Xi'an Jiaotong-Liverpool University
西交利物浦大学

Department of Electrical and Electronic Engineering

EEE220 Instrumentation and Control System

2018-2019, Semester 2

Experiment 2:

Control System CAD and CAS using Matlab

Name:___________________

Student ID:______________

Date:____________________

Part I: Examples

This part is to introduce how to use Matlab Control Toolbox for control system computer aided design and simulation. It is suggested to go through the examples in Part 1 before you start working on the assignment tasks.

1.1 Input a system described by a transfer function

To construct a system model in Matlab, the command "tf" can be used.

Example 1.1.1

Input transfer function

$$\frac{s+1}{s^4+2s^3+3s^2+3s+10}$$

```
» sys1=tf([1 1], [1 2 3 3 10])
Transfer function:
s + 1
------------------------------
s^4 + 2 s^3 + 3 s^2 + 3 s + 10
»
```

Example 1.1.2 Input transfer function

$$\frac{s^2+s+1}{s^4+2s^3+5s+2}$$

```
» sys1=tf([1 1 1], [1 2 0 5 2])
Transfer function:
s^2 + s + 1
---------------------
s^4 + 2 s^3 + 5 s + 2
>>
```

Example 1.1.3

Input transfer function

$$\frac{s+1}{s^4+2s^3+3s^2+3s+10}\cdot\frac{10}{s^2+3}$$

```
>> sys1=tf([1 1], [1 2 3 3 10])
Transfer function:
```

```
s + 1
------------------------------
s^4 + 2 s^3 + 3 s^2 + 3 s + 10
>> sys2=tf([10], [ 1 3])
Transfer function:
10
-----
s + 3
>> sys=sys1*sys2
Transfer function:
10 s + 10
----------------------------------------
s^5 + 5 s^4 + 9 s^3 + 12 s^2 + 19 s + 30
>>
```

1.2 Find the system zeros and poles using Matlab

The commands used for this task are "pole" and "zero". From the signs of system poles, you can find out if the system is stable.

Example 1.2.1

For system

$$G(s) = \frac{1}{s^2 + s + 1}$$

Using MATLAB, the system has the poles that can be obtained as follows:

```
>> pole(sys)
ans =
-0.5000 + 0.8660i
-0.5000 - 0.8660i
>>
```

Example 1.2.2

Find the following system zeros

$$G(s) = \frac{2s^2 + s + 3}{s^4 + 2s^3 + 5s^2 + 6s + 7}$$

```
>> sys=tf([2 1 3],[1 2 5 6 7])
Transfer function:
```

```
2 s^2 + s + 3
------------------------------
s^4 + 2 s^3 + 5 s^2 + 6 s + 7
>> zero(sys)
ans =
-0.2500 + 1.1990i
-0.2500 - 1.1990i
>>
```

1.3 Obtain the system model in pole-zero format

Example 1.3.1

Convert the system model

$$G(s) = \frac{1}{s^4 + s^3 + 3s^2 + 2s + 6}$$

into a pole-zero format.

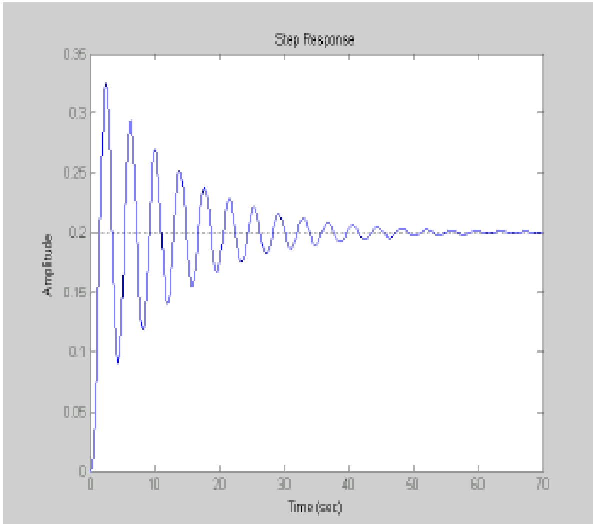
```
>> sys=tf([1], [1 1 3 2 6])
Transfer function:
1
---------------------------
s^4 + s^3 + 3 s^2 + 2 s + 6
>> zpk(sys)
Zero/pole/gain:
1
---------------------------------------------
(s^2 + 1.969s + 2.609) (s^2 - 0.9693s + 2.3)
>>
```

1.4 Obtain system step and impulse input responses using MATLAB

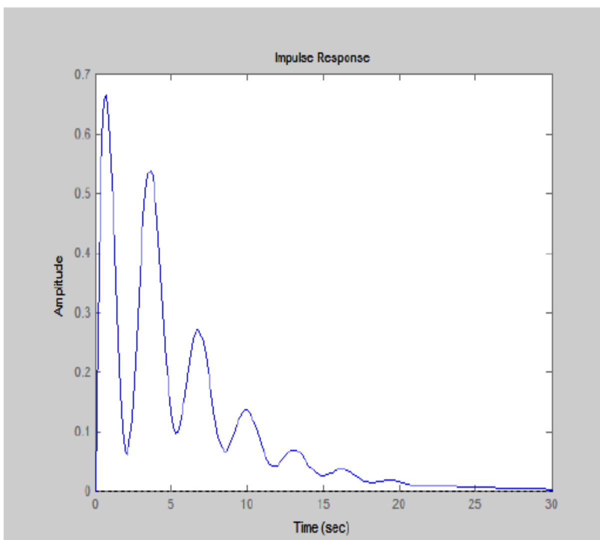
The command used for obtaining the step/impulse input response is "step() / impulse()". Step/impulse input response is the system response when the input is a step/impulse function.

Example 1.4.1

```
>> sys1=tf([1], [1 2 3 5])
Transfer function:
1
---------------------
s^3 + 2 s^2 + 3 s + 5
>> step(sys1)
```



>> impulse(sys)



Obtain system ramp input responses using MATLAB

<u>Example 1.4.2</u>

Find the ramp input response of the system

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

>> sys1=tf([1], [1 2 3 5])

Transfer function:

1

----

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

```
>> sys2=tf([1], [1 0])
Transfer function:
```
$\frac{1}{s}$
```
>> step(sys1*sys2)
>> title('ramp input response')
```

1.5 Obtain the closed-loop system transfer function using Matlab

The command used is "feedback"

You can obtain the explanation to the command from on-line help

Example 1.5.1

A system has a front path transfer function as

$$G(s)=\frac{1}{s^2+s+1}$$

and feedback path transfer function as

$$G(s)=\frac{1}{s+1}$$

then the closed-loop system transfer function is

```
> sys1=tf([1],[1 1 1])
Transfer function:
1
-----------
s^2 + s + 1
>> sys2=tf([1],[1 1])
Transfer function:
1
-----
s + 1
>>sys_closed_loop=feedback(sys1,sys2,-1)
Transfer function:
s + 1
---------------------
s^3 + 2 s^2 + 2 s + 2
```

You can find the closed-loop system transfer function manually to check if the computer gives you a correct answer. Furthermore, you may confer the simulation parts at the end of each chapter in text book.

Part II: Assignments

Hand written reports are NOT acceptable. You are advised to undertake this assignment using the MATLAB software and the Control Systems Toolbox.

Problem 1

Consider the differential equation

$$\ddot{y} + 3\dot{y} + 2y = u$$

Where $y(0) = \dot{y}(0) = 0$ and $u(t)$ is a unit step. Determine the solution $y(t)$ analytically and verify by co-plotting the analytic solution and the step and impulse response obtained with the **step** and **impulse** function.

USE functions : [**tf, step, impulse, plot**].

Problem 2

Consider the block diagram in Fig. P2.

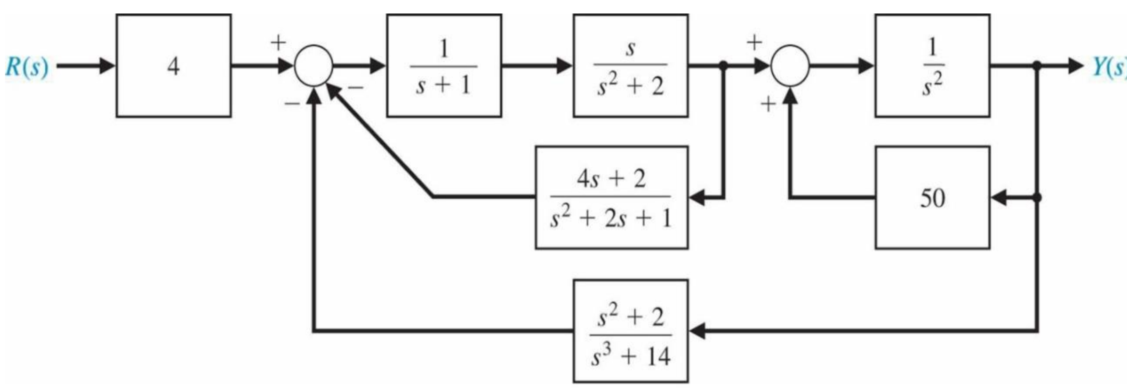


Fig. P2

(a) Use an m-file to reduce the block diagram in Fig. P2, and compute the closed-loop transfer function.

(b) Generate a pole-zero map of the closed-loop transfer function in graphical form using the **pzmap** function.

(c) Determine explicitly the poles and zeros of the closed-loop transfer function using the **pole** and **zero** functions and correlate the results with the pole-zero map in part (b).

USE functions : [**tf, feedback, series**]

Problem 3

Determine a state variable representation for the following transfer functions using the **ss** function.

(a) $G(s) = \frac{1}{s+12}$

(b) $G(s) = \frac{s^2+5s+3}{s^3+8s+5}$

(c) $G(s) = \frac{s^2+4s+1}{s^4+3s^2+3s+1}$

Problem 4

Consider the system

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -3 & -2 & -5 \end{bmatrix} x + \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} x$$

(a) Using the **tf** function, determine the transfer function $Y(s)/U(s)$.

(b) Plot the response of the system to the initial condition $x(0) = \begin{bmatrix} 0 & -1 & 1 \end{bmatrix}^T$ for $0 \le t \le 10$.

(c) Compute the state transition matrix using the **expm** function, and determine $x(t)$ at $t = 10$ for the initial condition given in part (b).

USE functions: [**tf, lsim, expm**]

Problem 5

Consider the closed loop transfer function

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

(a) Using the Routh-Hurwitz method, determine whether the system is stable. If it is not stable, how many poles are in the right-half plane?

(b) Compute the poles of $T(s)$ and then verify the result in part (a).

(c) Plot the unit step response, and discuss the results.

Problem 6

A control system is shown in Fig.P6. Sketch the root locus, and select K so that the step response of the system has an overshoot of less than 10 % and the settling time (with 2% criterion) is less than 4 seconds.

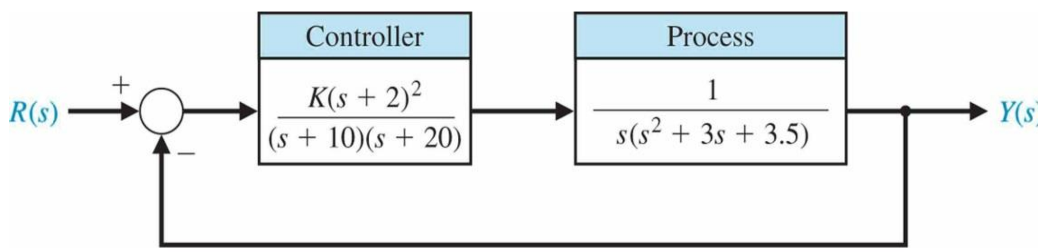


Fig. P6

Problem 7

Consider the feedback system shown in Fig.P7.

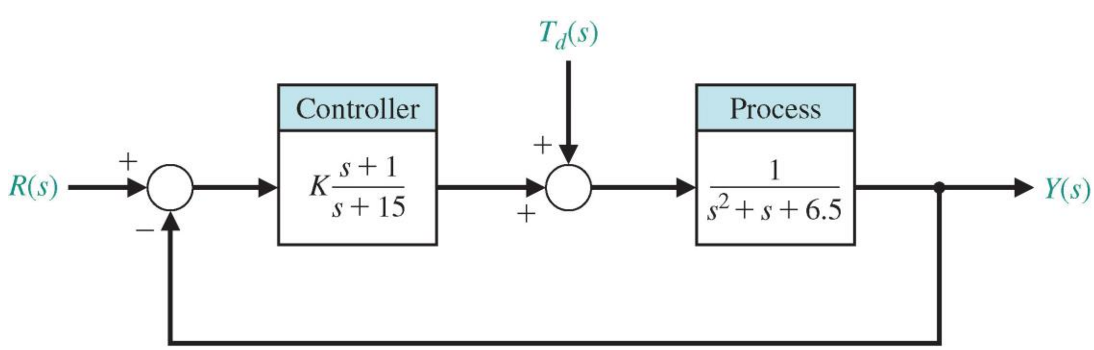


Fig. P7

(a) Determine the closed-loop transfer function $T(s) = Y(s)/U(s)$.

(b) Plot the response of the closed-loop system for $K = 5, 10,$ and 50 to unit step input.

(c) When the controller gain is $K = 10$, determine the steady-state value of $y(t)$ when the disturbance is a unit step, that is, when $T_d(s) = \frac{1}{s}$ and $R(s) = 0$.

Problem 8

Using the **rlocus** function, obtain the root locus for the following system shown in Fig. P8 when $0 \leq K < \infty$.

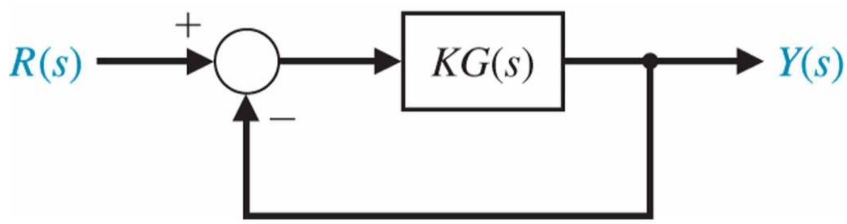


Fig. P8

(a) $G(s) = \frac{30s+12}{s^3+14s^2+43s+30}$

(b) $G(s) = \frac{s^5+4s^4+10s^2+6s+4}{s^6+7s^5+4s^4+s^3+s^2+10s+1}$

Problem 9

An autopilot designed to hold an aircraft in straight and level flight shown in Fig. P9.

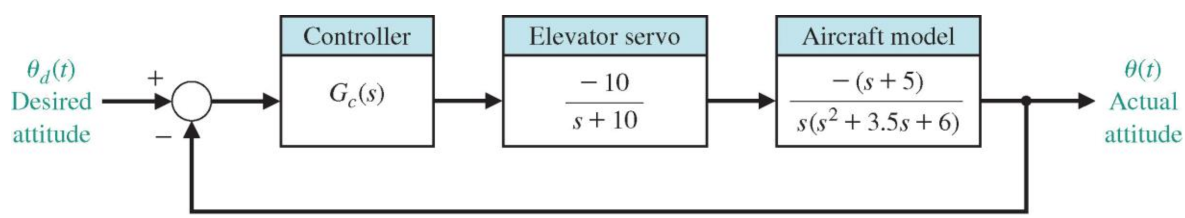


Fig. P9

(a) Suppose the controller is a constant gain controller given by $G_c(s) = 2$. Using the **lsim** function, compute and plot the ramp response for $\theta_d(t) = at$ where $a = 0.5^o/s$. Determine the attitude error after 10 seconds.

(b) If we increase the complexity of the controller, we can reduce the steady-state tracking error. Suppose we replace the constant gain controller with the more sophisticated PI controller

$$G_c(s) = K_1 + \frac{K_2}{s} = 2 + \frac{1}{s}.$$

Repeat the simulation in part (a), and compare the steady-state error obtained.

Problem 10

Consider the feedback control system in Fig. P10. We have three potential controllers for our system:

1. $G_c(s) = K$ (proportional controller)
2. $G_c(s) = K/s$ (integral controller)
3. $G_c(s) = K(1 + 1/s)$ (proportional integral (PI) controller)

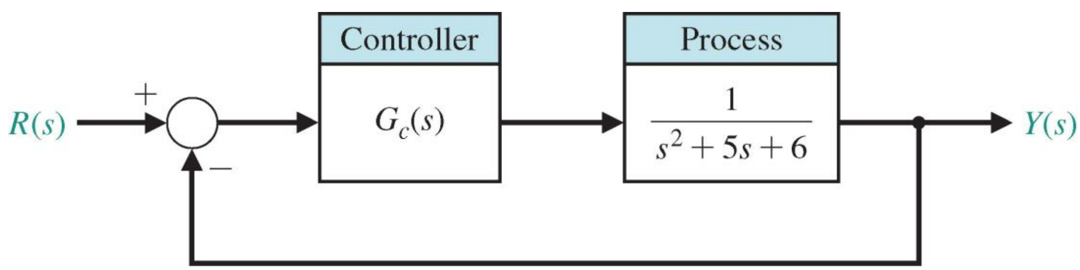


Fig. P10

The design specifications are $T_s \leq 10s$ and $P.O. \leq 10\%$ for a unit step input.

(a) For the proportional controller, develop and m-file to sketch the root locus for $0 \leq K < \infty$, and determine the value of K so that the design specifications are satisfied.

(b) Repeat part (a) for the integral controller.

(c) Repeat part (a) for the PI controller.

(d) Co-plot the unit step response for the closed-loop system with each controller designed in parts (a)-(c).

(e) Compare and contrast the three controllers obtained, concentrating on the steady-state errors and transient performances.