



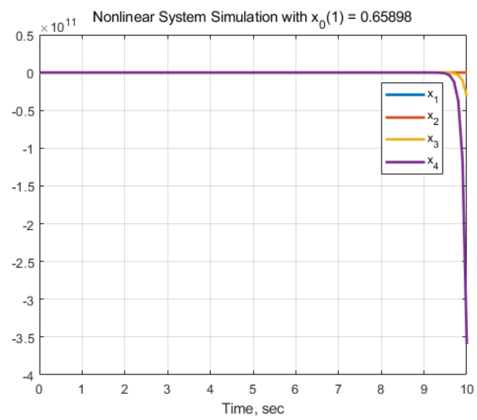
(a) Simulation of Nonlinear Model Observer-based Compensator for $x_0 = 0.3$



(b) Simulation of Observed-based Nonlinear System for $x_0 = 0.5$



(c) Simulation of Nonlinear Model Observer-based Compensator for $x_0 = 0.6$



(d) Simulation of Observed-based Nonlinear System for $x_0 = 0.65898$

Figure 5: Simulation Results of Nonlinear Model Observer-based Compensator

Based on the simulation, **the nonlinear model becomes unstable when $x_0 = 0.65898$.**

In addition, it is reasonable to conclude that **nonlinear model-based observer leads to better state estimation and hence better performance**. The stability of the model can be verified by linearizing the nonlinear model around the origin, which leads to its linearized model that has been verified to be asymptotically stable.