EE 547 (PMP) Lab 5

Wednesday, February 4, 2015

Professor Linda Bushnell, UW EE

Demo Problem 1 (Lyapunov Stability) Given a system as (1). Please write a MATLAB script to:

$$\begin{aligned} \dot{x} &= \begin{bmatrix} -4 & -1 \\ 6 & -10 \end{bmatrix} \mathrm{x} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \mathrm{u} \\ y &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \mathrm{x} + \begin{bmatrix} 0.1 \\ 0.2 \end{bmatrix} \mathrm{u} \end{aligned} \tag{1}$$

(a) Compute the characteristic polynomial and eigenvalues of matrix A.

(b) Evaluate the Jordan form of this matrix ($\bar{A}$) and corresponding transformation matrix Q.

(c) Check if the system is asymptotically stable by solving the Lyapunov equation:

$$A^T P + PA = -Q$$

$$Q = \begin{bmatrix} 7 & 5 \\ 5 & 7 \end{bmatrix}$$

where Q is a symmetric positive-definite matrix and P is a unique symmetric positive-definite matrix.

(d) Find the transfer function of this system.
(Note: The output is a 2-by-1 column vector.)

(e) Please use **initial** function to evaluate zero-input response with x_0 = [-3; 2]. Plot outputs as a function of time.

(f) Please use **step** evaluate step response of this system. Plot outputs as a function of time.

Individual Problem 1 Consider a system as (2).

$$\begin{aligned} \dot{x} &= \begin{bmatrix} -1 & 10 \\ 0 & 1 \end{bmatrix} x + \begin{bmatrix} -2 \\ 0 \end{bmatrix} u \\ y &= \begin{bmatrix} -2 & 3 \end{bmatrix} x \end{aligned} \tag{2}$$

(a) Please determine the transfer function of this system. Is the transfer function a proper rational function?

(b) Is the system BIBO stable?

(c) Evaluate the step input response of this system. The input is a unit step input: u(t) = 1, t≥0 and the initial condition is zero.

Solution of Demo Problem 1 (Lyapunov Stability)

(a) Compute the characteristic polynomial and eigenvalues of matrix A by MATLAB:

Characteristic Polynomials: $s^2 + 14s + 46$

Eigenvalues:
λ_1 =-5.2679
λ_2= -8.7321

(b) The Jordon form $\bar{A}$ and transformation matrix T are evaluated and attached as below.

$$\bar{A} = \begin{bmatrix} -8.7321 & 0 \\ 0 & -5.2679 \end{bmatrix}$$

$$T = \begin{bmatrix} 0.2113 & 0.7887 \\ 1 & 1 \end{bmatrix}$$

(c) Solve the Lyapunov equation by MATLAB **lyap** function.
P = **lyap**(A,Q)

$$P = \begin{bmatrix} 0.7213 & 0.6149 \\ 0.6149 & 0.7189 \end{bmatrix}$$

P is a symmetric matrix. We can check if it is positive definite by taking eigenvalues of P.

Eigenvalues of matrix P:
λ_1 = 0.1052
λ_2= 1.3350

(d) To find the transfer function of this system, we have to build up state-space of this system first. Then, use numerators and denominator to build up transfer functions.

transfer function 1 : $\frac{0.1s^2+2.4s+13.6}{s^2+14s+46}$

transfer function 2 : $\frac{0.02s^2+1.28s+10.92}{s^2+14s+46}$

(e) To evaluate zero-input response with x_0 = [-3; 2], the syntax of **initial** is as following

[y,t,x]=**initial**(ss1,xini1,tspan(end));

% y is output; t is simulation time; x is state trajectories

(f) To evaluate step response of this system, the syntax of step is:

[response,t1] = **step**(ss1,tspan(end))

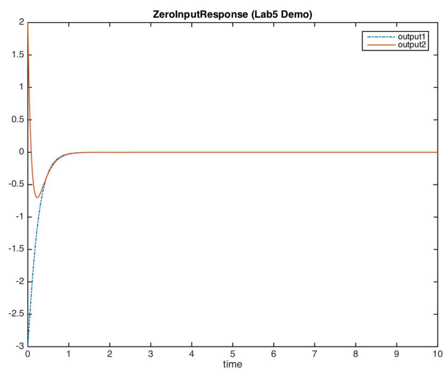

Figure 1 Zero Input Responses of the system

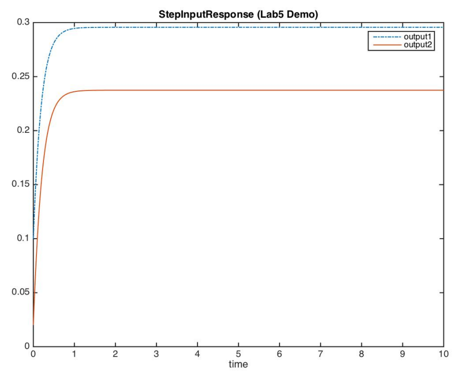

Figure 2 Step Input Responses of the system

Solution of Individual Problem 1

(a) To find the transfer function of this system, we can build up state-space of this system first. Then, use numerators and denominator to derive transfer functions.

Transfer function: $\frac{4}{s+1}$

It is a proper rational function.

(b) Since the only one pole of the transfer function is -1. It is BIBO stable.

(c) To evaluate step response of this system, we use **step** function as below.
[response1,t1] = **step**(ss1,tspan(end));

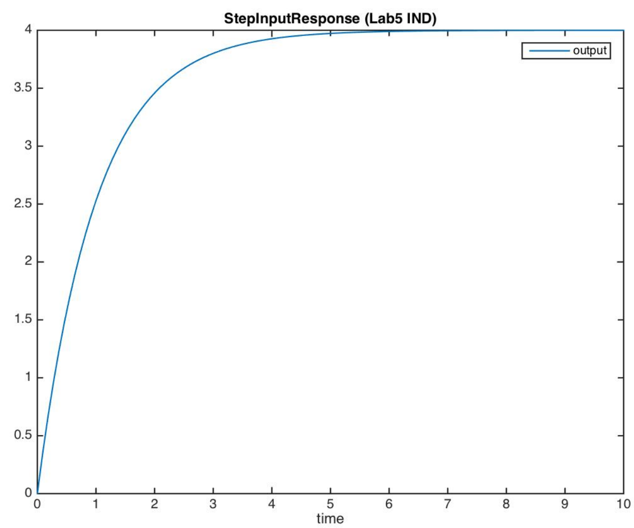

Figure 3 Step Input Responses of the system

MATLAB Demo Code:

```
%% Lab 5: Eigenvalues, Jordan form, Matrix function
close all;
clear all;


%% Define Parameters
syms p11 p12 p13 p21 p22 p23 p31 p32 p33
xini1 = [-3;2];
tspan = 0:0.01:10
f = 1e2;
u = sin(2*pi*f*tspan)'

%% Define Matrices
A = [-4 -1; 6 -10];
B = [1;1]
C = eye(size(A));
D = [0.1;0.02];
Q = [7 5;5 7];

%% Eigenvalues and characteristic polynomial
ChaPoly = poly(A);  % return the characteristic polynomials of matrix A
rts = roots(ChaPoly); % Another way to evaluate eignevalues

%% Jordan form and transformation matrix
[T,JA] = jordan(A)  % Q is transformation matrix and JA is Jordon form


%% Lyapunov Equation A'P+PA = -Q
%P = lyap(A,Q)
P = lyap(A',Q)
eigP = eig(P)

if eigP > 0
    fprintf('This is asymptotically stable.\n\n')
else
    fprintf('This is not asymptotically stable.\n\n')
end

%% State-space and transfer function formulation
ss1 = ss(A,B,C,D);
[num,den] = ss2tf(A,B,C,D);
tf1 = tf(num(1,:),den);
tf2 = tf(num(2,:),den);

%% Zero-initial state
[y,t,x]=initial(ss1,xini1,tspan(end)); % y is output; t is simulation time; x
is state trajectories
figure(1)
plot(t,y(:,1),'-.',t,y(:,2));
xlabel('time')
legend('output1','output2')
title1 = 'ZeroInputResponse (Lab5 Demo)'
title(title1)
print('-djpeg',title1)
```

```
%% step response
figure(2)
[response,t1]=step(ss1,tspan(end))
plot(t1,response(:,1),'-.',t1,response(:,2));
xlabel('time')
legend('output1','output2')
title2 = 'StepInputResponse (Lab5 Demo)'
title(title2)
print('-djpeg',title2)
```

MATLAB IND Code

```
%% Lab 5: Eigenvalues, Jordan form, Matrix function
%Developer: HRLin
close all;
clear all;
clc;

%% Define Parameters and Matrices
tspan = 0:0.1:10;
A = [-1 10; 0 1];
B = [-2;0];
C = [-2 3];
D = [0];

%% Eigenvalues and characteristic polynomial
ChaPoly = poly(A);  % return the characteristic polynomials of matrix A
rts = roots(ChaPoly); % Another way to evaluate eignevalues

%% Jordan form and transformation matrix
[T,JA] = jordan(A);  % Q is transformation matrix and JA is Jordon form

%% State-space and transfer function formulation
ss1 = ss(A,B,C,D);
[num,den] = ss2tf(A,B,C,D);
tf1 = tf(num,den);
tf11 = minreal(tf(num,den));

%% BIBO stability
poletf1 = pole(tf11);

if poletf1 < 0
    fprintf('This is BIBO stable.\n\n')
else
    fprintf('This is not BIBO stable.\n\n')
end

%% Zero-initial state
% [y,t,x]=initial(ss1,xini1,tspan(end)); % y is output; t is simulation time;
x is state trajectories
% figure(1)
% plot(t,y);
% xlabel('time')
% legend('output')
% title1 = 'ZeroInputResponse (Lab5 IND)'
% title(title1)
% print('-djpeg',title1)

%% step response
figure(2)
[response,t1]=step(ss1,tspan(end));
plot(t1,response);
xlabel('time')
legend('output')
title2 = 'StepInputResponse (Lab5 IND)'
title(title2)
```

```
print('-djpeg',title2)
```