Scilab Manual for Feedback Control Lab Practices by Prof Deepti Khimani Instrumentation Engineering Vivekanand Education Society'S Institute Of Technology[1]

Solutions provided by
Prof Deepti Khimani
Instrumentation Engineering
Vivekanand Education Society'S Institute Of Technology

July 16, 2024

[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 Experiments

List of Figures

Experiment: 1

Observe the effect of change in time constant of the first-order systems.

Scilab code Solution 1.1 Effect of time constant

```
//Lab 1 : Observe the effect of change in time constant of the first-order systems.
// ____________________________________________________________
// Problem Statement: Compare step response of
// G1=1/(2s+1), G2=1/(s+1) and
// G3=1/(4s+1)
// ____________________________________________________________
// Operating System OSX (Mac) 10.14.6
// Scilab Version 6.1.0
// ____________________________________________________________

```

```
xdel(winsid()) // close the figure windows
clear; // clear the workspace
clc; // clear the console
//------------------------------------------------------------------------------
// system representation

s=%s  //Laplace operator

//system with various time constants.

G1=syslin('c',1/(2*s+1))     //system 1 represented by G1(s)=1/(2s+1)
G2=syslin('c',1/(s+1))       //system 2 represented by G2(s)=1/(s+1)
G3=syslin('c',1/(4*s+1))     //system 3 represented by G3(s)=1/(4s+1)

//------------------------------------------------------------------------------
// Unit step response
u=1
t=0:0.2:15                  // Simulation time
c1=csim('step',t,G1*u)  // Response of system G1(s)=1/(2s+1)
c2=csim('step',t,G2*u)  // Response of system G2(s)=1/(s+1)
c3=csim('step',t,G3*u)  // Response of system G3(s)=1/(4s+1)

//plots

plot(t,c1,t,c2,'r-',t,c3,'LineWidth',2)

// plot of response value at one time-constant
plot(2,0.6321,'o',1,0.6321,'ro',4,0.6321,'o','
```

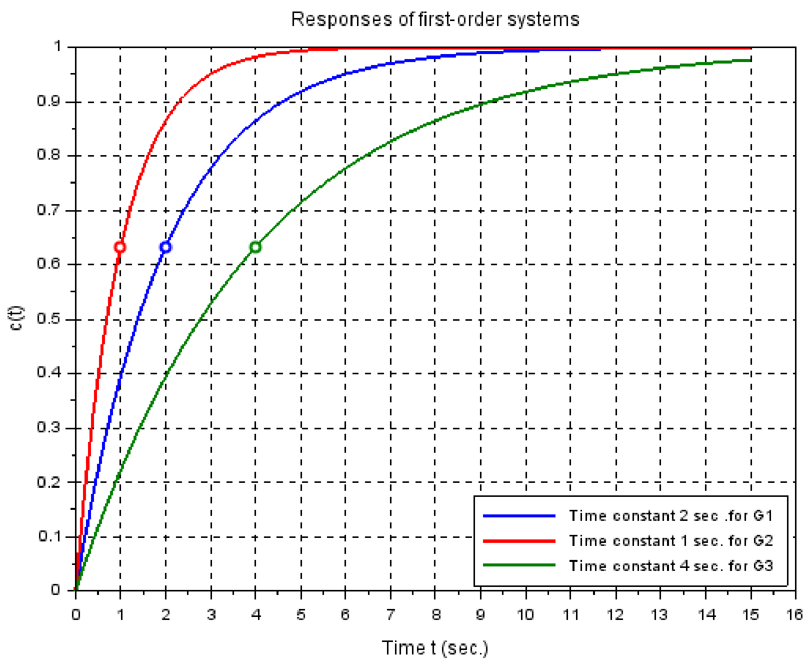

Figure 1.1: Effect of time constant

```
LineWidth',2)
plot(t,0.6321,'--','LineWidth',1.5)
xgrid

//Title, labels and grid to the figure
legend('Time constant 2 sec .for G1','Time constant 1 sec. for G2', 'Time constant 4 sec. for G3',4)
title('Responses of first-order systems' ,'fontsize',2)
xlabel('Time t (sec.)','fontsize',2)
ylabel('c(t)','fontsize',2)
```

Experiment: 2

Plot the time response of second-order system with real poles and compare it with responses of it's individual poles.

Scilab code Solution 2.1 Time response of SOS

```
//Lab 2 : Plot the time response of second−order system with real poles
// and compare it with responses of it's individual poles.
//
// Problem Statement: Plot the time response of
//G=1/(2s+1)/(3s+1)) and comapare it with responses of
// G1=1/(2s+1) and G2=1/(3s+1))
//
```

```

// ------------------------------------------------------------------------
// Operating System OSX (Mac) 10.14.6
// Scilab Version 6.1.0
// ------------------------------------------------------------------------

xdel(winsid()) // close the figure windows
clear; // clear the workspace
clc; // clear the console
// ------------------------------------------------------------------------
// system representation

s=%s  //Laplace operator

//system with various time constants.

G=syslin('c',1/(2*s+1)/(3*s+1))    // representation of original system G(s)=1/(2s+1)/(3s+1)

G1=syslin('c',1/(2*s+1))            // G1(s)=1/(2s+1)
G2=syslin('c',1/(3*s+1))            // G2(s)=1/(3s+1)

// ------------------------------------------------------------------------
// Unit step response

t=0:0.2:30              // Simulation time
dim=size(t);
u=ones(dim(1),dim(2)) // unit step input

```

```
c=csim(u,t,G)  // Response of system G

c1=csim(u,t,G1)  // Response of system G1
c2=csim(u,t,G2)  // Response of system G2

//plot
plot(t,c,t,c1,t,c2,'LineWidth',2)

xgrid

//Title, labels and grid to the figure
legend('System G (SO)','System G1(FO)', 'System G2(FO)',4)
title('Responses of second-order system and its first-order poles' ,'fontsize',2)
xlabel('Time t (sec.)','fontsize',2)
ylabel('c(t)','fontsize',2)
```

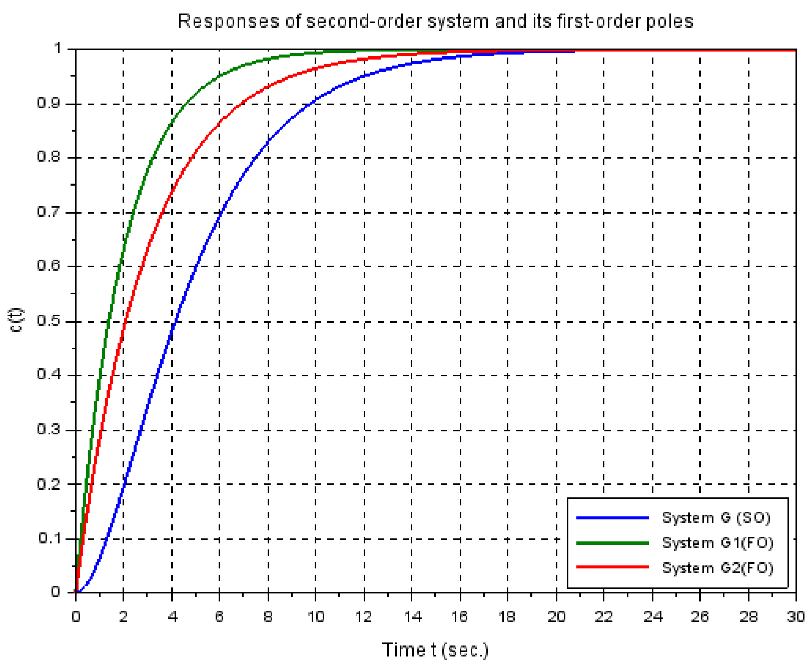

Figure 2.1: Time response of SOS

Experiment: 3

Plot the time-responses of the second-order system with various damping factors.

Scilab code Solution 3.1 Time response of SOS

```
//Lab 3 : Plot the time-responses of the second-order systems with
//various damping factors.
//
// ------------------------------------------------------------
// Problem Statement: Compare Step response of G=1/(s^2+2*xi*wn*s+wn^2)
//with natural frequency wn=1 rad/sec and xi=0.3 (underdamped),
//xi=1.5 (overdamped) and xi=0 (undamped).
//
// ------------------------------------------------------------

//
// ------------------------------------------------------------
```

```
// Operating System OSX (Mac) 10.14.6
// Scilab Version 6.1.0
// ------------------------------------------------------------------
xdel(winsid()) // close the figure windows
clear; // clear the workspace
clc; // clear the console
// ------------------------------------------------------------------
// system representation
s=%s // Laplace operator
s=poly(0,'s');
wn=1 // Natural frequency wn=1 rad/sec.

// case 1 : underdamped system with xi=0.3
xi=0.3
G1=syslin('c',1/(s^2+2*xi*wn*s+wn^2))   //G1(s)=1/(s^2+0.3s+1)

// case 2 : overdamped system with xi=1.5
xi=1.5
G2=syslin('c',1/(s^2+2*xi*wn*s+wn^2))   //G2(s)=1/(s^2+1.5s+1)

// case 3 : undamped system with xi=0
xi=0
G3=syslin('c',1/(s^2+2*xi*wn*s+wn^2))   //G3(s)=1/(s^2+1)
// ------------------------------------------------------------------
// Unit step response
u=1
t=0:0.1:20             // Simulation time
c1=csim('step',t,G1*u)  // Response of system G1
```

```
c2=csim('step',t,G2*u)  // Response of system G2
c3=csim('step',t,G3*u)  // Response of system G3

//plots

plot(t,c1,t,c2,'r-',t,c3,'LineWidth',2) //Responses
plot(t,u,'--','LineWidth',1) // input signal

xgrid

//Title, labels and grid to the figure
legend('Underdamped System','overdamped System', 'Undamped System',4)
title('Responses of second-order systems' ,'fontsize',3)
xlabel('Time t (sec.)','fontsize',2)
ylabel('c(t)','fontsize',2)
```

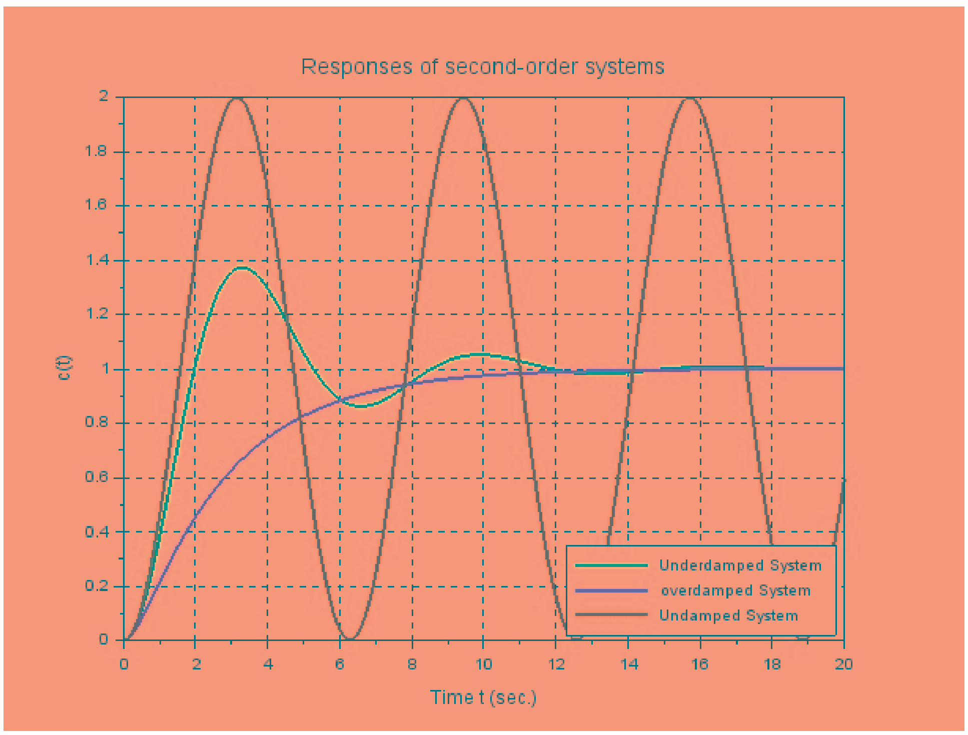

Figure 3.1: Time response of SOS

Experiment: 4

Examine the steady state errors for Type 0 system.

Scilab code Solution 4.1 Steady state error analysis

```
//Lab 4 :  Examine the steady state errors for Type
    0 system.
//
    ________________________________________________________________

// (a) Problem Statement: Examine the steady state
    errors in step,
// ramp and parabolic response of Type 0 system G
    =0.5/(s+0.8).
//
    ________________________________________________________________


//
    ________________________________________________________________

// Operating System OSX (Mac) 10.14.6
// Scilab Version 6.1.0
//
```

```
-----------------------------------------------------------------------
xdel(winsid()) // close the figure windows
clear; // clear the workspace
clc; // clear the console
//-----------------------------------------------------------------------
// system representation

s=%s  //Laplace operator

//Type 0 system G=0.5/(s+0.8)

G=syslin('c',0.5/(s+0.8))

Gcl=G/(1+G) //closed loop Gcl=0.5/(s+1.3)

//-----------------------------------------------------------


t=0:0.2:12              // Simulation time
dim=size(t);

u1=ones(dim(1),dim(2)) // unit step input
u2=t                    // unit ramp;
u3=t^2/2                // unit parabolic input

// Unit step response

c1=csim(u1,t,Gcl)  // unit step response of closed loop system
c2=csim(u2,t,Gcl)  // unit rmp response of closed loop system
c3=csim(u3,t,Gcl)  // unit parabolic response closed loop system

```

```
plot(t,c1,t,c2,t,c3,t,'LineWidth',2)
plot(t,u1,'--',t,u2,'--',t,u3,'--','LineWidth',1.5)

zoom_rect([0,0,8,1.5]) //zoom the grapics window for clarity

xgrid

//Title, labels and grid to the figure
legend('unit step response','unit ramp response', 'unit parabolic response',4)
title('Responses of Type 0 system (dashed lines are inputs)' ,'fontsize',2)
xlabel('Time t (sec.)','fontsize',2)
ylabel('c(t)','fontsize',2)
```

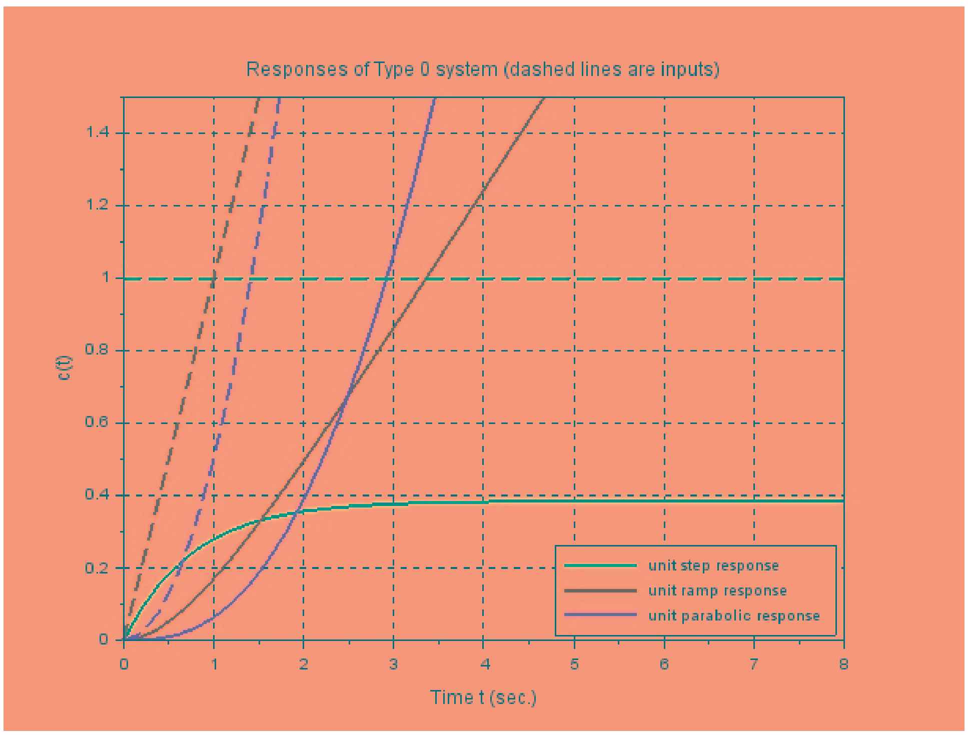

Figure 4.1: Steady state error analysis

Experiment: 5

Examine the steady state errors for Type 1 system.

Scilab code Solution 5.1 Steady state error analysis

```
//Lab 5 :  Examine the steady state errors for Type 1 system.
//________________________________________________________________________

//Problem Statement: Examine the steady state errors in step,
// ramp and parabolic response of Type 1 system G=0.5/s/(s+0.8)
//________________________________________________________________________

//________________________________________________________________________

// Operating System OSX (Mac) 10.14.6
// Scilab Version 6.1.0
//________________________________________________________________________
```

```
xdel(winsid()) // close the figure windows
clear; // clear the workspace
clc; // clear the console
//---------------------------------------------------------------------------------------
// system representation

s=%s  //Laplace operator

//Type 1 system G=0.5/s/(s+0.8)

G=syslin('c',0.5/s/(s+0.8))
Gcl=G/(1+G) //closed loop Gcl=0.5/(s^2+0.8s+;0.5)

//------------------------------------------------------------------------

t=0:0.2:12                  // Simulation time
dim=size(t);

u1=ones(dim(1),dim(2)) // unit step input
u2=t                   // unit ramp;
u3=t^2/2               // unit parabolic input

// Unit step response

c1=csim(u1,t,Gcl)  // unit step response of closed loop system
c2=csim(u2,t,Gcl)  // unit rmp response of closed loop system
c3=csim(u3,t,Gcl)  // unit parabolic response closed loop system

plot(t,c1,t,c2,t,c3,t,'LineWidth',2)
plot(t,u1,'--',t,u2,'--',t,u3,'--','LineWidth',1.5)
```

```
zoom_rect([0,0,8,20]) //zoom the grapics window for
    clarity

xgrid

//Title, labels and grid to the figure
legend('unit step response','unit ramp response', '
    unit parabolic response',2)
title('Responses of Type 1 system (dashed lines are
    inputs)' ,'fontsize',2)
xlabel('Time t (sec.)','fontsize',2)
ylabel('c(t)','fontsize',2)
```

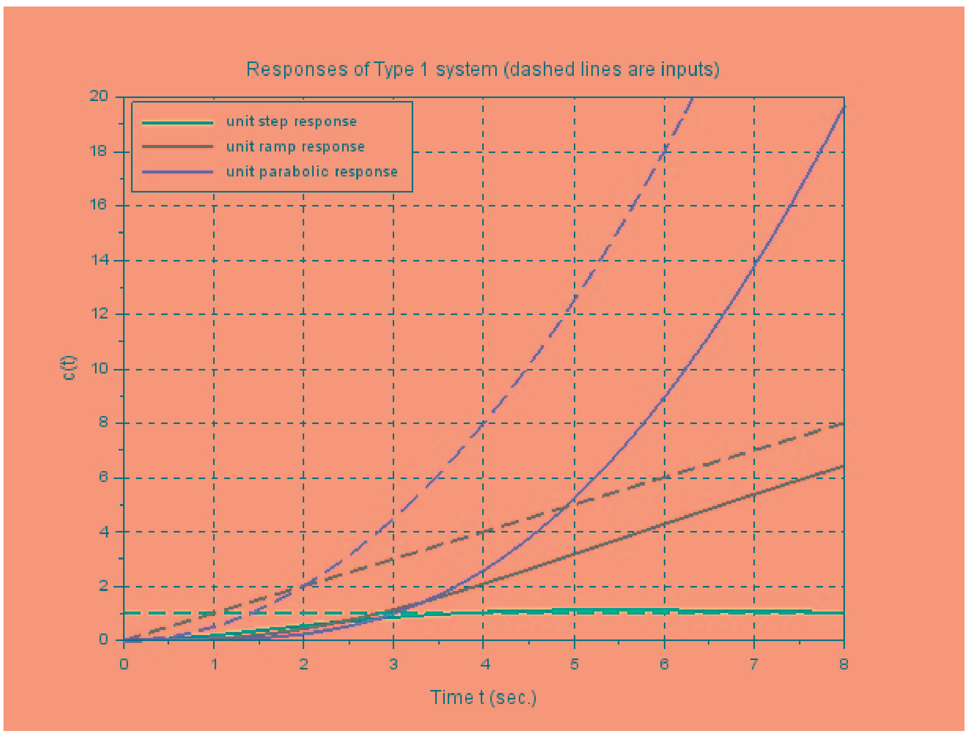

Figure 5.1: Steady state error analysis

Experiment: 6

Examine the steady state errors for Type 2 system.

Scilab code Solution 6.1 Steady state error analysis

```
//Lab 6 :  Examine the steady state errors for Type
   2 system.
//
   ____________________________________________________________________

//Problem Statement: Examine the steady state errors
    in step,
//ramp and parabolic response of Type 2 system G=(s
   +0.5)/s^2/(s+0.8)
//
   ____________________________________________________________________


//
   ____________________________________________________________________

// Operating System OSX (Mac) 10.14.6
// Scilab Version 6.1.0
//
```

```
----------------------------------------------------------------------
xdel(winsid()) // close the figure windows
clear; // clear the workspace
clc; // clear the console
//----------------------------------------------------------------------
// system representation

s=%s  //Laplace operator

//Type 1 system G=(s+0.5)/s^2/(s+0.8)

G=syslin('c',(s+0.5)/s^2/(s+0.8))
Gcl=G/(1+G) //closed loop Gcl=(s+0.5)/(s^3+0.8s^2+s+0.5)

//-------------------------------------------------------

t=0:0.2:10                // Simulation time
dim=size(t);

u1=ones(dim(1),dim(2)) // unit step input
u2=t                      // unit ramp;
u3=t^2/2                  // unit parabolic input

// Unit step response

c1=csim(u1,t,Gcl)  // unit step response of closed loop system
c2=csim(u2,t,Gcl)  // unit rmp response of closed loop system
c3=csim(u3,t,Gcl)  // unit parabolic response closed loop system

```

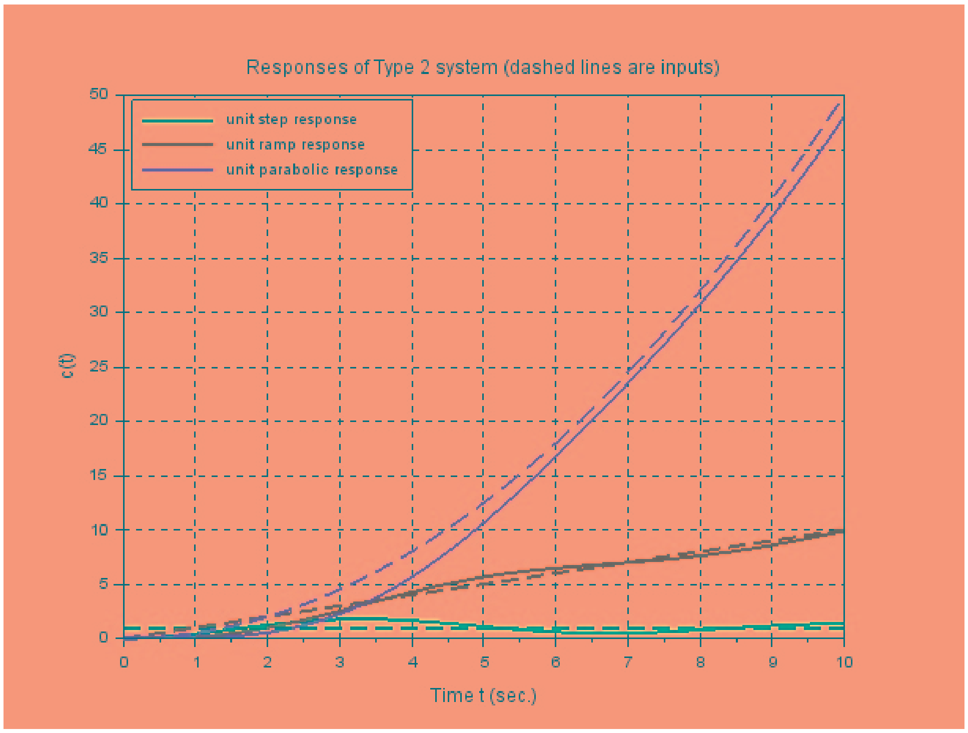

Figure 6.1: Steady state error analysis

```
plot(t,c1,t,c2,t,c3,t,'LineWidth',2)
plot(t,u1,'--',t,u2,'--',t,u3,'--','LineWidth',1.5)

xgrid

//Title, labels and grid to the figure
legend('unit step response','unit ramp response', 'unit parabolic response',2)
title('Responses of Type 2 system (dashed lines are inputs)' ,'fontsize',2)
xlabel('Time t (sec.)','fontsize',2)
ylabel('c(t)','fontsize',2)
```

Experiment: 7

Inspect the relative stability of systems by Root-Locus.

Scilab code Solution 7.1 Stability analysis

```
//Lab 7 : Inspect the relative stability of systems
   by Root-Locus.
//
   ________________________________________________________________________________

//Problem Statement: Draw the root locus of the
   system G=1/s/(s^2+1.5*s+1).
//Observe settling time of of the system with
   different gains.
//
   ________________________________________________________________________________


//
   ________________________________________________________________________________

// Operating System OSX (Mac) 10.14.6
// Scilab Version 6.1.0
//
```

```
------------------------------------------------------------
xdel(winsid()) // close the figure windows
clear; // clear the workspace
clc; // clear the console
//------------------------------------------------------------
// system representation
s=%s // Laplace operator
s=poly(0,'s');

G=syslin('c',1/(s*(s^2+1.5*s+1)))   // system representation G(s)=1/(s(s^2+1.5s+1))

//Compute marginal (critical) gain
//gain margin
gm= g_margin(G);
//gm=log10(kmar) ==> kmar=10^(gm/20)
kmar=10^(gm/20)

kmax=2 // maximum gain for ploting of root locus
evans(G,kmax)
xgrid
// for various k, system transfer functions are
k1=0.4
G1=k1*G/(1+k1*G)          //G1=0.4/(s^3+1.5s^2+s+0.4)
[z1,p1,k1]=tf2zp(G1)
plot(real(p1),imag(p1),'bx')

k2=0.8
G2=k2*G/(1+k2*G)         //G2=0.8/(s^3+1.5s^2+s+0.8)

[z2,p2,k2]=tf2zp(G2)
plot(real(p2),imag(p2),'gx')

k3=kmar                  //kmar=1.5
G3=k3*G/(1+k3*G)         //G1=0.4/(s^3+1.5s^2+s+1.5)
```

```

[z3,p3,k3]=tf2zp(G3)
plot(real(p3),imag(p3),'cx')
//------------------------------------------------------------------------
// Unit step response
t=0:0.1:40               // Simulation time
dim=size(t);

u=ones(dim(1),dim(2)) // unit step input

c1=csim(u,t,G1)   // Response of system G with k1
c2=csim(u,t,G2)   // Response of system G2 with k2
c3=csim(u,t,G3)   // Response of system G2 with k3

//plots
figure(1)
clf
plot(t,c1,t,c2,t,c3,'c:','LineWidth',2)
xgrid

//Title, labels and grid to the figure
legend('k=0.4','k=0.8', 'k=1.5',4)
title('Responses of system with different gains' ,'fontsize',3)
xlabel('Time t (sec.)','fontsize',2)
ylabel('c(t)','fontsize',2)
```

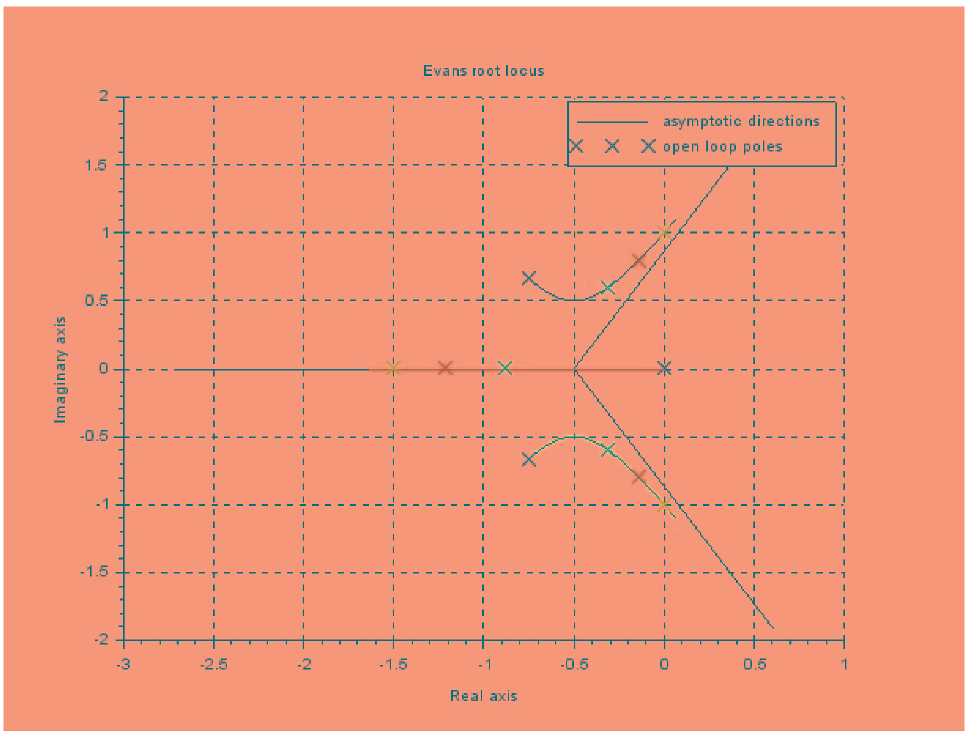

Figure 7.1: Stability analysis

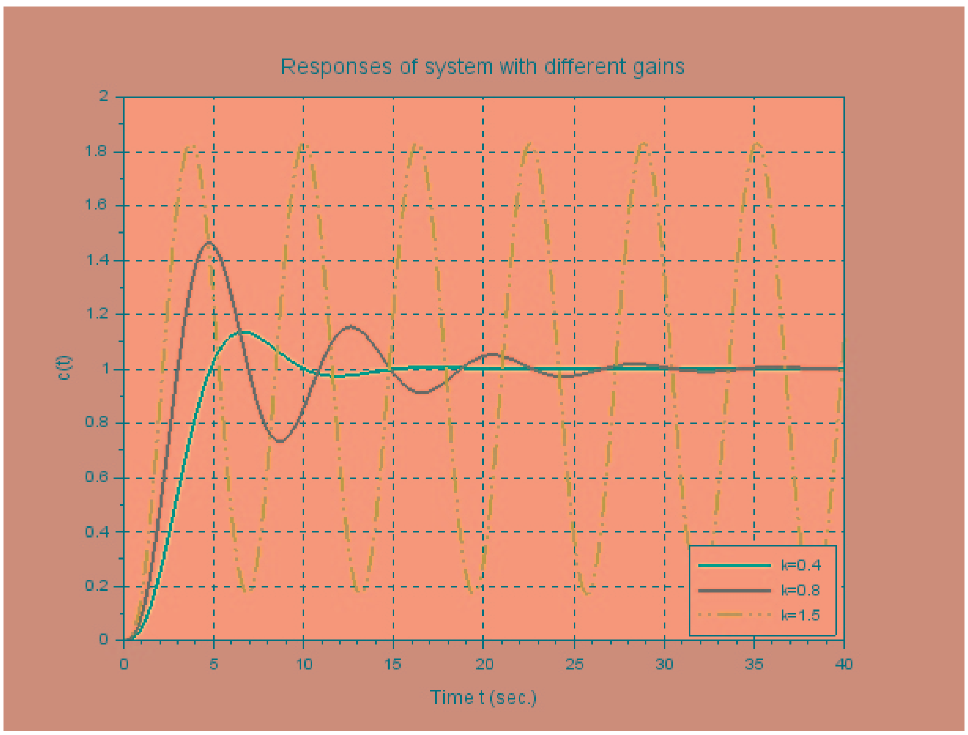

Figure 7.2: Stability analysis

Experiment: 8

Inspect the stability of systems by Bode plot.

Scilab code Solution 8.1 Stability analysis

```
//Lab 8 : Inspect the relative stability of systems
    from the Bode plot.
//
    ________________________________________________________________________

//Problem Statement: Draw the Bode of the system G
    =1/s/{s^2+1.8s+1}$.
//Determine the stability margins.
//
    ________________________________________________________________________


//
    ________________________________________________________________________

// Operating System OSX (Mac) 10.14.6
// Scilab Version 6.1.0
//
    ________________________________________________________________________
```

```
xdel(winsid()) // close the figure windows
clear; // clear the workspace
clc; // clear the console
//
    -----------------------------------------------------------------------------
// system representation
s=%s // Laplace operator
s=poly(0,'s');
G=syslin('c',1/s/(s^2+1.8*s+1))
//Bode plot with stability margins
show_margins(G,'bode')
[Gm,pcw] = g_margin(G)
[Pm,gcw] = p_margin(G)

disp("Gain Margin",Gm,"Phase crossover frequency",
    pcw)
disp("Phase Margin",Pm,"Gain crossover frequency",
    gcw)
```

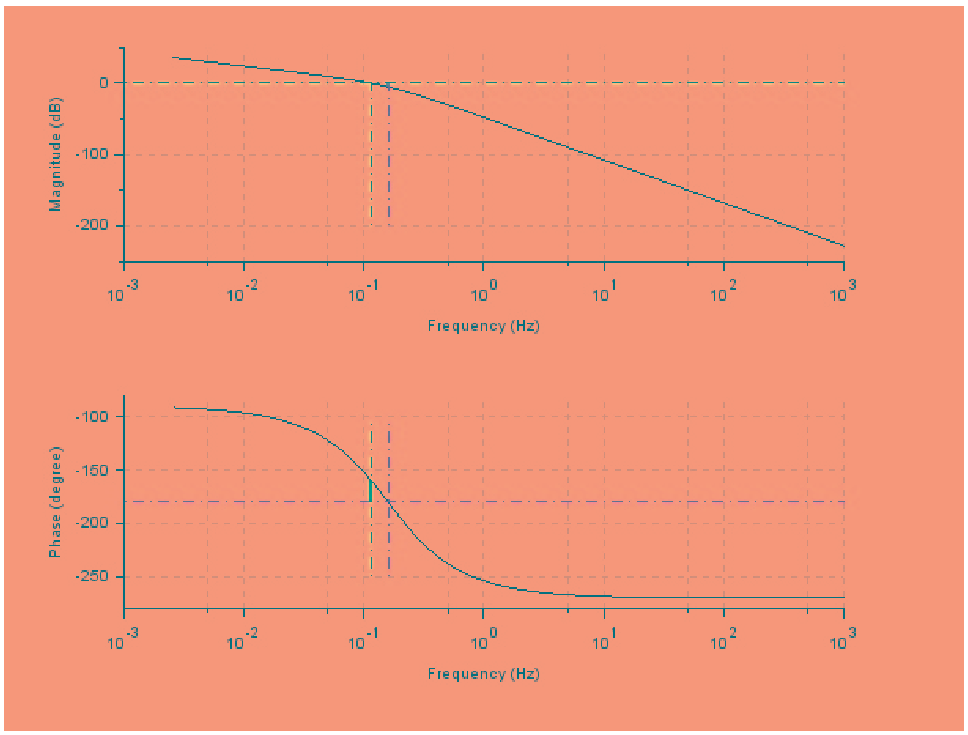

Figure 8.1: Stability analysis

Experiment: 9

Determine the frequency response specifications from Nyquist plot of system.

Scilab code Solution 9.1 Nyquist plot

```
//Lab 9 : Determine the frequency response
   specifications from Nyquist plot of system.
//
   ____________________________________________________________________
//Problem Statement: Draw the Nyquist plot of the
   system G=1/s/{s^2+5s+1}.
//Determine the stability margins.
//
   ____________________________________________________________________
//
   ____________________________________________________________________
// Operating System OSX (Mac) 10.14.6
// Scilab Version 6.1.0
//
```

```
  ----------------------------------------------------------------------
xdel(winsid()) // close the figure windows
clear; // clear the workspace
clc; // clear the console
//----------------------------------------------------------------------

// system representation
s=%s // Laplace operator
s=poly(0,'s');
G=syslin('c',1/s/(s^2+5*s+1))   //System representation G(s)=1/s/(s^2+5s+1)

//----------------------------------------------------------------------

//Nyquist plot with stability margins
show_margins(G,'nyquist')
zoom_rect([-1.5 -1.2 0 1.2])

[Gm,pcw] = g_margin(G)
[Pm,gcw] = p_margin(G)

disp("Gain Margin",Gm,"Phase crossover frequency",pcw)
disp("Phase Margin",Pm,"Gain crossover frequency",gcw)
```

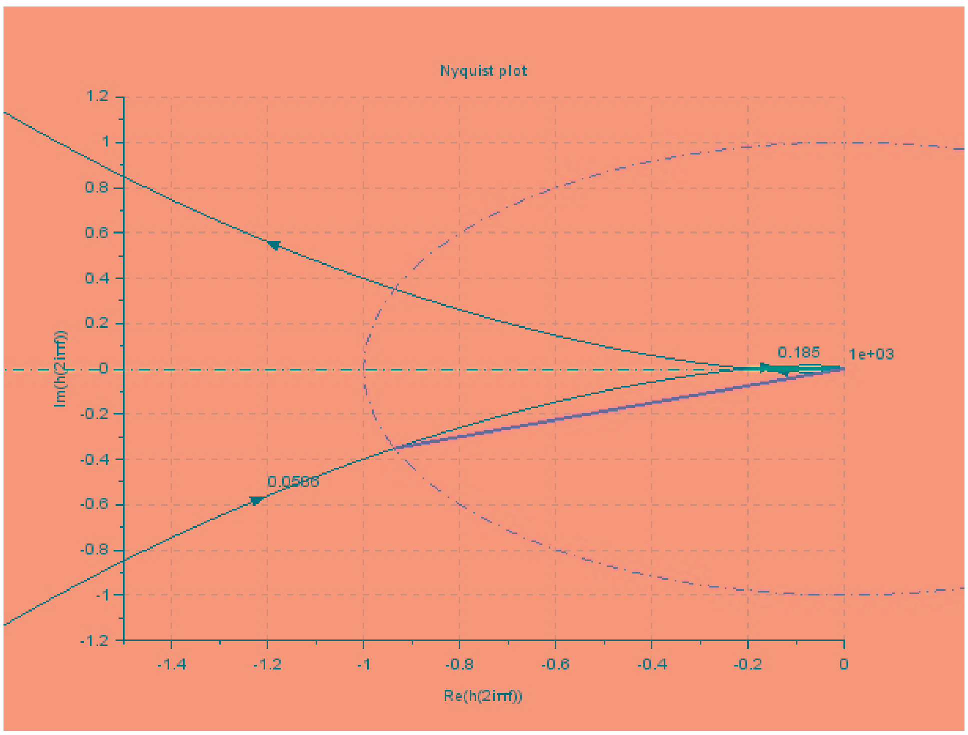

Figure 9.1: Nyquist plot