Scilab Manual for Advanced Control System (Rev. 2015) by Prof Mrs. Deepti Khimani Instrumentation Engineering Vivekanand Education Society'S Institute Of Technology/ Mumbai University[1]

Solutions provided by
Prof Mrs. Deepti Khimani
Instrumentation Engineering
Vivekanand Education Society'S Institute Of Technology/ Mumbai University

November 5, 2026

[1]Funded by a grant from the National Mission on Education through ICT, http://spoken-tutorial.org/NMEICT-Intro. This Scilab Manual and Scilab codes written in it can be downloaded from the "Migrated Labs" section at the website http://scilab.in

Contents

List of Scilab Solutions 3

1 Construct phase trajectory of the system using Delta method and verify the same using Scilab 5

2 Check the existance and stability of limit cycles in Wien Bridge Oscillator. 7

3 Plot amplitude versus describing function of saturation nonlinearity. 9

4 Plot amplitude and phase versus describing function of realy with hysterisis nonlinearity. 14

5 Investigate stability of a system with relay control using describing function. 19

6 Design optimal control for a system to minimize performance index. 22

7 Design IMC controller for the non minimum phase delay system. 25

8 Computation of PID Controller gains based on IMC design for the first order delay system 29

List of Experiments

Solution 3.1	Lab	9
Solution 4.1	Lab	14
Solution 5.1	Lab	19
Solution 6.1	Lab	22
Solution 7.1	Lab	25
Solution 8.1	Lab	29
AP 1	figure properties	32

List of Figures

1.1 Lab 6

2.1 Lab 8

3.1 Lab 12
3.2 Lab 13

4.1 Lab 17
4.2 Lab 18

5.1 Lab 21

6.1 Lab 24

7.1 Lab 27
7.2 Lab 28

8.1 Lab 31

Experiment: 1

Construct phase trajectory of the system using Delta method and verify the same using Scilab

This code can be downloaded from the website wwww.scilab.in

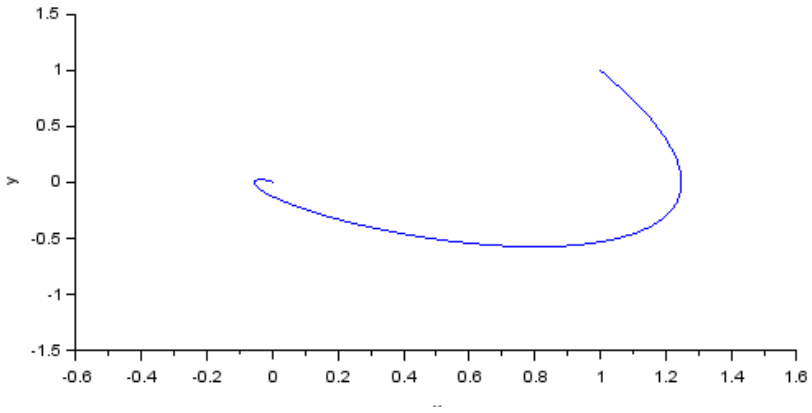


Figure 1.1: Lab

Experiment: 2

Check the existance and stability of limit cycles in Wien Bridge Oscillator.

This code can be downloaded from the website wwww.scilab.in

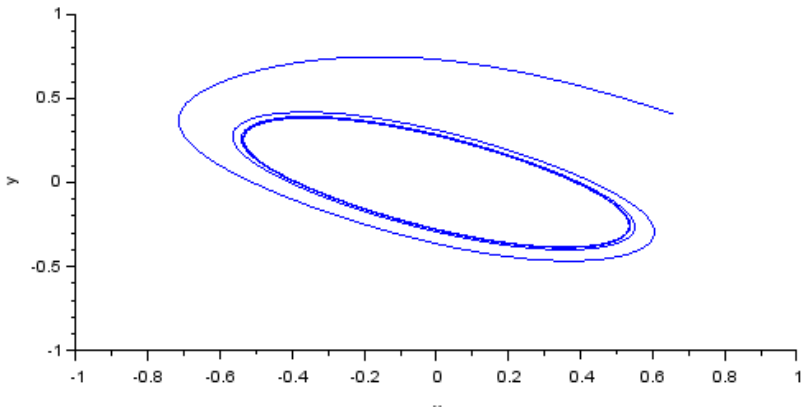


Figure 2.1: Lab

Experiment: 3

Plot amplitude versus describing function of saturation nonlinearity.

Scilab code Solution 3.1 Lab

```
1 //Lab 03:Plot amplitude versus describing function
     of saturation nonlinearity.
2
3 //scilab - 6.0.0
4 // Operating System : Windows 10, 64-bit
5
6 //
  ------------------------------------------------------------------------

7 close;
8 clc;
9 clear;
10 //
  ------------------------------------------------------------------------

11
```

```
12 //Response of the saturation nonlinearity to
      sinusoidal input
13 figure;
14 importXcosDiagram(".\Lab03_DF_Saturation_Model.xcos"
      )
15 xcos_simulate(scs_m,4);
16 scs_m.props.context
17 plot(yt.time,yt.values(:,1),'r--','thickness',2)
18 plot(yt.time,yt.values(:,2),'b','thickness',2)
19 xlabel('Time (sec.)','fontsize',2);
20 ylabel('Amplitude','fontsize',2);
21 xset('font size',2);
22 title("Saturation nonlinearity output to sinusoidal
      input",'fontsize',4);
23 exec .\figure_properties.sci; //custom script for
      setting figure properties
24 //
   ------------------------------------------------------------------------

25 //Describing Functin for saturation nonlinearity.
26 k=1;
27 N=1;
28 i=1;
29 Keq=[];
30
31 for a=0:0.2:10
32     if k*a/N > 1 then
33     Keq(i,1)=2/%pi*(k*asin(N/a/k)+N/a*sqrt(1-(N/k/a)
          ^2))
34     else
35     Keq(i,1)=k
36     end
37     i=i+1;
38 end
39
40 a=0:0.2:10;
41 a=a';
42 figure,
```

```
43 plot(a,Keq,'thickness',2)
44 xlabel('$a$','fontsize',2);
45 ylabel('$K_{eq}$','fontsize',2);
46
47 xset('font size',2);
48 title("Describing Function for a saturation
      nonlinearity with k=N=1",'fontsize',4);
49 exec .\figure_properties.sci; //custom script for
      setting figure properties
50 zoom_rect([0 0 10 1.1])
51 //
```

This code can be downloaded from the website wwww.scilab.in

check Appendix AP 1 for dependency:

`figure_properties.sci`

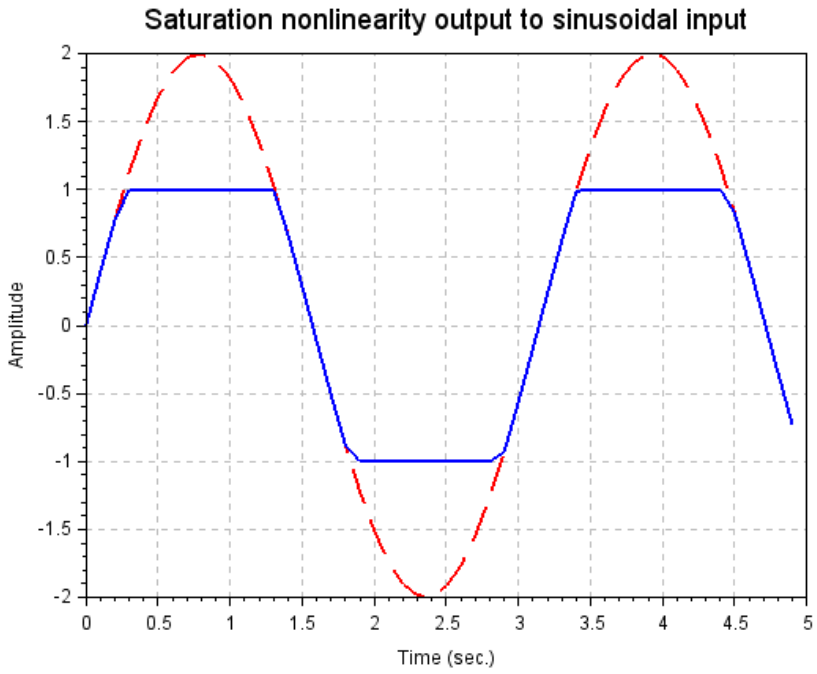


Figure 3.1: Lab

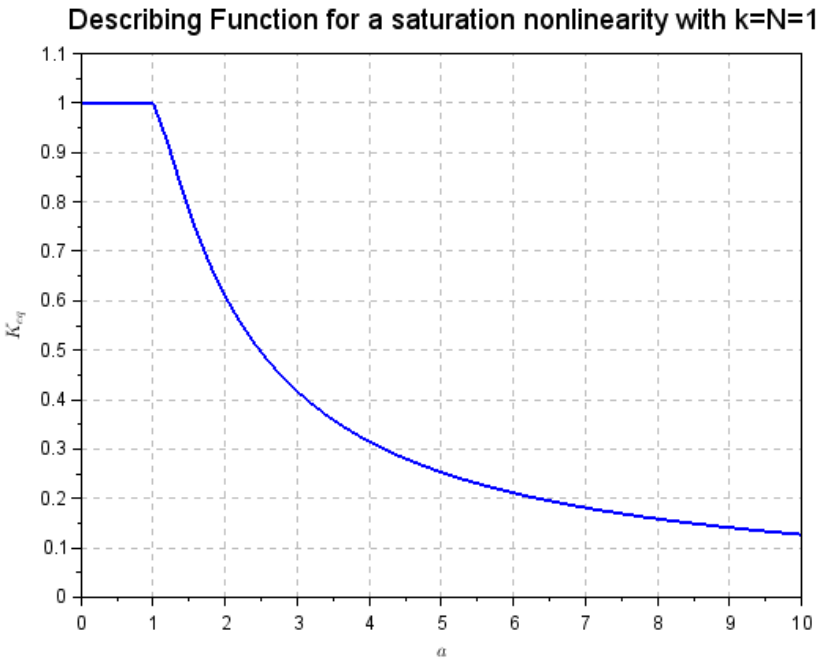


Figure 3.2: Lab

Experiment: 4

Plot amplitude and phase versus describing function of realy with hysterisis nonlinearity.

Scilab code Solution 4.1 Lab

```
1 //Lab 04. Plot amplitude and phase versus describing
      function of realy with hysterisis nonlinearity
2
3 //scilab - 6.0.0
4 // Operating System : Windows 10, 64-bit
5
6 //
  ------------------------------------------------------------------------

7 close;
8 clc;
9 clear;
10 //
  ------------------------------------------------------------------------
```

```
11 //Response of the Relay with hysteresis nonlinearity
       to sinusoidal input
12 figure;
13 importXcosDiagram(".\Lab04_DF_RelayNHyst_Model.xcos"
     )
14 xcos_simulate(scs_m,4);
15 scs_m.props.context
16 plot(yt.time,yt.values(:,1),'r--','thickness',2)
17 plot(yt.time,yt.values(:,2),'b','thickness',2)
18
19 xlabel('Time (sec.)','fontsize',2);
20 ylabel('Amplitude','fontsize',2);
21 title("Relay with hysteresis nonlinearity output to
     sinusoidal input",'fontsize',4);
22 exec .\figure_properties.sci; //custom script for
     setting figure properties
23 zoom_rect([0 -1.2 5 1.2])
24 //
     ---------------------------------------------------------------------------

25 ////Describing Functin for relay with hysteresis
     nonlinearity.
26 h=0.1;
27 N=1;
28 i=1;
29
30 for a=0.1:0.025:1
31     if a<h then
32         Keq(i,1)=0;
33         ro(i,1)=0;
34         theta(i,1)=0
35     else
36         Keq(i,1)=4*N/(%pi*a)*(sqrt(1-(h/a)^2)-%i*h/a
               )
37         [r th]=polar(Keq(i,1));
38         ro(i,1)=r; //magnitude
39         theta(i,1)=clean(th); //angle in radians
40     end
```

```
41      i=i+1;
42  end
43
44  a=0.1:0.025:1
45  a=a';
46  figure,
47
48  subplot(2,1,1), plot(a,ro,'thickness',2)
49  xlabel('$a$','fontsize',2);
50  ylabel(['Magnitude', '$|K_{eq}|$'],'fontsize',2);
51
52  xset('font size',2);
53  exec .\figure_properties.sci; //custom script for
       setting figure properties
54  title("Describing Functin for relay with hysteresis
       nonlinearity with h=0.1 and N=1",'fontsize',4);
55
56  subplot(2,1,2), plot(a,theta*180/%pi,'thickness',2)
57  xlabel('$a$');
58  ylabel(['Phase', '$ \angle K_{eq}$','deg.']);
59  xset('font size',2);
60  exec .\figure_properties.sci; //custom script for
       setting figure properties
61  //
```

This code can be downloaded from the website wwww.scilab.in

check Appendix AP 1 for dependency:

`figure_properties.sci`

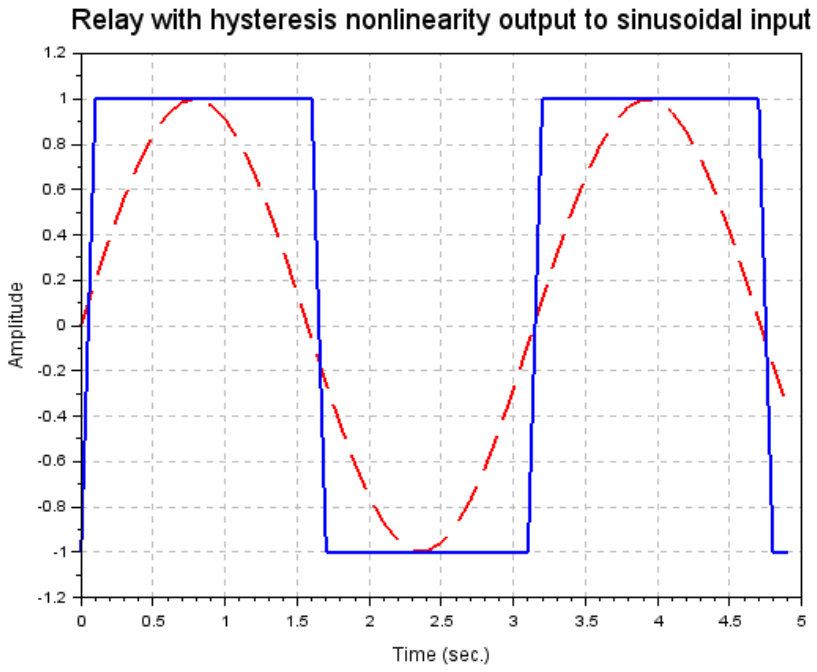


Figure 4.1: Lab

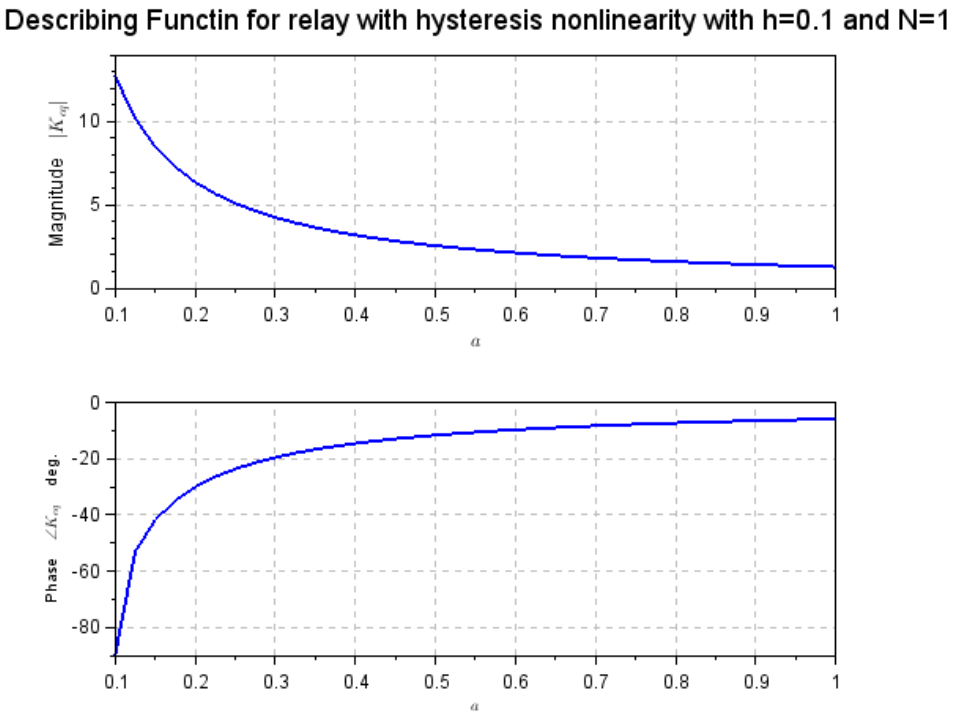


Figure 4.2: Lab

Experiment: 5

Investigate stability of a system with relay control using describing function.

Scilab code Solution 5.1 Lab

```
1 //Lab 05:Investigate stability of a system with
     relay control using describing function.
2 //scilab - 6.0.0
3 // Operating System : Windows 10, 64-bit
4
5 //
     ------------------------------------------------------------------------

6 close;
7 clc;
8 clear;
9 //
     ------------------------------------------------------------------------

10
11 // System Decription
12 s=poly(0,'s');
```

```
num=5;
den=(s+1)*(0.1*s+1)^2;
g=syslin('c',num/den)

//Nyquist plot of the system
nyquist(g)
title("Nyquist plot and describing function to
   determine limit cycle",'fontsize',4)
f=gca();
f.x_location = "origin"
f.y_location = "origin"
h=legend('');
h.visible = "off"
xset("color",2);

// Nyquist Plot of Describing Function for
   saturation nonlinearity.
omegat=0.05:0.05:%pi;
a=sin(omegat);
k=1;
Keq=4/%pi./a
DF_nyq=-1 ./Keq;

plot(DF_nyq,zeros(1,length(DF_nyq)),'m-.','thickness
   ',2)
exec .\figure_properties.sci; //custom script for
   setting figure properties
//zoom_rect([-0.8 -1.4 0.4 1.4])

//limit cycle point (Identified on plot)
plot(-0.2064, 0,'bo');

// Frequency and Magnitude of limit cycle
p=roots(den);
p=abs(p);
omega=sqrt(sum(1./p)/prod(1./p)) //rad/sec
set(gca(),"grid",[1 1])
xset('font size',2)
```

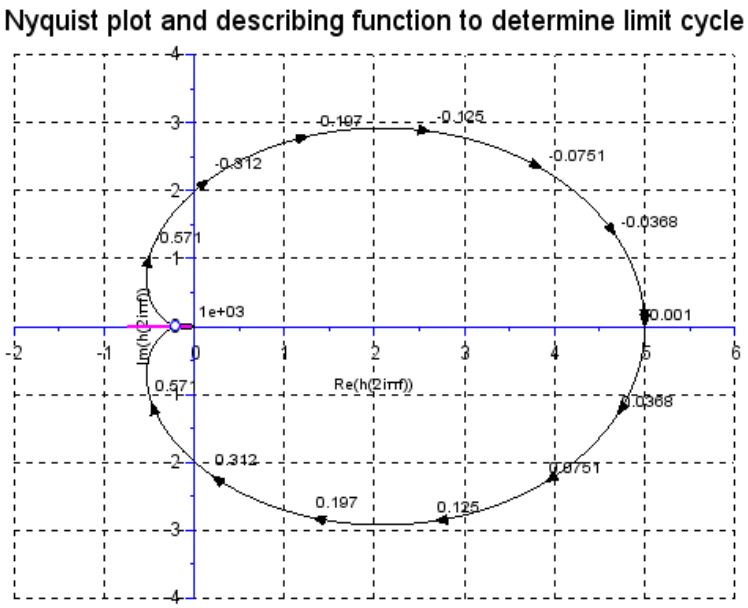


Figure 5.1: Lab

```
47 magn=prod(0.2064-roots(den))
```

---

check Appendix AP 1 for dependency:

`figure_properties.sci`

Experiment: 6

Design optimal control for a system to minimize performance index.

Scilab code Solution 6.1 Lab

```
1 //Lab 06: Design optimal control for a system to
     minimize quadratic
2 //performance index (P.I.)
3
4 //scilab - 6.0.0
5 // Operating System : Windows 10, 64-bit
6
7 //
  ------------------------------------------------------------------------

8 close;
9 clc;
10 clear;
11 //
  ------------------------------------------------------------------------

12 // System Decription
```

```
13 a=[0 1;1 -3];
14 b=[0 1]';
15 c=[1 0];
16 d=0;
17 sys=syslin('c',a,b,c,d)
18
19 // coefficients of P.I.
20 Q=[1 0;0 1];
21 R=1;
22
23 // Solution P of the riccati equation and feedback
      gain that minimizes P.I.
24 [K,P]=lqr(sys,Q,R);
25
26 //Closed loop system with optimal control
27
28 sysc=syslin('c',a+b*K,b,c,d)
29
30 // Close loop solution
31 t=0:0.1:10;
32 [y x]=csim('step',t,sysc);
33
34 //Closed loop response
35 plot(t,x,'thickness',2)
36 title('Evolution of states using optimal control','
      fontsize',4)
37 xlabel('Time t','fontsize',2)
38 ylabel('Response x(t)','fontsize',2)
39 set(gca(),"grid",[1 1])
40 h=legend('x1','x2');
41 h.font_size = 2;
42 xset('font size',2)
```

check Appendix AP 1 for dependency:

`figure_properties.sci`

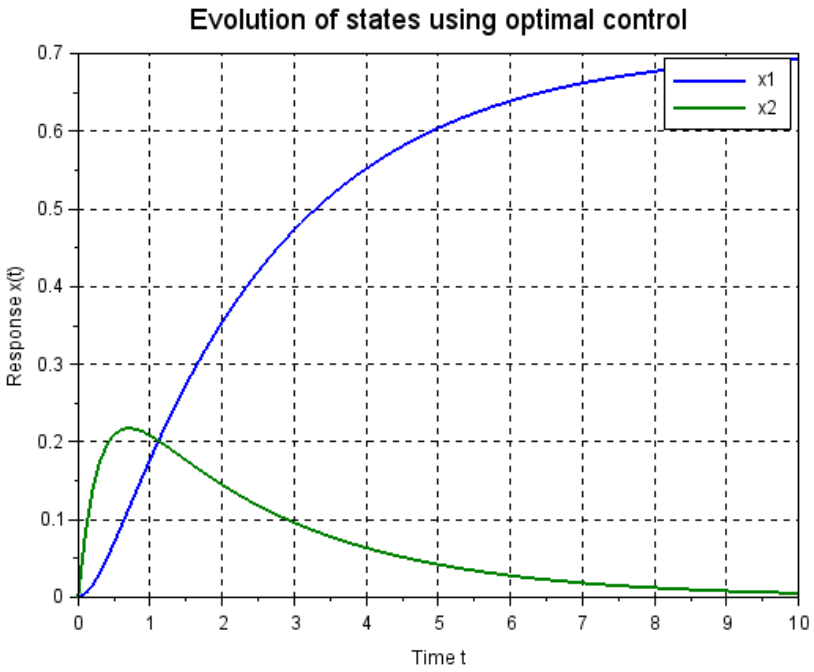


Figure 6.1: Lab

Experiment: 7

Design IMC controller for the non minimum phase delay system.

Scilab code Solution 7.1 Lab

```
1 //Lab 07: Design IMC controller for the Non-minimum
     Phase Delay system.
2
3 //scilab - 6.0.0
4 // Operating System : Windows 10, 64-bit
5
6 //
     ------------------------------------------------------------------------

7 close;
8 clc;
9 clear;
10 //
     ------------------------------------------------------------------------

11
12 //System transfer function without delay (delay
```

```
    introduced in zcos)
13 s=poly(0,'s');
14 num=0.5848*(-0.32*s+1);
15 den=0.18*s^2+0.87*s+1;
16
17 //model
18 num_m=0.5848*(-0.3549*s+1);
19 den_m=0.1858*s^2+0.8627*s+1;
20
21 // Response of the system with IMC Control
22 figure;
23 importXcosDiagram(".\Lab07_IMC_Model.xcos")
24 xcos_simulate(scs_m,4);
25 scs_m.props.context
26 plot(ye.time,ye.values(:,1),'thickness',2)
27 xlabel('t (sec.)','fontsize',2);
28 ylabel('y(t)','fontsize',2);
29 title("Response of the system with IMC controller.",
    'fontsize',4);
30 set(gca(),"grid",[1 1])
31 xset('font size',2)
32
33 exec .\figure_properties.sci; //custom script for
    setting figure properties
34
35 // Error in plant and model output.
36 figure;
37 plot(ye.time,ye.values(:,2),'thickness',2)
38 xlabel('t (sec.)','fontsize',2);
39 ylabel('e(t)','fontsize',2);
40 set(gca(),"grid",[1 1])
41 xset('font size',2)
42
43 title("Error in plant and model output.",'fontsize'
    ,4);
44 exec .\figure_properties.sci; //custom script for
    setting figure properties
```

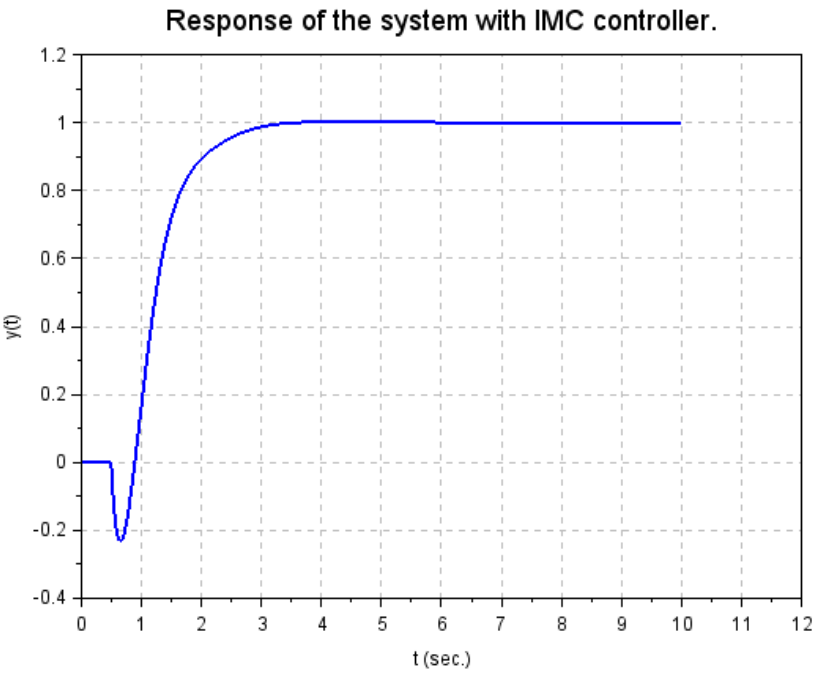


Figure 7.1: Lab

This code can be downloaded from the website wwww.scilab.in

check Appendix AP 1 for dependency:

`figure_properties.sci`

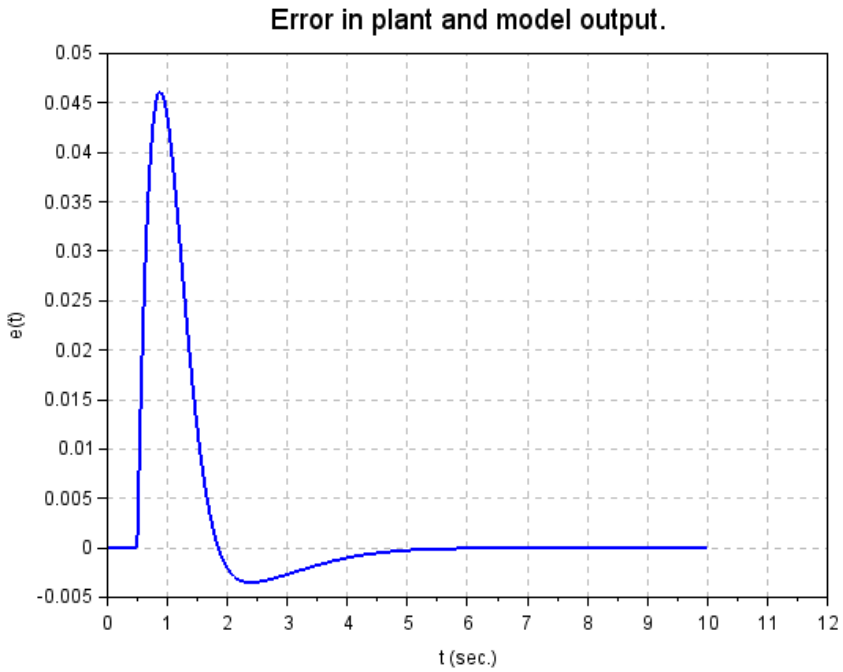


Figure 7.2: Lab

Experiment: 8

Computation of PID Controller gains based on IMC design for the first order delay system

check Appendix AP 1 for dependency:

```
figure_properties.sci
```

Scilab code Solution 8.1 Lab

```
1 //Lab 08: Computation of PID Controller gains based
      on IMC Design for the First order Delay system.
2
3 //scilab - 6.0.0
4 // Operating System : Windows 10, 64-bit
5
6 //
  ------------------------------------------------------------------------

7 close;
8 clc;
9 clear;
```

```
10 //
   ------------------------------------------------------------------------

11
12 //System transfer function
13 kp=0.5;
14 tp=1.5;
15 theta=0.8; //for Pade approximation of the delay of
      0.8sec
16 s=poly(0,'s');
17 num=kp;
18 den=tp*s+1;
19
20 //IMC filter designed parameters
21 lambda=0.5;
22
23 //PID gain computation
24 kc=(tp+0.5*theta)/kp/(lambda+0.5*theta);
25 ti=tp+0.5*theta;
26 td=tp*theta/(2*tp+theta);
27
28 // Response of the system with IMC based PID control
29 figure;
30 importXcosDiagram(".\Lab08_IMC_PID_Model.xcos")
31 xcos_simulate(scs_m,4);
32 scs_m.props.context
33 plot(yt.time,yt.values,'thickness',2)
34 xlabel('t (sec.)','fontsize',2);
35 ylabel('y(t)','fontsize',2);
36 title("Response of the system with IMC based PID
      control.",'fontsize',4);
37 set(gca(),"grid",[1 1])
38 xset('font size',2)
39
40 exec .\\figure_properties.sci; //custom script for
      setting figure properties
```

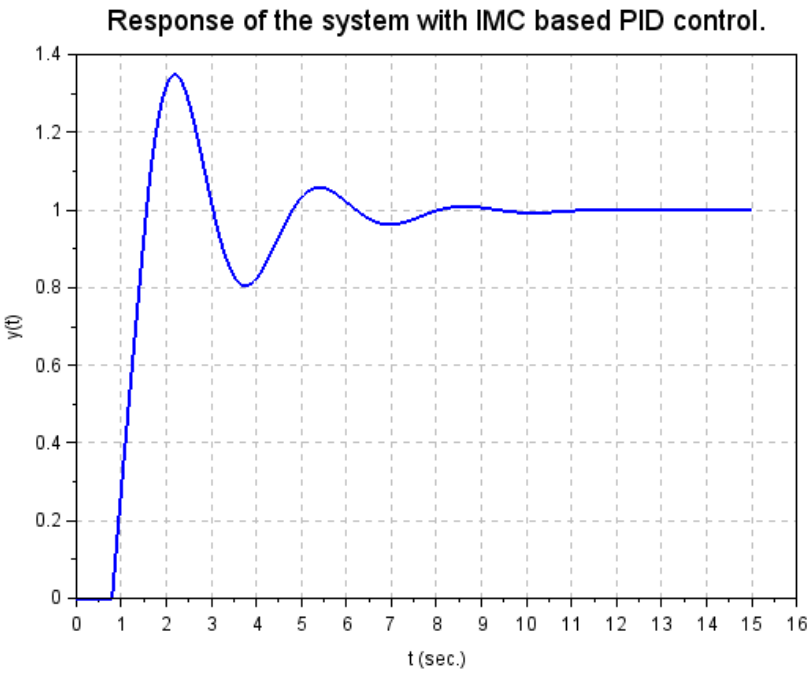


Figure 8.1: Lab

This code can be downloaded from the website wwww.scilab.in

Appendix

Scilab code AP 1 //

```
2 //figure handel settings
3 f=get("current_figure"); //Current figure handle
4 f.background=8; //make the figure window background
    white
5 l=f.children(1);
6 l.background=8 ;//make the text background white
7 id=color('grey');
8 xgrid(id);
9 //
```

figure properties