

Scilab Manual for
Control Systems
by Mrs Supanna S Kumar
Electrical Engineering
KLE Dr M.S Sheshgiri College of Engg &
Tech¹

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

June 7, 2026

¹Funded 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 Simulation of a typical second order system & determination of step response eval of time domain specifications	4
2 Evaluation of effect of additional poles & zeroes on Time response of second order system	7
3 Evaluation of effect of pole location on stability	11
4 Effect of loop gain of a negative feedback system on stability	14
5 To examine the relationships between open-loop frequency response and stability, open loop frequency and C.L transient response	16

List of Experiments

Solution 1.1	Step Response and Time domain specifications . . .	4
Solution 1.2	Step Responses for different Damping ratios . . .	5
Solution 2.1	Addition of Poles to Open Loop transfer function	7
Solution 2.2	Adding Zeroes to Open Loop transfer function . . .	8
Solution 2.3	Adding Poles to LClosed Loop transfer function .	9
Solution 2.4	Adding Zeroes to Closed Loop transfer function .	10
Solution 3.3	Effect of Pole Location on Stability	11
Solution 4.1	Effect of Loop Gain on stability	14
Solution 5.1	Relation between Open Loop Frequency Response and Closed Loop Transient Response	16

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

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

```

    2 * zeta * Wn = z(1,2)
12 zeta=z(1,2)/(2*Wn)           // zeta = d a m
    p i n g f a c t o r
13 Wd=Wn*sqrt(1-zeta^2)
14 Tp=%pi/Wd                   // Tp= p
    e a k t i m e
15 Mp=100*exp((-%pi*zeta)/sqrt(1-zeta^2)) // Mp=p e
    a k o v e r s h o o t
16 Td=(1+0.7*zeta)/Wn         // Td= d e
    l a y t i m e
17 a=atan(sqrt(1-zeta^2)/zeta)
18 Tr=(%pi-a)/Wd             // T r = r
    i s e t i m e
19 Tset=4/(zeta*Wn)          // T s =
    s e t t l i n g t i m e
20
21 Peak_time = sprintf("Peak Time = %6.3f secs",Tp);
22 Peak_overshoot = sprintf("Peak Overshoot = %6.3f
    percent",Mp);
23 Delay_time = sprintf("Delay_time = %6.3f secs",Td);
24 Rise_time = sprintf("Rise_time = %6.3f secs",Tr);
25 Settling_time = sprintf("Settling_time = %6.3f secs"
    ,Tset);
26
27 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

```

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

```

```

4 cv=[1 2 3];
5 s=%s;
6 for n=1:3
7 num = 10^10;
8 den = s^2 + 2*zeta(n)*100000*s +10^10; //wn=100k
      rad/sec
9 P = syslin('c',num,den);
10 Ps=csim('step',t,P);
11 plot2d(t,Ps,style=cv(n));
12 end;
13 xgrid;
14 xtitle(['Step Responses of a II order system for
          zeta=0.1(underdamped), zeta=1(critically damped)
          & zeta=1.5(overdamped)'], 'Time', 'Amplitude' );
15 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

```
1 // 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+2
   zeta wn Tp)s^2+2zeta wn s+wn^2) of a II order
   system
2 s=%s;
3 t=0:0.1:30;
4 zeta=1; wn=1;
5 Tp=[0 1 2 5]; // Poles added are s=-1/Tp i.e
   poles at -1/0, -1/1, -1/2, -1/5
6 line_style=[1 2 3 4]; // for dashed, dotted
   ,.... lines
7 for n=1:4
8     T=syslin('c',wn^2,(Tp(n)*s^3+(1+2*zeta *wn*Tp(n))
   *s^2+2*zeta *wn *s+wn^2));
```

```

9     Ts=csim('step',t,T);
10    xset("line_style",line_style(n));
11    plot2d(t,Ts,style=1); // style=1 for black line
12 end;
13 xgrid(3); //green grid
14 xtitle(['Effect of Adding a Pole (1+Tp s) to OL tr
fn G = wn^2 / ( s ( s + 2zeta wn))'], 'Time(sec)',
'c(t)');
15 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

```

1 //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
2 s=%s;
3 t=0:0.001:20;
4 zeta=0.1; wn=1;
5 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
6 line_style=[1 2 3 4]; // for dashed, dotted
,.... lines
7 for n=1:4
8 T=syslin('c',wn^2*(1+Tz(n)*s),(s^2+(2*zeta*wn+ Tz
(n)*wn^2)*s+wn^2));
9 Ts=csim('step',t,T);
10 xset("line_style",line_style(n));
11 plot2d(t,Ts,style=1); // style=1 for black line
12 end;
13 xgrid(3); //green grid
14 xtitle(['Effect of Adding a Zero (1+Tz s) to OL tr
fn G = wn^2 / ( s ( s + 2zeta wn))'], 'Time(sec)',
'c(t)');

```

```

15 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

```

1 // Effect of addition of poles to CL tr fn T(s)
  =100/(s^2+4s+100) on its Time response
2 s=%s;
3 a=[15 4];
4
5 for n=1:3
6 if n==1 then
7     num=100
8     den=(s^2 +4*s + 100)
9 else
10    num=100*a(n-1)
11    den=(s^2 +4*s + 100)*(s+a(n-1));
12 end
13
14 T=syslin('c',num,den);
15 t=0:0.005:5;
16 Ts=csim('step',t,T);
17 xset("line style",n)
18 plot2d(t,Ts);
19 xgrid(3); // 3 - light shade (
    green) grid
20 end
21 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)');
22 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

```
1 // Effect of Addition of Zeroes to CL tr fn T(s)
   =100/(s^2+4s+100) on its Time response
2 s=%s;
3 a=[8 2];
4 den=(s^2 +2*s + 9);
5
6 for n=1:3
7     if n==1 then
8         num=9
9         else
10            num=9*(s+a(n-1))/a(n-1);
11 end
12
13 T=syslin('c',num,den);
14
15 t=0:0.005:5;
16 Ts=csim('step',t,T);
17 xset("line style",n)
18 plot2d(t,Ts);
19 xgrid(3); // 3 - light shade (
   green) grid
20 end
21 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)');
22 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

```
1 //Evaluation of effect of Pole location on stability
  of II order system
2 s=%s;
3 t=0:0.001:20;
4 wn=1;
5
6 //Poles on -ve real axis(zeta>1)
7 zeta=2 ;
8 R=roots(s^2 + 2*zeta*wn*s + wn^2) // R(1) = -
   3.7320508 , R(2) = - 0.2679492
9 T=syslin('c',wn^2,(s-R(1))*(s-R(2)))/T=syslin('c',
   wn^2,s^2 + 2*zeta*wn*s + wn^2);
10 Ts1=csim('step',t,T);
11 subplot(231)
12 xtitle("Poles on -ve Real axis (zeta>1)")
13 plot(t,Ts1);
14 xgrid;
15
16 //Equal Poles on -ve Real axis (zeta=1)
```

```

17 zeta=1 ;
18 R=roots(s^2 + 2*zeta*wn*s + wn^2) // R(1) = R(2) =
    -1
19 T=syslin('c',wn^2,(s-R(1))*(s-R(2)))/T=syslin('c',
    wn^2,s^2 + 2*zeta*wn*s + wn^2);
20 Ts1=csim('step',t,T);
21 subplot(232)
22 xtitle("Equal Poles on -ve Real axis (zeta=1)")
23 plot(t,Ts1);
24 xgrid;
25
26 //Complex conjugate Poles with -ve Real part (0<zeta
    <1)
27 zeta=0.5 ;
28 R=roots(s^2 + 2*zeta*wn*s + wn^2) // R(1)= -0.5
    +0.8660254i, R(2)= -0.5 -0.8660254i
29 T=syslin('c',wn^2,(s-R(1))*(s-R(2)))/T=syslin('c',
    wn^2,s^2 + 2*zeta*wn*s + wn^2);
30 Ts1=csim('step',t,T);
31 subplot(233)
32 xtitle("Complex conj Poles with -ve Real part (0<
    zeta<1)")
33 plot(t,Ts1);
34 xgrid;
35
36 //Complex conj Poles on Imag axis (zeta=0)
37 zeta=0 ;
38 R=roots(s^2 + 2*zeta*wn*s + wn^2) // R(1)= i, R(2)
    = -i
39 T=syslin('c',wn^2,(s-R(1))*(s-R(2)))/T=syslin('c',
    wn^2,s^2 + 2*zeta*wn*s + wn^2);
40 Ts1=csim('step',t,T);
41 subplot(234)
42 xtitle("Complex conj Poles on Imag axis (zeta=0)")
43 plot(t,Ts1);
44 xgrid;
45
46 //Complex conj Poles with +ve Real part (0>zeta>-1)

```

```

47 zeta=-0.5 ;
48 R=roots(s^2 + 2*zeta*wn*s + wn^2) // R(1)= 0.5 +
    0.8660254i, R(2) = 0.5 - 0.8660254i
49 T=syslin('c',wn^2,(s-R(1))*(s-R(2)))/T=syslin('c',
    wn^2,s^2 + 2*zeta*wn*s + wn^2);
50 Ts1=csim('step',t,T);
51 subplot(235)
52 xtitle("Complex conj Poles with +ve Real part (0 >
    zeta > -1)")
53 plot(t,Ts1);
54 xgrid;
55
56 //Poles on +ve Real axis (zeta < -1)
57 zeta=-1.2 ;
58 R=roots(s^2 + 2*zeta*wn*s + wn^2) // R(1)=
    5.8284271, R(2) = 0.1715729
59 T=syslin('c',wn^2,(s-R(1))*(s-R(2)))/T=syslin('c',
    wn^2,s^2 + 2*zeta*wn*s + wn^2);
60 Ts1=csim('step',t,T);
61 subplot(236)
62 xtitle("Poles on +ve Real axis (zeta < -1)")
63 plot(t,Ts1);
64 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

```
1 // Effect of Loop Gain K of a Negative feedback
  system on Stability.
2 //  $G(s) = \frac{\omega_n^2}{s(s+2\zeta\omega_n)}$  ,  $H(s) = K$  ,  $T(s) = \frac{K\omega_n^2}{s(s+2\zeta\omega_n) + K\omega_n^2}$ 
3 s=%s;
4 t=0:0.01:10;
5 wn=1;zeta=1;
6 K=[1,2,5,10]
7 for n=1:4
8     T=syslin('c', K(n)*wn^2 , s*(s + 2*zeta*wn) + K(
        n)*wn^2 );
9     Ts=csim('step',t,T);
10    xset("line style",n);
11    plot2d(t,Ts);
12    xgrid(3);
13 end
14 xtitle('Effect of Loop Gain K of a - ve feedback
    system on Stability.', 'Time(sec)', 'C(t)');
15 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

```
1 //OpenLoop Frequency Response & ClosedLoop Transient
   Response
2
3 // 1) Correlation b/w ub(Normalized bandwidth) &
   zeta(Damping factor) for a II order system
4 def f1(zeta) = sqrt(1-2*zeta^2+sqrt(2-4*zeta^2+4*zeta^4))
5 zeta=[0:0.01:0.9]; // don't end with 1 bec, division
   by 0 error
6 subplot(2,2,1)
7 fplot2d(zeta,f1,[1])
8 xgrid(3)
```

```

9  xtitle(['Correlation b/w ub(Normalized bandwidth) &
        zeta for a II order system'], 'zeta (Damping ratio)
        ', 'wb / wn' );
10
11 // 2) Correlation b/w Mp(Peak overshoot) & Mr(
        Resonance Peak) for a II order system
12 deff(" [Mp]=f2 ( zeta )", "Mp=exp((-%pi*zeta)/sqrt(1-zeta
        ^2))")
13 deff(" [Mr]=f3 ( zeta )", "Mr=1/(2*zeta*sqrt(1-zeta ^2))")
14 zeta=[0.05:0.01:0.9]; //don't start from 0 & end
        with 0 because, division by 0 error
15 subplot(222)
16 xset("line style",4);
17 fplot2d(zeta,f2,[1])
18 xset("line style",1);
19 fplot2d(zeta,f3,[1])
20 xgrid(3)
21 xtitle(['Correlation between Mp & Mr for a II order
        system'], 'zeta (Damping ratio)', 'Mp, Mr' );
22 legends(['Mp (Peak Gain)'; 'Mr (Gain at Resonance)
        '], [[1;4], [1;1]], opt=1);
23
24 // 3) Correlation between wr(Resonant frequency) &
        wd(Damped frequency) for a II order system
25 deff(" [wrbywd]=f4 ( zeta )", "wrbywd=sqrt(1-2*zeta ^2)/
        sqrt(1-zeta ^2)")
26 zeta=[0:0.01:0.9]; // don't end with 1 bec, division
        by 0 error
27 subplot(223)
28 fplot2d(zeta,f4,[1])
29 xgrid(3)
30 xtitle(['Correlation between wr & wd for a II order
        system'], 'zeta (Damping ratio)', 'wr / wd' );

```
