The University of Sheffield
Department of Automatic Control and Systems Engineering

Laboratory Session: ACS214/2214

Computer Aided Control Systems Analysis and Design Using MATLAB

1. Laboratory Objective

This laboratory is designed to familiarise the student with computer aided design techniques and to the valuable nature of software packages in the analysis and design of control systems. Several such packages exist, for example MATRIX $_X$ ACSL, CODAS, and MATLAB. We shall restrict attention to MATLAB-SIMULINK in this laboratory exercise. This package will be used to analyse a third-order system in the frequency and time domains, as well as in root locus terms. Once familiarity with the package has been gained, it will be used to design suitable controllers so that the system satisfies the required performance specifications. It is worth noting that when using the package, the design is actually done by the individual, but the package allows the results of any design to be rapidly available, and hence speed up the overall procedure, as well as makes it less tiresome.

2. Familiarisation of the MATLAB Package

Investigate the system whose transfer function is given by $G(s)$, where

$$G(s) = \frac{16500}{s(s+20)(s^2+10s+106)}$$

on the package. You are commended to follow the procedure outlined on the next page to familiarise yourself with the package. Modify the value of the gain to 8000 and notice the changes in the behaviour of the system.

2.1 Example

The example in question is the transfer function

$$G(s) = \frac{16500}{s(s+20)(s^2+10s+106)}$$

Enter this according to the steps below. Note that ‘>>’ is the prompt while in the MATLAB command mode.

Command	Description
>>num=[16500]	Input $G(s)$ numerator.
>>den=[1 30 306 2120 0]	Input $G(s)$ denominator.
>>[zeros, poles, gain]=tf2zp(num,den)	Show zeros, poles, and gain of the system.
>>t=0:2.5:60	Set up a 24 element vector at which to evaluate the response.
>>response=step(num,den,t)	Calculate the response to a unit step.
>>figure(1)	
>>plot(t,response),title('Step,response')	Plot response.
>>grid	Display grid lines.
>>who	Display assigned variables
>>w=logspace(-1,2,50);	Set up a vector of 50 frequencies from 10^{-1} to 10^{2}. Note ";" stops display of w.
>>[re,im]=nyquist(num,den,w);	Calculate points for Nyquist plot.
>>figure(2)	
>>plot(re,im)	Plot and display Nyquist diagram.
>>axis([-1.5,0.5, -5, 1])	Set [Xmin, Xmax, Ymin, Ymax] for a plot.
>>grid	
>>axis auto	Return to auto scaling.
>>figure(3)	
>>[mag,phase] = bode(num,den,w);	Calculate points of Bode plot.
>>mag=20*log10(mag)	Change magnitude into decibels.
>>subplot(211), semilogx(w,mag);	Plot magnitude part of Bode.
>>grid	
>>subplot(212), semilogx(w,phase)	Plot phase part of Bode.
>>clf	Return to single graphics window.
>>k=logspace(-2,1,50);	Set up vector of feedback gains.
>>figure(4)	
>>locus=rlocus(num,den,k);	Calculate the root locus.
>>plot(locus,'x'),xlabel('re'),ylabel('im')	Plot root locus.
>>[a,b,c,d]=tf2ss(num,den)	Calculate the state space form.
>>aa=a-b*c	Obtain the closed loop SS matrix.
>>figure(5)	
>>clresponse = impulse(aa,b,c,0,1,t);	Calculate the closed-loop impulse response.
>>plot(t,clresponse)	Plot impulse response.

To find the gain and phase margins:

```
>>[mag,phase]=bode(num,den,w);
>>[gain_margin,phase_margin,wcg,wcp]=margin(mag,phase,w);
```

Professor M Mahfouf
Dept. of ACSE, The University of Sheffield

3 Controller Design

For this part of the laboratory you should concentrate on the closed-loop system shown in Figure 1.

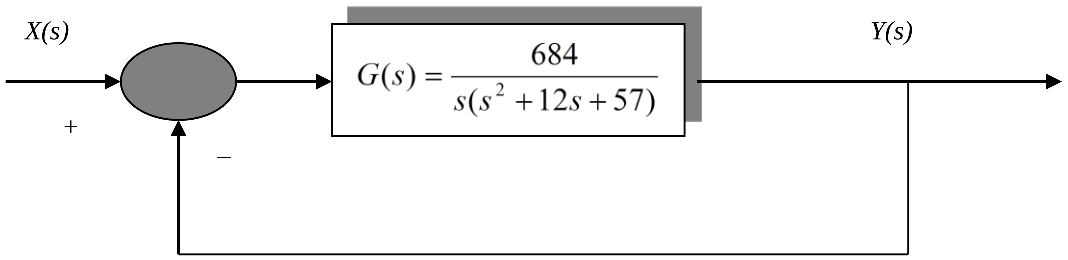

Figure 1

Convince yourself that the system is unstable and that some form of compensation is required. This leads to the block diagram shown in Figure 2.

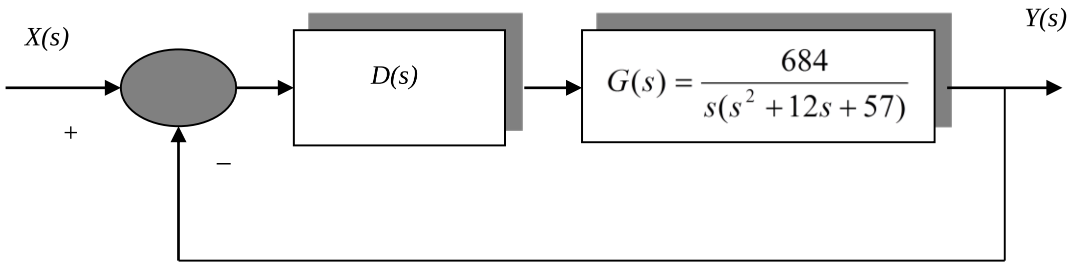

Figure 2

Design a suitable controller *D(s)* with the aid of the package and assess the performance of the compensator. It is recommended that students consult their relevant material and/or references listed under Section 6 so that the design can be performed correctly. Your design should satisfy the following specifications:

i. Percentage overshoot < 15%.
ii. Rise-time < 100 msec.
iii. Settling time < 500 msec.
iv. Zero steady-state error to a step.

You should be able to explain fully the design procedure you are using and justify the decisions made.

Professor M Mahfouf
Dept. of ACSE, The University of Sheffield

3.1 Design Objectives

We normally require the compensated system to have:

i. Good transient behaviour (fast and damped responses).
ii. Good steady-state behaviour (small or no steady-state errors).

These can be specified in the frequency or time-domains as follows:

i. Frequency domain: Gain and Phase margins and bandwidth.
ii. Time-domain: percentage of overshoot, rise-time, settling-time, etc.

4 Digital Analysis

Use the package to analyse the system in Part Section 3 in digital terms. The package can be used to digitise $G(s)$ by using a suitable sampling interval. Such a value can be obtained by finding the bandwidth ($f_T Hz)$ of the closed-loop system and choosing the sampling interval as

$$T = \frac{1}{20 f_T}$$

The whole procedure in Section 3 can be repeated for the digital system, except that only one design methodology is required, instead of two. Compare the performance of the digital controller with that obtained in Section 3 using $D(s)$.

4.1 Example

Assuming a sampling period *T* to be 0.2 sec, transform

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

into $G(z)$.

Command	**Description**
>>num=[0 0 1];	Input $G(s)$ numerator
>>den=[1 1 0];	Input $G(s)$ denominator
>>format long	Set the computations to be done in double precision
>>[numz,denz]=c2dm(num,den,0.2);	Obtain the discrete-time transfer function

Check that the following transfer function in *"z"* is obtained:

$$G(z) = \frac{0.01873z + 0.01752}{z^2 - 1.8187z + 0.8187}$$

5 Conclusion

After completing this laboratory, the student should fully appreciate the merits of CAD packages. She/he should have a sound understanding of the analysis of control systems and designing controllers using a variety of approaches.

6 References

[1] D'Azzo ad Houpis: Linear Control Systems Analysis and Design, McGraw-Hill, 1981.
[2] O.Elgerd: Control Systems Theory, McGraw-Hill, 1967.
[3] F.Raven: Automatic Control Engineering, McGraw-Hill, 1978.
[4] Philips and Harbor: Feedback Control Systems, Prentice-Hall, 1988.
[5] Van de Vegte: Feedback Control Systems, Prentice-Hall, 1986.
[6] Kuo: Digital Control Systems, Holt-Saunders, 1980.
[7] Franklin and Powell: Digital Control and Dynamic Systems, Addison-Wesley, 1980.

Professor M Mahfouf
Dept. of ACSE, The University of Sheffield

Guidelines to Designing Compensators in the Continuous Domain: ACS214/ACS2214

Phase-Lead and Phase-Lag Compensators

- Given time-domain performance specifications, one needs to use the following formula to obtain their equivalents in the frequency domain:

$$O_{\max} = e^{-\frac{\pi\zeta}{\sqrt{1-\zeta^2}}}$$

$$\phi_M \cong \tan^{-1}\left\{\frac{2\zeta}{(\sqrt{1+4\zeta^4}-2\zeta^2)^{0.5}}\right\}$$

where,

ϕ_M : Phase margin.

ζ : Damping ratio.

ω_g : Gain crossover frequency.

ω_n : Natural angular frequency.

- After having obtained the values for the phase margin and the gain crossover frequency, compare them to those obtained using the uncompensated system and make a decision whether to use a phase-lead or a phase-lag compensator.

- For instance, the transfer function for a phase-lead compensator is written as follows:

$$G_c(s) = \frac{1+Ts}{1+\alpha Ts}$$

$$\alpha < 1$$

Use the well-known phase-lead table to determine the value of α (given the phase margin required). The value of T can be calculated as follows:

$$T = \frac{1}{\omega_g' \sqrt{\alpha}}$$

where ω_g' is the new gain crossover frequency.

Professor M Mahfouf
Dept. of ACSE, The University of Sheffield

Phase-Lead Table

α	0.05	0.1	0.2	0.24	0.3	0.5	1
$\lvert G \rvert dB$	13	10	7	6	5	3	0
φ_m (deg.)	65	55	42	38	33	19	0

USING MATLAB TO OBTAIN TRANSFER FUNCTIONS IN THE W-DOMAIN

Reminder

$$z = \frac{1+\frac{T}{2}w}{1-\frac{T}{2}w} \qquad \text{and} \qquad w = \frac{2}{T}\frac{z-1}{z+1}$$

Let us take an example to illustrate the following ideas:

Let : $G(s) = \dfrac{2}{s(s+1)}$

Let *G(z)* be the z-Transform of *G(s)* as follows:

$$G(z) = \frac{0.0374z + 0.03504}{z^2 - 1.8187z + 0.8187} \qquad (1) \qquad \text{for} \qquad G(s) = \frac{2}{s^2+s}$$

$T = 0.2\,\text{sec}.$

We have: $z = \dfrac{1+0.1w}{1-0.1w}$ (2)

The command **"bilinear(num,den,0.5)"** transforms any transfer function in "z" into a transfer function in *x*, such that:

$$z = \frac{x-1}{x+1} \qquad (3)$$

Using $\underline{x = -0.1w}$, let us express "z" in terms of "w".

Hence, Equation (3) becomes:

$$z = \frac{-0.1w-1}{-0.1w+1} = -\frac{0.1w+1}{1-0.1w}$$

or,

$$-z = \frac{1+0.1w}{1-0.1w} \qquad (4)$$

Hence, these the steps to follow in order to transform (1) into an expression in *"w"*.

Professor M Mahfouf
Dept. of ACSE, The University of Sheffield

Step 1

Replace z by $-z$ in Equation (1), i.e. $G(-z)=\dfrac{-0.0374z+0.0354}{z^2+1.8187z+0.8187}$ (5)

>> num=[0 –0.0374 0.0354];
>> den=[1 1.8187 0.8187];

Step 2

Use the MATLAB command "bilinear" to obtain the transfer function (5) but in x, i.e.:

[numx,denx]=bilinear(num,den,0.5);

$$G(x)=\frac{-0.00055x^2+0.01946x+0.020014}{x^2-0.0997x} \qquad (6)$$

Step 3

Let us convert (6) from x to w ($x=-0.1w$):

For the numerator: $-0.00055(-0.1w)^2+0.01946.(-0.1w)+0.020014$.

For the denominator: $(-0.1w)^2-0.0997(-0.1w)$

Hence, to obtain (6) in terms of "w", we perform the following operation:

>> numw = [numx].*[(-0.1)^2 –0.1 1];
>> denw = [denx].*[(-0.1)^2 -0.1 1];

If one wants to make the first coefficient of the denominator unity (1), then the above statementss can rewritten as follows:

>> numw = [numx].*[1 –10 100];
>> denw = [denx].*[1 –10 100];

All we did was to multiply [(-0.1)^2 –0.1 1] by 100.

This should give the following transfer function in *"w"*:

$$G(w)=\frac{-0.00055w^2-0.1946w+2.0014}{w^2+0.997w}$$

Note:

It is always recommended to work in double precision, i.e.
>> format long;

As far as the Laboratory Design Session is concerned, you will need to do the following:

- Find the sampling interval (see important note below).

- Transform $G(s)=\dfrac{684}{s(s^2+12s+57)}$ into $G(z)$ using the above sampling interval and using the MATLAB command *c2dm* with the Zero-Order Hold.

- Follow Steps 1-3 to transform $G(z)$ into $G(w)$.

- Because the analysis in $"w"$ in similar to the analysis in $"s"$ follow the same procedures you followed to design a continuous controller only this time you use $"w"$.

- Once you have obtained all your controllers revert back to $"z"$ by substituting $"w"$ with $"z"$ using $w=\dfrac{2}{T}\left[\dfrac{z-1}{z+1}\right]$.

- Simulate the closed-loop system in $"z"$.

<u>Important Note:</u>

The closed-loop bandwidth, given a settling time T_s and a damping ratio ζ, is given by:

$$\omega_B=\frac{4}{T_s\zeta}\sqrt{(1-2\zeta^2)+\sqrt{4\zeta^4-4\zeta^2+2}}$$

$T_s = settling-time$

$\omega_B = Bandwidth$

$\zeta = damping\ ratio$

Professor M Mahfouf
Dept. of ACSE, The University of Sheffield