Scilab Manual for
Control Systems
by Mrs Supanna S Kumar
Electrical Engineering
KLE Dr M.S Sheshgiri College of Engg & Tech[1]

Solutions provided by
Mr. Supanna S Kumar
Electrical Engineering
KLE DR M.S Sheshgiri College of Engg&Tech, Belgaum

August 7, 2025

[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

Experiment: 1

Simulation of a typical second order system & determination of step response eval of time domain specifications

Scilab code Solution 1.1 Step Response and Time domain specifications

```
s=%s ;
T=syslin('c',25,25+4*s+s^2);
t=0:0.0005:5;
Ts=csim('step',t,T);
plot2d(t,Ts);
xgrid;
xtitle('Response of II order fn to unit-step input for T(s)=25/(s^2+6s+25)','Time(sec)',' C (t) ')
y=denom(T)                       // extracting the denominator of CL
z=coeff(y)      //extracting the coefficients of the denominator polynomial
                //Wn^2= z(1,1), comparing the coefficients
Wn=sqrt(z(1,1))          // Wn= natural frequency //
```

```
    2 * zeta * Wn = z(1,2)
zeta=z(1,2)/(2*Wn)                    // zeta = damping factor
Wd=Wn*sqrt(1-zeta^2)
Tp=%pi/Wd                                   // Tp= peak time
Mp=100*exp((-%pi*zeta)/sqrt(1-zeta^2))      // Mp=peak overshoot
Td=(1+0.7*zeta)/Wn                         // Td= delay time
a=atan(sqrt(1-zeta^2)/zeta)
Tr=(%pi-a)/Wd                              // Tr = rise time
Tset=4/(zeta*Wn)                             // Ts = settling time

Peak_time = sprintf("Peak Time = %6.3f secs",Tp);
Peak_overshoot = sprintf("Peak Overshoot = %6.3f
    percent",Mp);
Delay_time = sprintf("Delay_time = %6.3f secs",Td);
Rise_time = sprintf("Rise_time = %6.3f secs",Tr);
Settling_time = sprintf("Settling_time = %6.3f secs"
    ,Tset);

messagebox([Peak_overshoot,Peak_time,Delay_time,
    Rise_time,Settling_time],"Time response
    quantities");
```

Scilab code Solution 1.2 Step Responses for different Damping ratios

```
//Step Responses of a II order system for zeta=0.1(
    underdamped), zeta=1(critically damped) & zeta
    =1.5(overdamped)
t=0:0.0000001:0.0002;
zeta=[0.5 1 1.5];
```

```
cv=[1 2 3];
s=%s;
for n=1:3
num = 10^10;
den = s^2 + 2*zeta(n)*100000*s  +10^10; //wn=100k
    rad/sec
P = syslin('c',num,den);
Ps=csim('step',t,P);
plot2d(t,Ps,style=cv(n));
end;
xgrid;
xtitle(['Step Responses of a II order system for
    zeta=0.1(underdamped), zeta=1(critically damped)
    & zeta=1.5(overdamped)'],'Time', 'Amplitude' );
legends(['zeta=0.5';'zeta=1';'zeta=1.5'],[1,2,3],opt
    =4);
```

Experiment: 2

Evaluation of effect of additional poles & zeroes on Time response of second order system

Scilab code Solution 2.1 Addition of Poles to Open Loop transfer function

```
//Effect of Adding a Pole (1+Tp s)to OL tr fn G=wn^2/(s(s+2zeta wn))on CL tr fn T=wn^2/(Tp s^3+(1+2zeta wn Tp)s^2+2zeta wn s+wn^2)of a II order system
s=%s;
t=0:0.1:30;
zeta=1;  wn=1;
Tp=[0 1 2 5];      //Poles added are s=-1/Tp i.e poles at -1/0, -1/1, -1/2, -1/5
line_style=[1 2 3 4];        // for dashed, dotted ,.... lines
for n=1:4
  T=syslin('c',wn^2,(Tp(n)*s^3+(1+2*zeta *wn*Tp(n))*s^2+2*zeta *wn *s+wn^2));
```

```
  Ts=csim('step',t,T);
  xset("line style",line_style(n));
  plot2d(t,Ts,style=1);  // style=1 for black line
end;
xgrid(3);                    //green grid
xtitle(['Effect of Adding a Pole (1+Tp s) to OL tr fn G = wn^2 / ( s (s + 2zeta wn))'],'Time(sec)','c(t)' );
legends(['s = -1/0';'s = -1/1';'s = -1/2';'s = -1/5'],[[1;1],[1;2],[1;3],[1;4]],opt=4);
```

Scilab code Solution 2.2 Adding Zeroes to Open Loop transfer function

```
//Effect of Adding a Zero,(1+Tz s)to OL tr fn G=wn^2/(s(s+2zeta wn))on CL tr fn T=wn^2(1+Tz s)/(s^2+(2zeta wn + wn^2 Tz)s + wn^2)of a II order system
s=%s;
t=0:0.001:20;
zeta=0.1;  wn=1;
Tz=[0 0.5 2 5];              //Zeroes added are s=-1/Tz i.e zeroes at -1/0, -1/0.5, -1/2, -1/5
line_style=[1 2 3 4];          // for dashed, dotted ,.... lines
for n=1:4
  T=syslin('c',wn^2*(1+Tz(n)*s),(s^2+(2*zeta*wn+ Tz(n)*wn^2)*s+wn^2));
  Ts=csim('step',t,T);
  xset("line style",line_style(n));
  plot2d(t,Ts,style=1);  // style=1 for black line
end;
xgrid(3);                    //green grid
xtitle(['Effect of Adding a Zero (1+Tz s) to OL tr fn G = wn^2 / ( s ( s + 2zeta wn))'],'Time(sec)', 'c(t)' );
```

```
legends(['s = -1/0';'s = -1/0.5';'s = -1/2';'s =
    -1/5'],[[1;1],[1;2],[1;3],[1;4]],opt=4);
```

Scilab code Solution 2.3 Adding Poles to LClosed Loop transfer function

```
// Effect of addition of poles to CL tr fn T(s)
    =100/(s^2+4s+100) on its Time response
s=%s;
a=[15 4];

for n=1:3
if n==1 then
   num=100
   den=(s^2 +4*s + 100)
else
   num=100*a(n-1)
   den=(s^2 +4*s + 100)*(s+a(n-1));
end

T=syslin('c',num,den);
t=0:0.005:5;
Ts=csim('step',t,T);
xset("line style",n)
plot2d(t,Ts);
xgrid(3);                          // 3 - light shade (
    green) grid
end
xtitle('Effect of addition of poles to CL tr fn T(s)
    =100/(s^2+4s+100) on its Time response ','t(sec)'
    ,' C (t)');
legends(['Original tr.fn.','Added pole at s = - 15',
    'Added pole at s = - 4'],[[1;1],[1;2],[1;3]],opt
    =4);
```

Scilab code Solution 2.4 Adding Zeroes to Closed Loop transfer function

```
// Effect of Addition of Zeroes to CL tr fn T(s)=100/(s^2+4s+100) on its Time response
s=%s;
a=[8 2];
den=(s^2 +2*s + 9);

for n=1:3
    if n==1 then
    num=9
    else
    num=9*(s+a(n-1))/a(n-1);
end

T=syslin('c',num,den);

t=0:0.005:5;
Ts=csim('step',t,T);
xset("line style",n)
plot2d(t,Ts);
xgrid(3);                          // 3 - light shade (green) grid
end
xtitle('Effect of Addition of Zeroes to CL tr fn T(s)=100/(s^2+4s+100) on its Time response','t(sec)','C (t)');
legends(['Original tr.fn.','Added zero at -8','Added zero at -2'],[[1;1],[1;2],[1;3]],opt=4);
```

Experiment: 3

Evaluation of effect of pole location on stability

Scilab code Solution 3.3 Effect of Pole Location on Stability

```
//Evaluation of effect of Pole location on stability of II order system
s=%s;
t=0:0.001:20;
wn=1;

//Poles on -ve real axis(zeta>1)
zeta=2 ;
R=roots(s^2 + 2*zeta*wn*s + wn^2)  // R(1) = -3.7320508 , R(2) = - 0.2679492
T=syslin('c',wn^2,(s-R(1))*(s-R(2)))//T=syslin('c',wn^2,s^2 + 2*zeta*wn*s + wn^2);
Ts1=csim('step',t,T);
subplot(231)
xtitle("Poles on -ve Real axis (zeta>1)")
plot(t,Ts1);
xgrid;

//Equal Poles on -ve Real axis (zeta=1)
```

```
zeta=1 ;
R=roots(s^2 + 2*zeta*wn*s + wn^2)  // R(1) = R(2) = -1
T=syslin('c',wn^2,(s-R(1))*(s-R(2)))//T=syslin('c',wn^2,s^2 + 2*zeta*wn*s + wn^2);
Ts1=csim('step',t,T);
subplot(232)
xtitle("Equal Poles on -ve Real axis (zeta=1)")
plot(t,Ts1);
xgrid;

//Complex conjugate Poles with -ve Real part (0<zeta<1)
zeta=0.5 ;
R=roots(s^2 + 2*zeta*wn*s + wn^2) // R(1)= -0.5+0.8660254i, R(2)= -0.5 -0.8660254i
T=syslin('c',wn^2,(s-R(1))*(s-R(2)))//T=syslin('c',wn^2,s^2 + 2*zeta*wn*s + wn^2);
Ts1=csim('step',t,T);
subplot(233)
xtitle("Complex conj Poles with -ve Real part (0<zeta<1)")
plot(t,Ts1);
xgrid;

//Complex conj Poles on Imag axis (zeta=0)
zeta=0 ;
R=roots(s^2 + 2*zeta*wn*s + wn^2)  // R(1)= i, R(2) = -i
T=syslin('c',wn^2,(s-R(1))*(s-R(2)))//T=syslin('c',wn^2,s^2 + 2*zeta*wn*s + wn^2);
Ts1=csim('step',t,T);
subplot(234)
xtitle("Complex conj Poles on Imag axis (zeta=0)")
plot(t,Ts1);
xgrid;

//Complex conj Poles with +ve Real part (0>zeta>-1)
```

```
zeta=-0.5 ;
R=roots(s^2 + 2*zeta*wn*s + wn^2)   // R(1)= 0.5 + 0.8660254i, R(2) = 0.5 - 0.8660254i
T=syslin('c',wn^2,(s-R(1))*(s-R(2)))//T=syslin('c',wn^2,s^2 + 2*zeta*wn*s + wn^2);
Ts1=csim('step',t,T);
subplot(235)
xtitle("Complex conj Poles with +ve Real part (0>zeta>-1)")
plot(t,Ts1);
xgrid;

//Poles on +ve Real axis (zeta<-1)
zeta=-1.2 ;
R=roots(s^2 + 2*zeta*wn*s + wn^2)   // R(1)= 5.8284271, R(2) = 0.1715729
T=syslin('c',wn^2,(s-R(1))*(s-R(2)))//T=syslin('c',wn^2,s^2 + 2*zeta*wn*s + wn^2);
Ts1=csim('step',t,T);
subplot(236)
xtitle("Poles on +ve Real axis (zeta<-1)")
plot(t,Ts1);
xgrid;
```

Experiment: 4

Effect of loop gain of a negative feedback system on stability

Scilab code Solution 4.1 Effect of Loop Gain on stability

```
// Effect of Loop Gain K of a Negative feedback
   system on Stability.
// G(s) = wn^2 / s(s+2zeta wn) , H(s) = K , T(s) = K
    wn^2 / ( s(s+2zeta wn) + K wn^2 )
s=%s;
t=0:0.01:10;
wn=1;zeta=1;
K=[1,2,5,10]
for n=1:4
    T=syslin('c', K(n)*wn^2 , s*(s + 2*zeta*wn) + K(
       n)*wn^2 );
    Ts=csim('step',t,T);
    xset("line style",n);
    plot2d(t,Ts);
    xgrid(3);
end
xtitle('Effect of Loop Gain K of a - ve feedback
   system on Stability.','Time(sec)','C(t)');
legends(['K=1';'K=2';'K=5';'K=10'
```

```
;],[[1;1],[1;2],[1;3],[1;4]], opt=4);
```

Experiment: 5

To examine the relationships between open-loop frequency response and stability, open loop frequency and C.L transient response

Scilab code Solution 5.1 Relation between Open Loop Frequency Response and Closed Loop Transient Response

```
//OpenLoop Frequency Response & ClosedLoop Transient
    Response

// 1) Correlation b/w ub(Normalized bandwidth) &
   zeta(Damping factor) for a II order system
deff("[wbbywn]=f1(zeta)","wbbywn=sqrt(1-2*zeta^2+
   sqrt(2-4*zeta^2+4*zeta^4))")
zeta=[0:0.01:0.9];// don't end with 1 bec, division
   by 0 error
subplot(221)
fplot2d(zeta,f1,[1])
xgrid(3)
```

```
xtitle(['Correlation b/w ub(Normalized bandwidth) & zeta for a II order system'],'zeta (Damping ratio)', 'wb / wn' );

// 2) Correlation b/w Mp(Peak overshoot) & Mr(Resonance Peak) for a II order system
deff("[Mp]=f2(zeta)","Mp=exp((-%pi*zeta)/sqrt(1-zeta^2))")
deff("[Mr]=f3(zeta)","Mr=1/(2*zeta*sqrt(1-zeta^2))")
zeta=[0.05:0.01:0.9];  //don't start from 0 & end with 0 because, division by 0 error
subplot(222)
xset("line style",4);
fplot2d(zeta,f2,[1])
xset("line style",1);
fplot2d(zeta,f3,[1])
xgrid(3)
xtitle(['Correlation between Mp & Mr for a II order system'],'zeta (Damping ratio)', 'Mp, Mr' );
legends(['Mp (Peak Gain)';'Mr (Gain at Resonance)'],[[1;4],[1;1]],opt=1);

// 3) Correlation between wr(Resonant frequency) & wd(Damped frequency) for a II order system
deff("[wrbywd]=f4(zeta)","wrbywd=sqrt(1-2*zeta^2)/sqrt(1-zeta^2)")
zeta=[0:0.01:0.9];// don't end with 1 bec, division by 0 error
subplot(223)
fplot2d(zeta,f4,[1])
xgrid(3)
xtitle(['Correlation between wr & wd for a II order system'],'zeta (Damping ratio)', 'wr / wd' );
```