In the name of God

دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

HOMEWORK 2
(STATE-SPACE REALIZATION)

By

Reza Nopour
402126924

Advanced Automatic Control

Mechanical Engineering Department

Amirkabir University of Technology

5 Nov. 23

Problem 1 Description

Determine the following:

-Canonical controllable and observable (Companion) and Jordan forms for part a.

-Jordan form for part b.

a. $H(s) = \dfrac{s+2}{s((s+1)^2+4)}$

b. $H(s) = \dfrac{s+3}{(s+1)^2(s+2)}$

Solution

a) The equation can be rewritten as,

$$H(s) = \frac{s+2}{s((s+1)^2+4)} = \frac{s+2}{s(s^2+2s+1+4)} = \frac{s+2}{s(s^2+2s+5)} = \frac{s+2}{s^3+2s^2+5s}$$

Controllable Canonical Form can be introduced as shown here, where $n = 3, a_1 = 2, a_2 = 5, a_3 = b_0 = b_1 = 0, b_2 = 1, b_3 = 2.$

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix} = \begin{bmatrix}0 & 1 & 0\\ 0 & 0 & 1\\ 0 & -5 & -2\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} + \begin{bmatrix}0\\ 0\\ 1\end{bmatrix}[u]$$

$$y = \begin{bmatrix}2 & 1 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} + [0][u]$$

Observable Canonical Form

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix} = \begin{bmatrix}0 & 0 & 0\\ 1 & 0 & -5\\ 0 & 1 & -2\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} + \begin{bmatrix}2\\ 1\\ 0\end{bmatrix}[u]$$

$$y = \begin{bmatrix}0 & 0 & 1\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} + [0][u]$$

Observable Companion Form

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix}=\begin{bmatrix}0&0&0\\1&0&-5\\0&1&-2\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}+\begin{bmatrix}1\\0\\0\end{bmatrix}[u]$$

$$y=\begin{bmatrix}0&1&0\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}+[0][u]$$

Jordan (Diagonal) Form

$$H(s)=\frac{s+2}{s^3+2s^2+5s}=\frac{0.4}{s}+\frac{-2-0.15i}{\big(s+(1-i2)\big)}+\frac{-2+0.15i}{\big(s+(1+i2)\big)}$$

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix}=\begin{bmatrix}0&0&0\\0&-1+i2&0\\0&0&-1-i2\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}+\begin{bmatrix}1\\1\\1\end{bmatrix}[u]$$

$$y=\begin{bmatrix}0.4&-2-0.15i&-2+0.15i\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}+[0][u]$$

b) The equation can be rewritten as,

$$H(s)=\frac{s+3}{(s+1)^2(s+2)}=\frac{c_1}{(s+1)^2}+\frac{c_2}{(s+1)}+\frac{c_3}{(s+2)}$$

Solving for $c_i, i=1,2,3$.

$$c_1s+2c_1+c_2(s+2)(s+1)+c_3(s+1)^2$$
$$=c_1s+2c_1+c_2s^2+3c_2s+2c_2+c_3s^2+2c_3s+c_3$$

$$\left.\begin{aligned}ex_1 &\rightarrow c_1+3c_2+2c_3=1\\ ex_2 &\rightarrow 2c_1+2c_2+c_3=3\\ x_3 &\rightarrow c_2+c_3=0\end{aligned}\right\}\rightarrow c_1=2, c_2=-1, c_3=1$$

therefore,

$$H(s)=\frac{s+3}{(s+1)^2(s+2)}=\frac{2}{(s+1)^2}+\frac{-1}{(s+1)}+\frac{1}{(s+2)}$$

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix}=\begin{bmatrix}-1&1&0\\0&-1&0\\0&0&-2\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}+\begin{bmatrix}0\\1\\1\end{bmatrix}[u]$$

$$y=\begin{bmatrix}2&-1&1\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}+[0][u]$$

Problem 2 Description

Referring to Friedland Prob.2.1, the inverted pendulum on a cart driven by an electric motor. Let the state vector, control, and outputs be defined by:

$$x = [x, \dot{x}, \theta, \dot{\theta}] \quad u = e \quad y = [x, \theta]'$$

a. Drive the given equations of motion. Find the matrices A, B, C, and D for the state-space equations of the system.

b. Draw the block-diagram representation of the system.

c. Find the transfer functions from the input u to the two outputs.

d. Write the observable and controllable companion forms for each TF.

Use the following numerical data:

$$m = 0.1kg\,, \quad M = 2.0\,kg, \quad l = 1.0\,m, \quad g = 9.8\,m.s^{-2}, \quad k = 1\,V.s,$$

$$R = 100\,\Omega, \quad \mathrm{r} = 0.02\ \mathrm{m}$$

Solution

a) Considering the position of M as x instead of y according to the illustration.

$$T = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m\left(\dot{x}^2 + l_2\dot{\theta}^2 + 2\dot{x}l\dot{\theta}\cos\theta\right)$$
$$U = mgl\cos\theta$$

So, the Lagrangian form of this system can be written as,

$$\frac{\partial T}{\partial \dot{x}} = M\dot{x} + m\left(\dot{x} + l\dot{\theta}\cos\theta\right)$$
$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{x}}\right) = M\ddot{x} + m\left(\ddot{x} + l\ddot{\theta}\cos\theta - l\dot{\theta}^2\sin\theta\right)$$
$$\frac{\partial T}{\partial \dot{\theta}} = m\left(\dot{x}l\cos\theta + l^2\dot{\theta}\right)$$
$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\theta}}\right) = m\left(\ddot{x}l\cos\theta + l^2\ddot{\theta} - \dot{x}\dot{\theta}l\sin\theta\right)$$
$$\frac{\partial T}{\partial x} = 0, \frac{\partial T}{\partial \dot{\theta}} = -m\dot{x}\dot{\theta}l\sin\theta$$
$$\frac{\partial U}{\partial x} = 0, \frac{\partial U}{\partial \theta} = -mgl\sin\theta$$
$$Q_x = f, Q_\theta = 0.$$

The system equation of motion can be derived as,

$$\begin{cases} M\ddot{x} + m\left(\ddot{x} + l\ddot{\theta}\cos\theta - l\dot{\theta}^2\sin\theta\right) = f \\ m\left(\ddot{x}l\cos\theta + l^2\ddot{\theta} - \dot{x}\dot{\theta}l\sin\theta\right) + m\dot{x}\dot{\theta}l\sin\theta - mgl\sin\theta = 0 \end{cases}$$

Above equations are nonlinear, considering small rotation, the equation can be rewritten as,

$$\begin{cases} M\ddot{x} + m\left(\ddot{x} + l\ddot{\theta} - l\dot{\theta}^2\theta\right) = f & (*) \\ \ddot{x} + l\ddot{\theta} - g\theta = 0 \rightarrow \ddot{\theta} = \dfrac{(g\theta - \ddot{x})}{l} & (**) \end{cases}$$

Substituting $\ddot{\theta}$ in $(*)$ yields,

$$M\ddot{x} + m\left(\ddot{x} + g\theta - \ddot{x} - l\dot{\theta}^2\theta\right) = f \rightarrow \ddot{x} + \frac{m}{M}\left(g\theta - l\dot{\theta}^2\theta\right) = \frac{f}{M}$$

it can be derived as,

$$\ddot{x} = \frac{f}{M} - \frac{mg}{M}\theta - \frac{ml}{M}\dot{\theta}^2\theta$$

Obtained $\ddot{x}$ can be inserted into $(**)$,

$$\ddot{\theta} - \left(\frac{M+m}{Ml}\right)g\theta - \frac{m}{M}\dot{\theta}^2\theta = -\frac{f}{Ml}$$

Final equation can be obtained as,

$$\begin{cases} \ddot{x} + \dfrac{mg}{M}\theta = \dfrac{f}{M} & (***) \\ \ddot{\theta} - \left(\dfrac{M+m}{Ml}\right)g\theta = -\dfrac{f}{Ml} & (****) \end{cases}$$

According to the question,

$$\tau = rf$$
$$r\theta = x \rightarrow r\omega = \dot{x}$$
$$\rightarrow f = \frac{\tau}{r} = \frac{k}{Rr}e - \frac{k^2}{Rr}\omega = \frac{k}{Rr}e - \frac{k^2}{Rr^2}\dot{x}$$

thus,

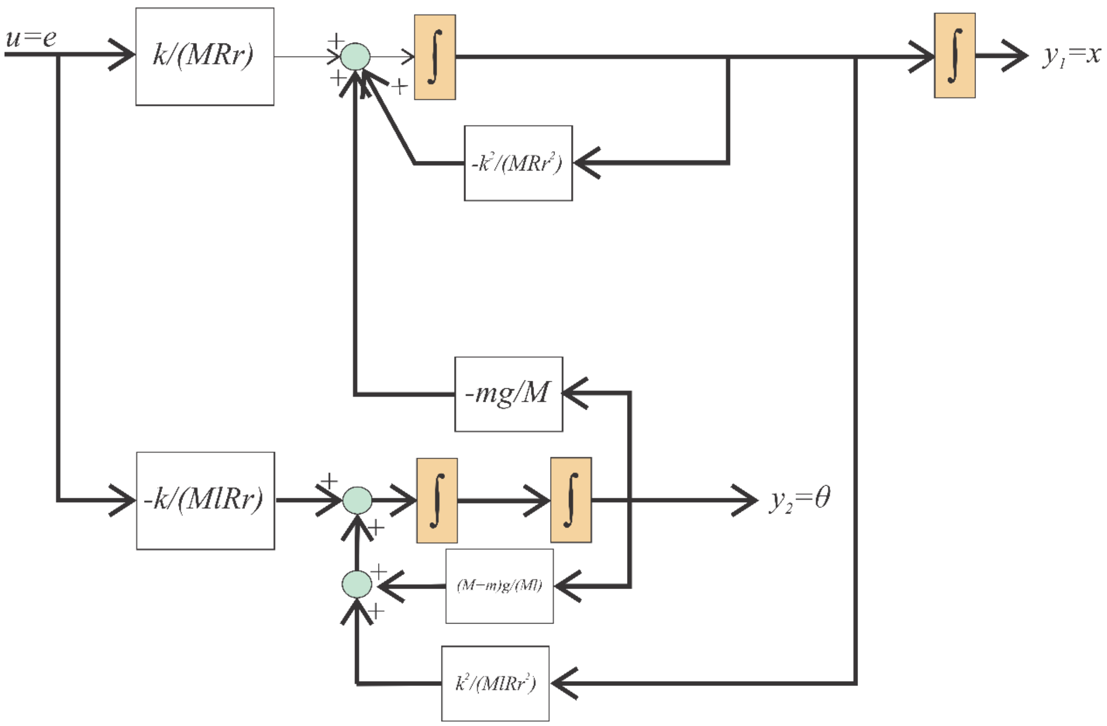
$$\begin{cases} \ddot{x} + \dfrac{k^2}{MRr^2}\dot{x} + \dfrac{mg}{M}\theta = \dfrac{k}{MRr}e & (*) \\ \ddot{\theta} - \dfrac{k^2}{MlRr^2}\dot{x} - \left(\dfrac{M+m}{Ml}\right)g\theta = -\dfrac{k}{MlRr}e & (**) \end{cases}$$

State Space representation of this system is,

$$\dot{\boldsymbol{x}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\dfrac{k^2}{MRr^2} & -\dfrac{mg}{M} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \dfrac{k^2}{MlRr^2} & \dfrac{M+m}{Ml}g & 0 \end{bmatrix} \begin{Bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{Bmatrix} + \begin{bmatrix} 0 \\ \dfrac{k}{MRr} \\ 0 \\ -\dfrac{k}{MlRr} \end{bmatrix} [e]$$

$$\boldsymbol{y} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} [e]$$

b) The block-diagram representation of this system can be shown as,



c) Transfer function,

Using given data, state space matrices can be obtained as,

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -12.5 & -0.49 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 12.5 & 10.29 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0.25 \\ 0 \\ -0.25 \end{bmatrix}$$

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

$$D = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$H = C(sI - A)^{-1}B + D$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \left(\begin{bmatrix} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \\ 0 & 0 & s & 0 \\ 0 & 0 & 0 & s \end{bmatrix} - \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -12.5 & -0.49 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 12.5 & 10.29 & 0 \end{bmatrix} \right)^{-1} \begin{bmatrix} 0 \\ 0.25 \\ 0 \\ -0.25 \end{bmatrix}$$

$$H_1(s) = -\frac{5(5s^2 - 49)}{s(-100s^3 - 1250s^2 + 1029s + 12250)}$$

$$H_1(s) = \frac{25s}{-100s^3 - 1250s^2 + 1029s + 12250}$$

$$H_1(s) = \frac{0.25s^2 - 2.45}{S^4 + 12.5s^3 - 10.29s^2 - 122.5s}$$

$$H_2(s) = \frac{-0.25s^2}{S^4 + 12.5s^3 - 10.29s^2 - 122.5s}$$

d) Observable and controllable form,

For the first transfer function we have, $n = 4, a_1 = 12.5, a_2 = -10.29, a_3 = -122.5, a_4 = 0, b_0 = b_1 = 0, b_2 = 0.25, b_3 = 0, b_4 = -2.45.$

Controllable Canonical Form:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 122.5 & 10.29 & -12.5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} [u]$$

$$y = \begin{bmatrix} -2.45 & 0 & 0.25 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + [0][u]$$

Observable Canonical Form:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 122.5 \\ 0 & 1 & 0 & 10.29 \\ 0 & 0 & 1 & -12.5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} -2.45 \\ 0 \\ 0.25 \\ 0 \end{bmatrix} [u]$$

$$y = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + [0][u]$$

For the second transfer function we have, $n = 4, a_1 = 12.5, a_2 = -10.29, a_3 = -122.5, a_4 = 0, b_0 = b_1 = 0, b_2 = -0.25, b_3 = b_4 = 0.$

Controllable Canonical Form:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 122.5 & 10.29 & -12.5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} [u]$$

$$y = \begin{bmatrix} 0 & 0 & -0.25 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + [0][u]$$

Observable Canonical Form:

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\\ \dot{x}_4\end{bmatrix}=\begin{bmatrix}0 & 0 & 0 & 0\\ 1 & 0 & 0 & 122.5\\ 0 & 1 & 0 & 10.29\\ 0 & 0 & 1 & -12.5\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\\ x_4\end{bmatrix}+\begin{bmatrix}0\\ 0\\ -0.25\\ 0\end{bmatrix}[u]$$

$$y=\begin{bmatrix}0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\\ x_4\end{bmatrix}+[0][u]$$

Problem 2.1 Description

To check the results of part.1-Q1 use `Gp = tf (num , den)` command to build the transfer function and then use `canon(Gp,'companion'),` this command directly produces the observable canonical form. You can find the controllable canonical form with this result. Finally use `[A, B, C, D] = tf2ss(num, den)` and then find the Jordan form with `[J, V] = jordan(A)`.

Solution

The results can be obtained from the following MATLAB code, (**Table 1**):

Table 1 MATLAB code of part.1-Q1.

```
%%%%%%%%%%% Advanced Automatic Control%%%%%%%%%
%%%%%%%% Mechanical Engineering Department%%%%%
%%%%%%% Amirkabir University of Technology%%%%%
%%%%%%%%%%%%%%%%% 5 Nov. 23 %%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%% Reza Nopour 402126924 %%%%%%%%%%%%%
clc
clear
close all
%%
num=[1 2];
den=[1 2 5 0];
%%
Gp = tf (num , den)
canon(Gp,'companion')
[A, B, C, D] = tf2ss(num, den);
[V, J] = jordan(A)
```

Table 2 shows the output

Table 2 Output of MATLAB code of part.1-Q1.

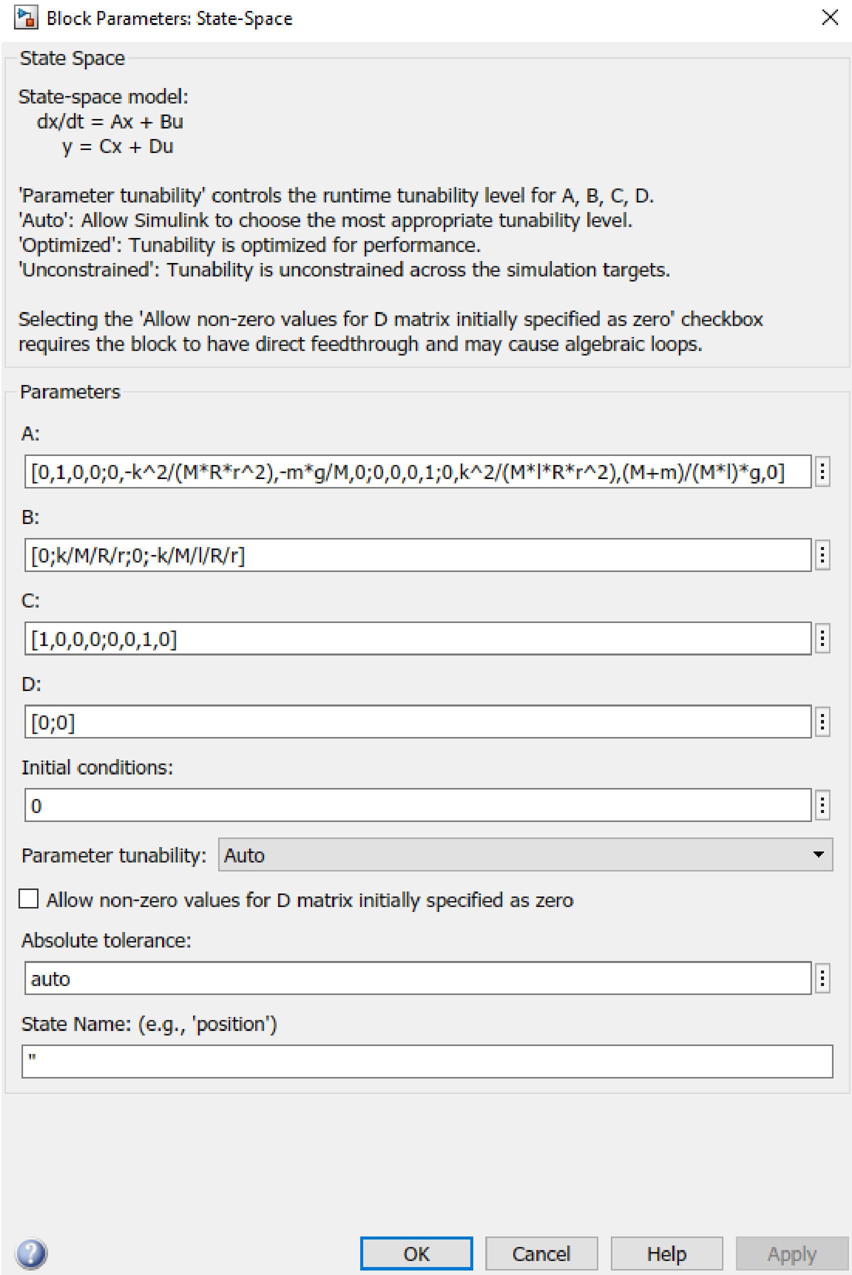
```
Gp =

        s + 2
  -----------------
  s^3 + 2 s^2 + 5 s

Continuous-time transfer function.


ans =

  A =
       x1  x2  x3
   x1   0   0   0
   x2   1   0  -5
   x3   0   1  -2

  B =
       u1
   x1   1
   x2   0
   x3   0

  C =
       x1  x2  x3
   y1   0   1   0

  D =
       u1
   y1   0

Continuous-time state-space model.


V =

   0.0000 + 0.0000i  -3.0000 + 4.0000i  -3.0000 - 4.0000i
   0.0000 + 0.0000i  -1.0000 - 2.0000i  -1.0000 + 2.0000i
   1.0000 + 0.0000i   1.0000 + 0.0000i   1.0000 + 0.0000i


J =

   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i
   0.0000 + 0.0000i  -1.0000 - 2.0000i   0.0000 + 0.0000i
   0.0000 + 0.0000i   0.0000 + 0.0000i  -1.0000 + 2.0000i
```

Problem 2.2 Description

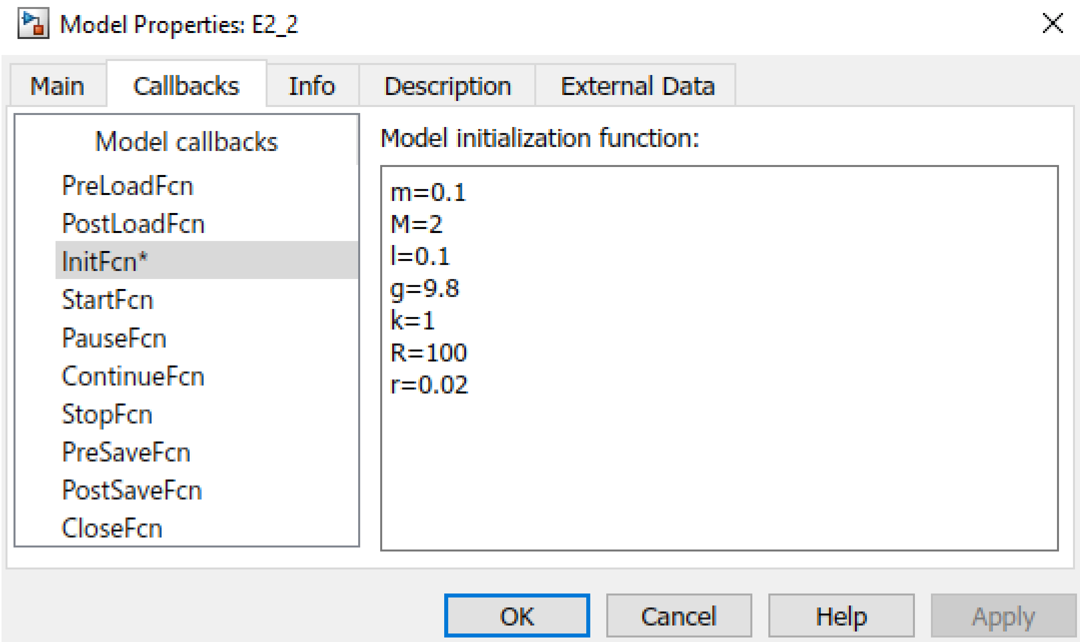
Implement the state-space of part.1-Q2 in Simulink (using state-space block), and then check the results of part.1-Q2. a, c, and d with the above commands.

Solution

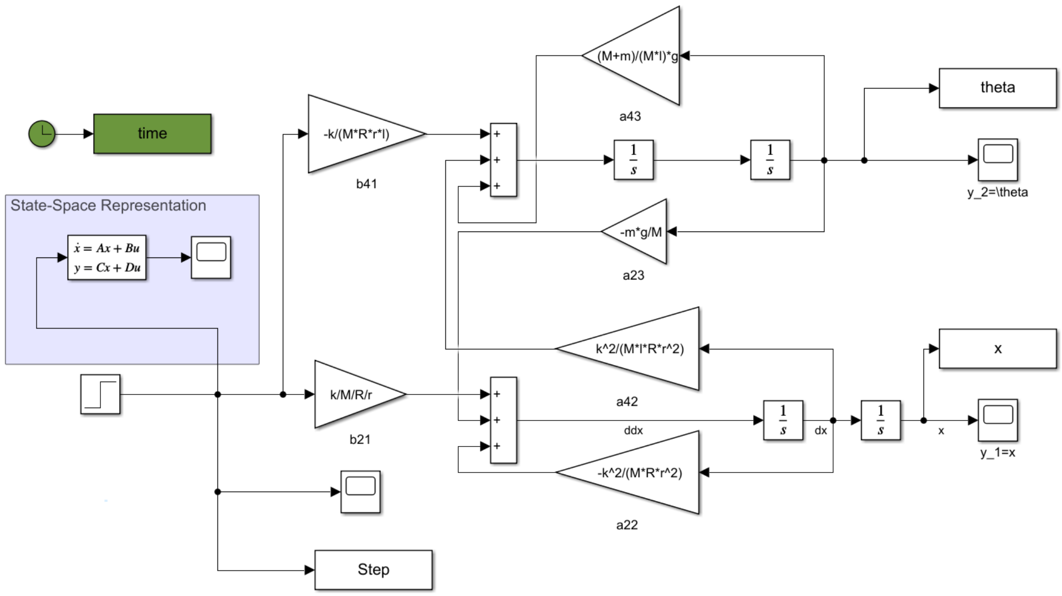
The state-space block of this system is shown in following figure,



in which the system parameters are as follows, defined in *Model Properties*.



the entire system is illustrated in following figure,



As an input, step function is considered and is illustrated here,

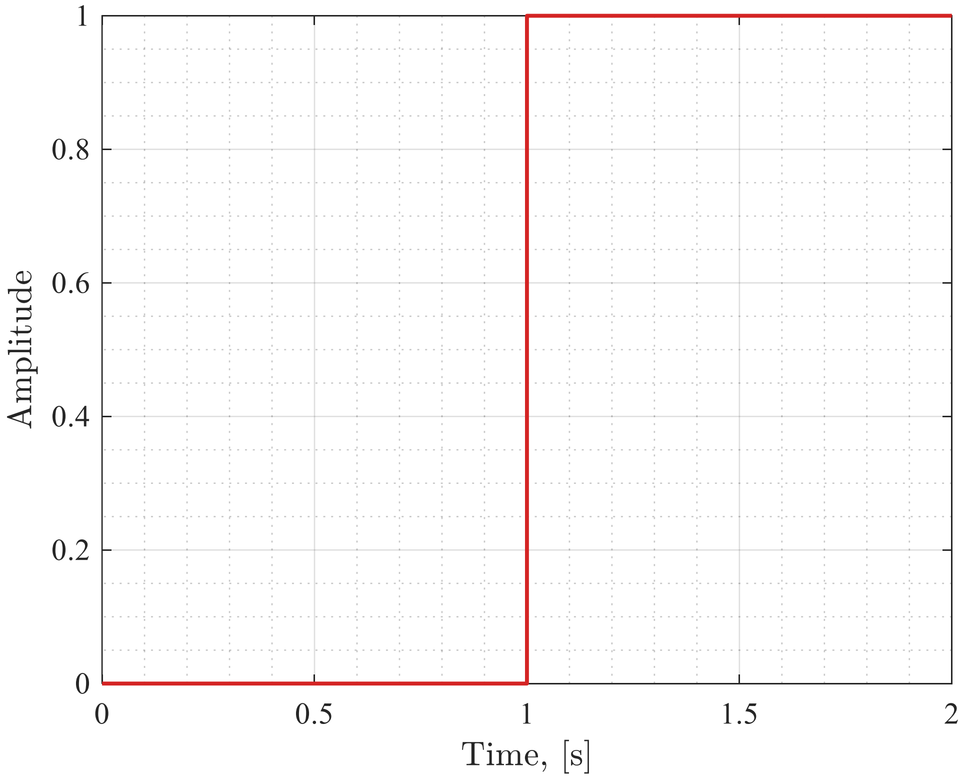


Figure 1 Step function as an input.

The findings of this study are visually represented in the accompanying figures. **Figure 2** showcases the angular displacement, θ, while **Figure 3** illustrates the displacement, x. Notably, the system exhibits linearity within a limited range of θ. However, over time, the limitations of the linear mathematical model become apparent, leading to inadequate prediction accuracy. Consequently, it is imperative to model the system in a nonlinear form to attain precise results.

This observation underscores the necessity of accounting for nonlinearity when describing the system's behavior. Nonlinear modeling techniques can capture the intricate dynamics that emerge as the system undergoes larger rotations.

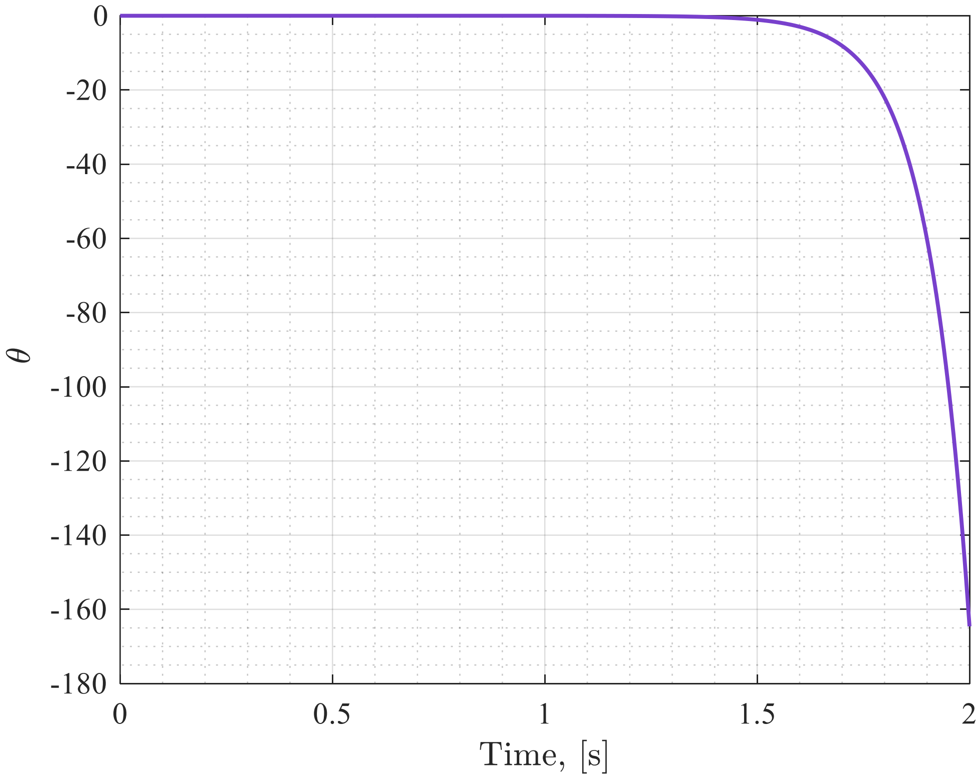


Figure 2 Variation of θ from initial condition is 2 seconds.

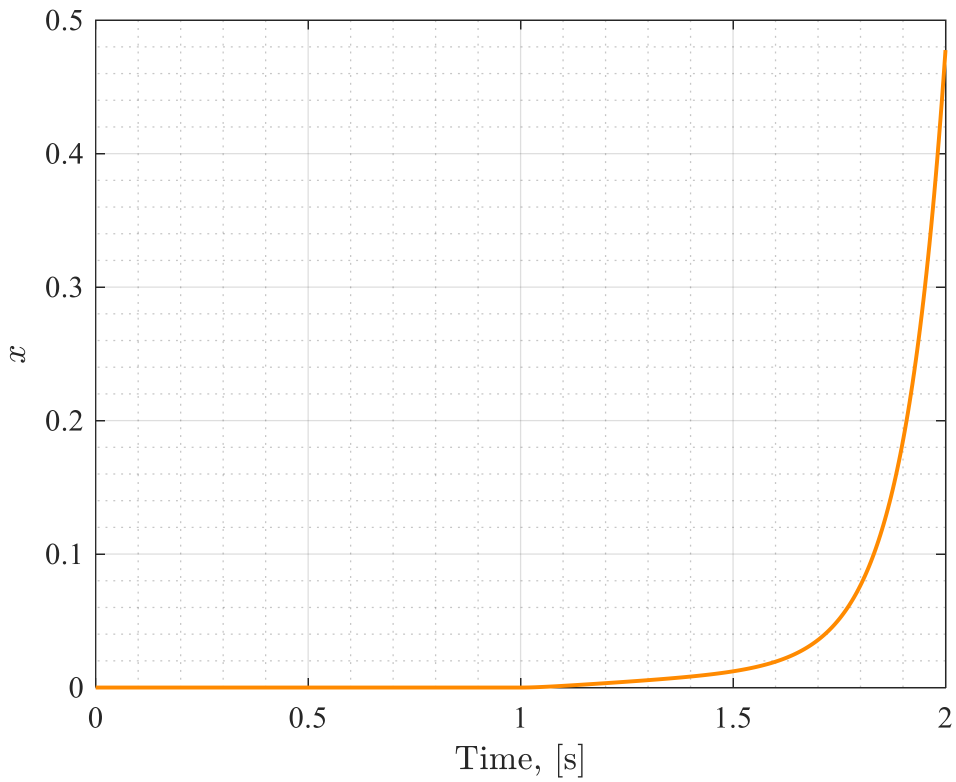


Figure 3 Variation of x from initial condition is 2 seconds.

MATLAB code for this problem is provided in **Table 3**.

Table 3 MATLAB code of part.1-Q2.

```
%%%%%%%%%%% Advanced Automatic Control%%%%%%%%%
%%%%%%%% Mechanical Engineering Department%%%%%
%%%%%%% Amirkabir University of Technology%%%%%
%%%%%%%%%%%%%%%%% 5 Nov. 23 %%%%%%%%%%%%%%%%%%
%%%%%%%%%%% Reza Nopour 402126924 %%%%%%%%%%%%
clc
clear
close all
%%
syms s
%% Initial
m=0.1;
M=2;
l=1;
g=9.8;
k=1;
R=100;
r=0.02;

%% State Space
A=[0,1,0,0;0,-k^2/M/R/r^2,-
m*g/M,0;0,0,0,1;0,k^2/M/l/R/r^2,(M+m)*g/M/l,0];
B=[0;k/M/R/r;0;-k/M/l/R/r];
C=[1,0,0,0;0,0,1,0];
D=[0;0];
%% SS2TF using definition
H=C*(s*eye(4)-A)^-1*B+D;
H=simplify(H);
pretty(H(1))
pretty(H(2))
%% SS2TF using functions
[b,a] = ss2tf(A,B,C,D,1)
Gp1 = tf (b(1,:) , a)
canon1=canon(Gp1,'companion')
Gp2 = tf (b(2,:), a)
canon2=canon(Gp2,'companion')
```

Then the result is

Table 4 MATLAB code of part.1-Q2.

```
                    2
               (5 s   - 49) 5
- ---------------------------------------
            3          2
  s (- 100 s   - 1250 s   + 1029 s + 12250)

                25 s
-----------------------------------
       3          2
- 100 s   - 1250 s   + 1029 s + 12250
```

```
b =

         0         0    0.2500         0   -2.4500
         0         0   -0.2500    0.0000    0.0000


a =

    1.0000   12.5000  -10.2900 -122.5000         0


Gp1 =

            0.25 s^2 - 2.45
  -----------------------------------
  s^4 + 12.5 s^3 - 10.29 s^2 - 122.5 s

Continuous-time transfer function.


canon1 =

  A =
              x1          x2          x3          x4
   x1          0           0           0  -1.805e-12
   x2          1           0           0       122.5
   x3          0           1           0       10.29
   x4          0           0           1       -12.5

  B =
       u1
   x1   1
   x2   0
   x3   0
   x4   0

  C =
           x1      x2      x3      x4
   y1       0    0.25  -3.125   39.18

  D =
       u1
   y1   0

Continuous-time state-space model.


Gp2 =

   -0.25 s^2 + 1.654e-24 s + 3.14e-16
  -----------------------------------
  s^4 + 12.5 s^3 - 10.29 s^2 - 122.5 s

Continuous-time transfer function.


canon2 =

  A =
```

```
               x1           x2           x3           x4
   x1           0            0            0   -1.805e-12
   x2           1            0            0        122.5
   x3           0            1            0        10.29
   x4           0            0            1        -12.5

  B =
       u1
   x1   1
   x2   0
   x3   0
   x4   0

  C =
           x1       x2       x3       x4
   y1       0    -0.25    3.125   -41.63

  D =
       u1
   y1   0

Continuous-time state-space model.
```

The observed distinction lies in the fact that near-zero (`-1.805e-12`) values in MATLAB results do not converge to absolute zero (`0`).

Problem 2.3 Description

Given the following open-loop single-input, single-output fourth–order linear time-invariant state equations, namely,

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \\ \dot{x}_4(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -962 & -126 & -67 & -4 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u(t)$$

$$y(t) = \begin{bmatrix} 300 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} + [0]u(t)$$

a) Find the associated open-loop transfer function *H (s)* with the command *ss2tf*, and also covert it to zero-pole description with the command *zpk*. (*tfdata* and *zpkdata* commands are used to review the results.)

```
clc
clear
close all
%% open-loop single-input, single-output
%fourth-order linear time-invariant state
A=[0,1,0,0;0,0,1,0;0,0,0,1;-962,-126,-67,-4];
B=[0;0;0;1];
C=[300,0,0,0];
D=[0];
%% Part 3.(a)
[b,a] = ss2tf(A,B,C,D);
H = tf (b , a); %associated open-loop transfer function H (s)
sys = zpk(H) %covert it to zero-pole description
[z,p,k] = zpkdata(H)
[num,den,ts] = tfdata(H)
```

The results are as follows,

```
H =

            300
  ----------------------------------
  s^4 + 4 s^3 + 67 s^2 + 126 s + 962

Continuous-time transfer function.


sys =

           300
  -------------------------------
  (s^2 + 2s + 26) (s^2 + 2s + 37)

Continuous-time zero/pole/gain model.


z =

  1×1 cell array

    {0×1 double}


p =

  1×1 cell array

    {4×1 double}


k =
```

```
  300


num =

  1×1 cell array

    {[0 0 0 0 300]}


den =

  1×1 cell array

    {[1 4.0000 67.0000 126.0000 962.0000]}


ts =

     0
```

b) Find the system characteristic polynomial coefficients with poly(A) and find the roots of the characteristic polynomial with roots(den).

```
%% Part 3.(b)
cpc=poly(A) % characteristic polynomial coefficients
rcp=roots(a) %roots of the characteristic polynomial
```

The output is,

```
cpc =

    1.0000    4.0000   67.0000  126.0000  962.0000



rcp =

  -1.0000 + 6.0000i
  -1.0000 - 6.0000i
  -1.0000 + 5.0000i
  -1.0000 - 5.0000i
```

c) Determine and plot the impulse response. (impulse(SysName))

```
%% Part 3.(c)
t = 0:0.01:10;
im = impulse(sys,t);
plot(t,im,'color',[0.25 0.80 0.54],'LineWidth',1.5)
ylabel('Amplitude','Interpreter','latex')
xlabel('Time, (seconds)','Interpreter','latex')
title('Impluse Response','Interpreter','latex')
set(gca,'FontSize',13)
```

```
set(gca,'fontname','Times New Roman')
grid minor
grid on
```

The plot is provided as follows (**Figure 4**),

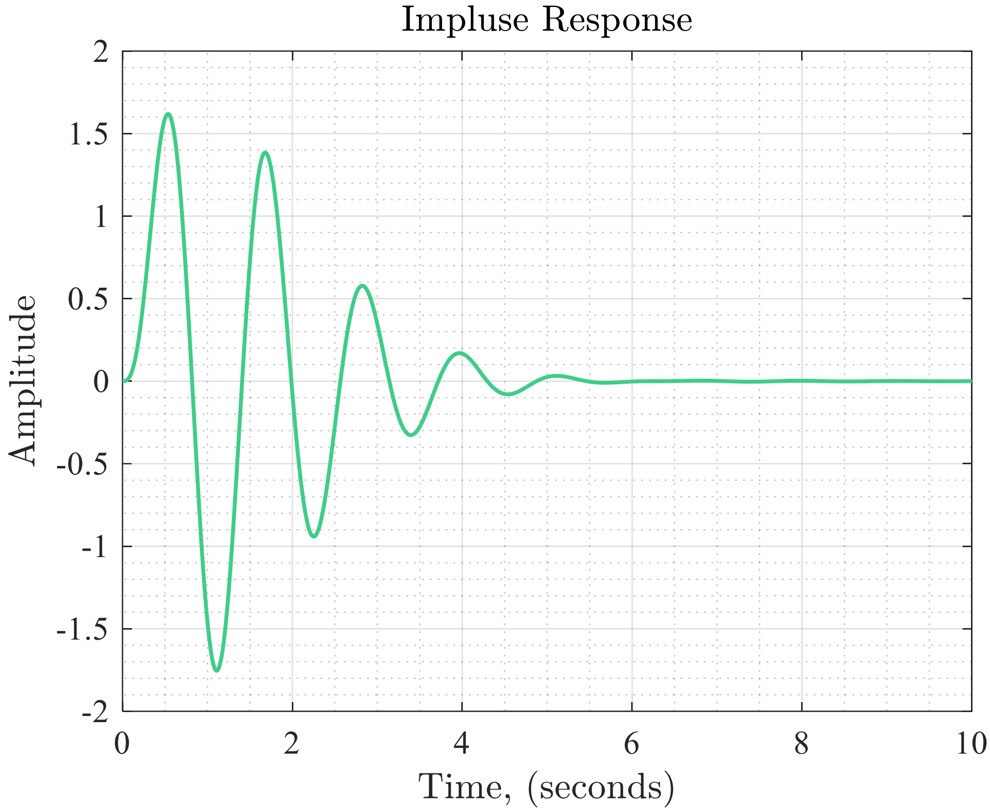


Figure 4 Impulse response of the system.

d) Determine and plot the unit step response for zero initial state. (step(SysName))

```
%% Part 3.(d)
t = 0:0.01:10;
st = step(sys,t);
plot(t,st,'color',[0.47 0.25 0.80],'LineWidth',1.5)
ylabel('Amplitude','Interpreter','latex')
xlabel('Time, (seconds)','Interpreter','latex')
title('Step Response','Interpreter','latex')
set(gca,'FontSize',13)
set(gca,'fontname','Times New Roman')
grid minor
grid on
```

The step response can be seen as follows (**Figure 5**),

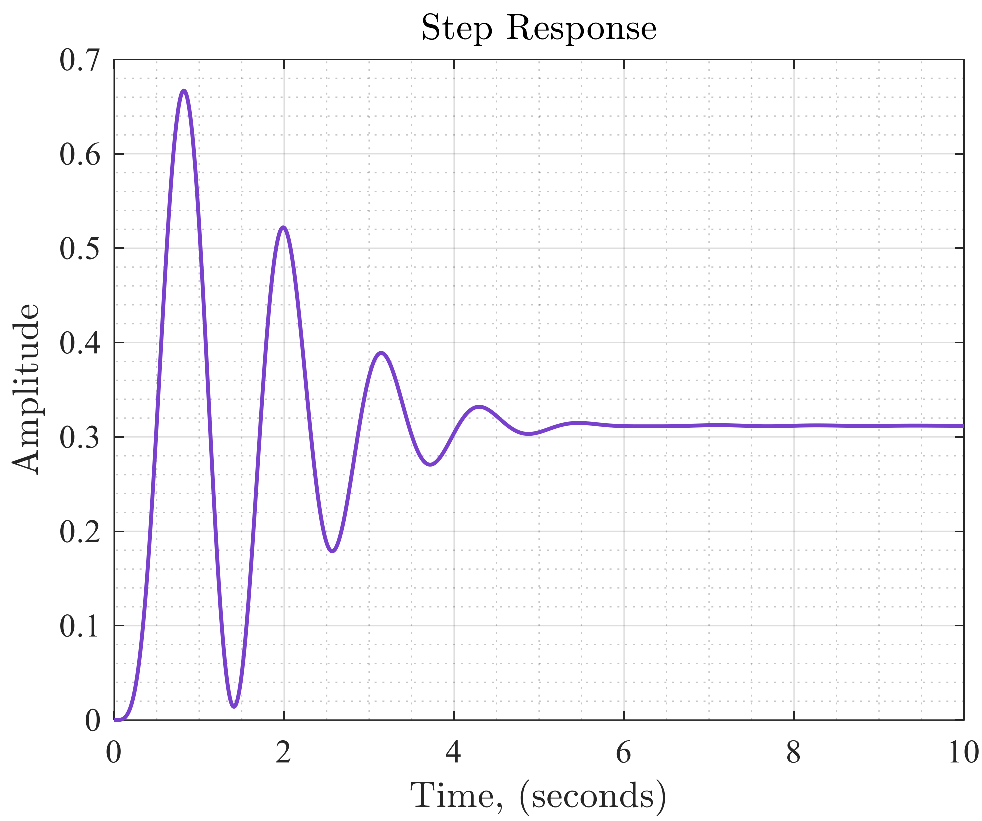
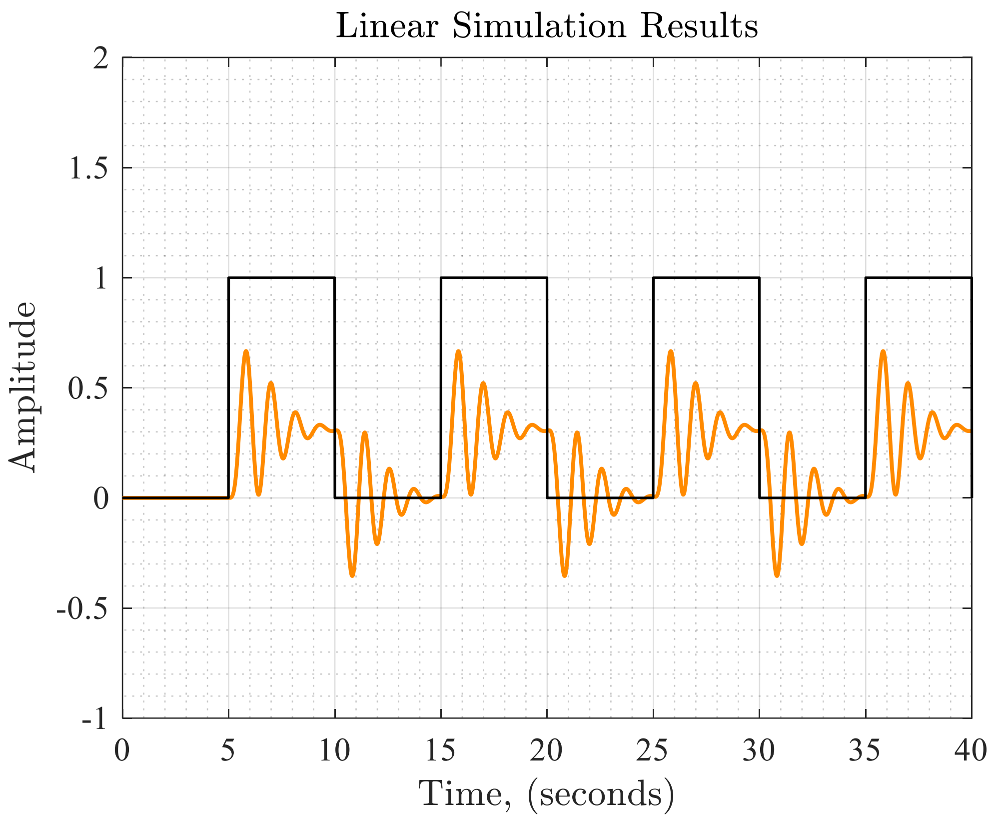


Figure 5 Step response of the system.

e) Determine and plot the zero input response given x_0 = [1, 2, 3, 4]^T.([y,t,x]= lsim(SysName,u,t,x0))

```
%% Part 3.(e)
x0 = [1, 2, 3, 4].';
[u,tt] = gensig("square",10,40,0.01);
[y,t,x]=lsim(sys,u,tt,x0);
plot(t,y,'color',[1.00 0.54 0.00],'LineWidth',1.5);
hold on
plot(tt,u,'color',"k",'LineWidth',1);
ylabel('Amplitude','Interpreter','latex')
xlabel('Time, (seconds)','Interpreter','latex')
title('Linear Simulation Results','Interpreter','latex')
set(gca,'FontSize',13)
set(gca,'fontname','Times New Roman')
grid minor
grid on
ylim([-1 2])
```

The response to the square wave is as follows,



f) Calculate the coordinate transformations and diagonal Canonical Form with [Tdcf,E] = eig(A).

```
%% Part 3.(f)
[Tdcf,E] = eig(A)
```

The coordinate transformations and diagonal canonical form can be obtained using *sysT = ss2ss(sys,T)* and *csys = canon(sys,'modal',condt)* functions. ss2ss performs the similarity transformation $z = Tx$ on the state vector x of a state-space model, and csys = canon(sys,'modal',condt) specifies an upper bound condt on the condition number of the block-diagonalizing transformation.

```
Tdcf =

   0.0021 + 0.0039i   0.0021 - 0.0039i   0.0041 + 0.0061i   0.0041 - 0.0061i
  -0.0252 + 0.0086i  -0.0252 - 0.0086i  -0.0348 + 0.0145i  -0.0348 - 0.0145i
  -0.0267 - 0.1600i  -0.0267 + 0.1600i  -0.0377 - 0.1886i  -0.0377 + 0.1886i
   0.9864 + 0.0000i   0.9864 + 0.0000i   0.9806 + 0.0000i   0.9806 + 0.0000i


E =

  -1.0000 + 6.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i
   0.0000 + 0.0000i  -1.0000 - 6.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i
```

```
  0.0000 + 0.0000i   0.0000 + 0.0000i  -1.0000 + 5.0000i   0.0000 + 0.0000i
  0.0      .0000i   0.0000 + 0.0000i   0.0000 + 0.0000i  -1.0000 - 5.0000i
```

g) [A1, B1, C1, D1] = ss2ss(A, B, C, D, Tdcf), use the first state-space data and the coordinate transformations that you found in section f and report the new state-space matrices.

```
%% Part 3.(g)
[A1, B1, C1, D1] = ss2ss(A, B, C, D, Tdcf);
```

The output is,

```
A1 =

   1.0e+05 *

  -0.0087 - 0.0139i  -0.0034 + 0.0007i  -0.0006 - 0.0004i  -0.0001 + 0.0001i
  -0.0608 + 0.0575i   0.0069 + 0.0160i  -0.0015 + 0.0032i   0.0005 + 0.0005i
   0.3485 + 0.2466i   0.0733 - 0.0506i   0.0173 + 0.0042i   0.0018 - 0.0027i
   0.8847 - 1.9902i  -0.3265 - 0.3158i   0.0037 - 0.0907i  -0.0155 - 0.0063i


B1 =

   0.0041 - 0.0061i
  -0.0348 - 0.0145i
  -0.0377 + 0.1886i
   0.9806 + 0.0000i


C1 =

   1.0e+04 *

  -2.6598 + 7.7578i   0.9815 + 1.2377i  -0.0111 + 0.3477i   0.0456 + 0.0247i


D1 =

     0
```

h) canon(Sys,'modal'). Can we use this command to find Jordan form for repeated or complex eigenvalues?

```
%% Part 3.(h)
canon(sys,'modal')
```

```
ans =

  A =
        x1   x2   x3   x4
   x1   -1    5    0    0
```

```
   x2   -5   -1    0    0
   x3    0    0   -1    6
   x4    0    0   -6   -1

  B =
               u1
   x1  2.101e-15
   x2      5.204
   x3          0
   x4      4.634

  C =
                x1          x2          x3          x4
   y1        1.048           0     -0.9809  -4.752e-16

  D =
       u1
   y1   0

Continuous-time state-space model.
```

According to Mathworks, Modal form is a diagonalized form that separates the system eigenvalues. In modal form, A or (A,E) are block-diagonal. The block size is typically 1-by-1 for real eigenvalues and 2-by-2 for complex eigenvalues. However, if there are repeated eigenvalues or clusters of nearby eigenvalues, the block size can be larger. However, this is not exactly the same as the Jordan form. The Jordan form is a special kind of matrix form that handles the case where eigenvalues are repeated. In the Jordan form, each block is either a Jordan block corresponding to a real eigenvalue or a combination of two blocks corresponding to a pair of complex conjugate eigenvalues.