Scilab Manual for Control Theory by Prof Shweta Gaur Instrumentation Engineering Dharmsinh Desai University[1]

Solutions provided by
Prof Shweta Gaur
Instrumentation Engineering
Dharmsinh Desai University

July 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 Experiments

List of Figures

Experiment: 1

Introduction to Control Systems

Scilab code Solution 1.5 To find Laplace Transform of given function

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 5 of Lab 1
// To find Laplace Transform of given function f(t)
   = sin(2*t)*cos(3*t)

clc
close

syms t s
F=laplace (sin(2*t)*cos(2*t),t,s)
disp (F,"F( s )=")
```

Scilab code Solution 1.6 To find Inverse Laplace Transform of given function

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 6 of Lab 1
// To find Inverse Laplace Transform of given
   function F(s)=56*s/(s^4+106*s^2+2025)

clc
close

syms s t
F=56*s/(s^4+106*s^2+2025)
f=ilaplace(F,s,t)
disp (f," f ( t )=")
```

Scilab code Solution 1.7 To find Response of given system with respect to Step input

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 7 of Lab 1
```

```
// To find Response of given system w.r.t. Step
   input

clc
close

syms t s

TF=5/(s^2+6*s+101)
R=1/s

C=R*TF

disp(C,"Response = ")
```

Scilab code Solution 1.8 To find Response of given system with respect to Ramp input

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 8 of Lab 1
// To find Response of given system w.r.t. Ramp
   input

clc
close

syms t s

TF=10/((s+2)*(s+3))
R=1/s^2
```

```
C=R*TF

disp(C,"Response = ")
```

Experiment: 2

Transfer Function

Scilab code Solution 2.3 To find Transfer function of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 3 of Lab 2
// To find Transfer function of given system

clc
close

syms s

z1=-2
p1=-1
p2=0
p3=-3

tf=(s-z1)/((s-p1)*(s-p2)*(s-p3))
disp(tf,"Transfer function = ")
```

Scilab code Solution 2.4 To find Transfer function of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 4 of Lab 2
// To find Transfer function of given system

clc
close

syms s R1 R2 C1 C2

//Preparing SFG for given circuit
//i1=(ei-V1)/R1
//V1=(i1-i2)*(1/(C1*s))
//i2=(V1-e0)/R2
//e0=i2*(1/(C2*s))

//Using Masons gain formula
P1=(1/R1)*(1/(C1*s))*(1/R2)*(1/(C2*s))
L1=(1/(C1*s))*(-1/R1)
L2=(1/R2)*(-1/(C1*s))
L3=(1/(C2*s))*(-1/R2)
L11=L1*L3
del1=1
del=1-(L1+L2+L3)+L11
tf=P1*del1/del
disp(tf,"Transfer function = ")
```

Scilab code Solution 2.5 To find Transfer function of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 5 of Lab 2
// To find Transfer function of given system

clc
close

syms s R1 R2 C

//Preparing SFG for given circuit
//i1=(ei-V1)/R1
//ei=R1*i1+R2*i1+(1/(C*s))*i1
//V1=i1*(R2+(1/(C*s)))
//e2=V1

//Using Masons gain formula
P1=(1/R1)*(R2+(1/(C*s)))*1
L1=(R2+(1/(C*s)))*(-1/R1)
del=1-L1
del1=1

tf=P1*del1/del
```

```
disp(tf,"Transfer function = ")
```

Scilab code Solution 2.6 To find Transfer function of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 6 of Lab 2
// To find Transfer function of given system

clc
close

syms s R1 R2 R3 R4 L1 L2 L3

//Preparing SFG for given circuit
//i1=(ei-V1)/(R1+L1*s)
//V1=(i1-i2)*(R2+L2*s)
//i2=(V1-e0)/(R3+L3*s)
//e0=R4*i2

//Using Masons gain formula
P1=R4*(1/(R1+L1*s))*(R2+L2*s)*(1/(R3+L3*s))
L1=-((R2+L2*s)/(R1+L1*s))
L2=-((R2+L2*s)/(R3+L3*s))
L3=-R4/(R3+L3*s)
L11=L1*L3
del1=1
del=1-(L1+L2+L3)+L11

tf=P1*del1/del
disp(tf,"Transfer function = ")
```

Scilab code Solution 2.7 To find Transfer function of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 7 of Lab 2
// To find Transfer function of given system

clc
close

syms s R1 R2 Rb C1 C2 L1 L2

//Preparing SFG for given circuit
//i1=(ei-V1)/Rb
//V1=(i1-i2)*(1/(C1*s))+V2
//i2=(V1-e0)/(R1+L1*s)
//V2=i1*(R2+L2*s)
//e0-V2=i2/(C2*s)

//Using Masons gain formula
P1=(1/Rb)*(1/(C1*s))*(1/(R1+L1*s))*(1/(C2*s))
P2=(1/Rb)*(R2+L2*s)*1
L1=(1/(C1*s))*(-1/Rb)
L2=(1/(R1+L1*s))*(-1/(C1*s))
L3=(1/(C2*s))*(-1/(R1+L1*s))
L4=(R2+L2*s)*1*(-1/Rb)
L5=(R2+L2*s)*1*(-1/(R1*L1*s))*(-1/(C1*s))*(-1/Rb)
L11=L1*L3

del1=1
```

```
del2=1-L2
del=1-(L1+L2+L3+L4+L5)+L11
tf=(P1*del1+P2*del2)/del
disp(tf,"Transfer function = ")
```

Experiment: 3

Mathematical Modelling

Scilab code Solution 3.4 To find Reflected Inertia and Reflected Coulomb Friction for given gear train system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 4 of Lab 3
// To find Reflected Inertia and Reflected Coulomb
   Friction for given gear train system

clc
close

J2=0.08;
T2=5;
teeth_ratio=1/10
J1=(teeth_ratio)^2*J2;
T1=(teeth_ratio)*T2

printf("Reflected Inertia = %f oz-in.-sec^3\n",J1)
```

```
printf("Reflected Coulumb Friction = %f oz-in.\n",T1
    )
```

Scilab code Solution 3.5 To find Transfer function of Mass spring system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 5 of Lab 3
// To find Transfer function of Mass spring system

clc
close

syms K M B
s=%s
P1=(1/M)*(s^-2)
L11=-(B/M)*(s^-1)
L21=-(K/M)*(s^-2)
del=1-(L11+L21)
del1=1
tf=P1*del1/del
disp(tf,"Transfer function = ")
```

Experiment: 4

Signal Flow Graph

Scilab code Solution 4.7 To obtain Transfer function using Masons gain formula

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 7 of Lab 4
// To obtain Transfer function using Mason's gain
   formula

clc
close

syms g1 g2 g3 g4 h1 h2

p1=g1*g2*g3
p2=g4
l1=-g2*h1
l2=g1*g2*h1
l3=-(g3*g2*h2)
```

```
d=1-(l1+l2+l3)
d1=1
d2=d
tf=(p1*d1+p2*d2)/(d)

disp(tf,"Transfer funtion = ")
```

Scilab code Solution 4.8 To obtain Transfer function using Masons gain formula

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 8 of Lab 4
// To obtain Transfer function using Mason's gain
   formula

clc
close

syms g1 g2 g3 g4 h1 h2

p1=g1*g3
p2=g1*g2
p3=g1*g3*g4*h2
p4=g1*g2*g4*h2
d1=1
d2=1
d3=1
d4=1
l1=-g1*g3*h1*h2
l2=-g1*g2*h1*h2
```

```
d=1-(l1+l2)
tf=(p1*d1+p2*d2+p3*d3+p4*d4)/d

disp(tf,"Transfer function = ")
```

Scilab code Solution 4.9 To obtain Transfer function using Masons gain formula

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 9 of Lab 4
// To obtain Transfer function using Mason's gain
   formula

clc
close

syms g1 g2 g3 g4 g5 g6 g7 g8 h1 h2

p1=g2*g4*g6
p2=g3*g5*g7
p3=g2*g1*g7
p4=g3*g6*g8
p5=-g2*g1*h2*g8*g6
p6=-g3*g8*h1*g1*g7
l1=-g4*h1
l2=-g5*h2
l3=g1*h2*g8*h1
d=1+g4*h1+g5*h2-g1*h2*g8*h1+g4*h1*g5*h2
d1=1+g5*h2
d2=1+g4*h1
```

```
d3=1
d4=1
d5=1
d6=1
tf=(p1*d1+p2*d2+p3*d3+p4*d4+p5*d5+p6*d6)/d

disp(tf,"Transfer function = ")
```

Scilab code Solution 4.10 To obtain Transfer function using Masons gain formula

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 10 of Lab 4
// To obtain Transfer function using Mason's gain
   formula

clc
close

syms t1 t2 t3 q1 q2 s
p1=t1/(s*(s+q1))
p2=t2/(s+q1)
p3=t3
l1=q2/(s*(s+q1))
d=1+(q2/(s*(s+q1)))
d1=1
d2=1
d3=1+(q2/(s*(s+q1)))
tf=(p1*d1+p2*d2+p3*d3)/d

```

```
disp(tf,"Transfer function = ")
```

Scilab code Solution 4.11 To obtain Transfer function using Masons gain formula

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 11 of Lab 4
// To obtain Transfer function using Mason's gain
   formula

clc
close

syms g1 g2 g3 g4 h1 h2

p1=g1*g2*g3
p2=g1*g4
l1=-g1*g2*h1
l2=-g2*g3*h2
l3=-g1*g2*g3
l4=-g4*h2
l5=-g4*g1
d=1+g1*g2*h1+g2*g3*h2+g1*g2*g3+g4*h2+g1*g4
d1=1
d2=1
tf=(p1*d1+p2*d2)/(d)

disp(tf,"Transfer function = ")
```

Scilab code Solution 4.12 To obtain Transfer function using Masons gain formula

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 12 of Lab 4
// To obtain Transfer function using Mason's gain formula

clc
close

syms g1 g2 g3 g4 h1 h2

p1=g1*g2*g3
p2=g4
l1=-g1*g2*h1
l2=-g1*g2*g3
l3=-g2*g3*h2
l4-g4
l5=g4*h2*g2*h1
d=1-(l1+l2+l3+l4+l5)
d1=1
d2=1
tf=(p1*d1+p2*d2)/d

disp(tf,"Transfer function = ")
```

Scilab code Solution 4.13 To obtain Transfer function using Masons gain formula

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 13 of Lab 4
// To obtain Transfer function using Mason's gain
   formula

clc
close

syms g1 g2 g3 g4 h1 h2 h3 h4

p1=g1*g2*g3
p2=g4*g3
l1=g3*g4*h1*h2
l2=g1*g2*g3*h1*h2
l3=-g1*h2*h3
d=1+g1*h2*h3-g3*g4*h1*h2-g1*g2*g3*h1*h2
d1=1
d2=1
tf=(p1*d1+p2*d2)/d

disp(tf,"Transfer function = ")
```

Scilab code Solution 4.14 To obtain Transfer function using Masons gain formula

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 14 of Lab 4
// To obtain Transfer function using Mason's gain formula

clc
close

syms g1 g2 g3 g4 g5 g6 g7 g8 g9

p1=g1*g2
p2=g4
p3=g7*g8
p4=g1*g5*g8
p5=g7*g6*g2
l1=g9
l2=g3
l3=g5*g6
d=1-(g3+g9+g5*g6)+g9*g3
d1=1-g9
d2=1-(g9+g3+g5*g6)+g9*g3
d3=1-g3
d4=1
d5=1
tf=(p1*d1+p2*d2+p3*d3+p4*d4+p5*d5)/d

disp(tf,"Transfer function = ")
```

Experiment: 5

Block Diagram Reduction Techniques

Scilab code Solution 5.7 To solve Cascade of two functions

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 7 of Lab 1
// To solve Cascade of two functions
clc
close

syms s

G1=5/(s*(s^2+s+2))
G2=(5*s+3)/((s+1)*(s+2))
G=G1*G2

disp(G,"Resultant function G(s) = ")
```

Scilab code Solution 5.8 To solve Parallel combination of two functions

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 8 of Lab 1
// To solve Parallel combination of two functions
clc
close

syms s

G1=19/((s+5)*(s+99))
G2=(10*s+9)/((s+3)*(s+5))
G=G1/G2

disp(G,"Resultant function G(s) = ")
```

Scilab code Solution 5.9 To find Transfer function for a closed loop

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 9 of Lab 1
// To find Transfer function for a closed loop
clc
```

```
close

syms s

G=10/(s*(s+20))
H=2
TF=G/(1+G*H)

disp(TF,"Transfer function = ")
```

Experiment: 6

Feedback Characteristics

Scilab code Solution 6.6 To find sensitivity of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 6 of Lab 6
// To find sensitivity of given system

clc
close

funcprot(0);
s=%i;
h=.25;
g=25/(s^2+s)
sg_complex=1/(1+g*h)
sg=abs(sg_complex)
disp(sg,"Sensitivity with respect to G=")
sh_complex=(-g*h/(1+g*h))
sh=abs(sh_complex)
```

```
disp(sh,"Sensitivity with respect to H=")
```

Scilab code Solution 6.7 To find sensitivity of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 7 of Lab 6
// To find sensitivity of given system

clc
close

funcprot(0);
s=%i*2;
h=1;
g=5/(s^2+s*2)
sg_complex=1/(1+g*h)
sg=abs(sg_complex)
disp(sg,"Sensitivity with respect to G=")
sh_complex=(-g*h/(1+g*h))
sh=abs(sh_complex)
disp(sh,"Sensitivity with respect to H=")
```

Scilab code Solution 6.8 To find sensitivity of given system

```
// OS : Windows 7
// Scilab : 5.4.1
```

```
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 8 of Lab 6
// To find sensitivity of given system

clc
close

// FOR w=1 rad/sec;
funcprot(0);
s=1*%i;
h=10;
g=10/(s^2+s)
sg_complex=1/(1+g*h)
sg=abs(sg_complex)
disp("For w=1 rad/sec")
disp(sg,"    Sensitivity with respect to G=")
sh_complex=(-g*h/(1+g*h))
sh=abs(sh_complex)
disp(sh,"    Sensitivity with respect to H=")

// FOR w=2 rad/sec;
funcprot(0);
s=2*%i;
h=10;
g=10/(s^2+s)
sg_complex=1/(1+g*h)
sg=abs(sg_complex)
disp("For w=2 rad/sec")
disp(sg,"    Sensitivity with respect to G=")
sh_complex=(-g*h/(1+g*h))
sh=abs(sh_complex)
disp(sh,"    Sensitivity with respect to H=")
```

Scilab code Solution 6.9 To find sensitivity of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 9 of Lab 6
// To find sensitivity of given system

clc
close

funcprot(0);
s=1*%i;
h=1;
k=1;
g=k/(s^2+15*s)
deff('t=f(k,s)','t=k/(s^2+15*s+k)');
sg_complex=(derivative(f,k))*(k/(k/(s^2+15*s+k)))
sg=abs(sg_complex)
disp(sg,"Sensitivity with respect to K=")
```

Scilab code Solution 6.10 To find sensitivity of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 10 of Lab 6
// To find sensitivity of given system

clc
close
```

```
funcprot(0);
s=%i;
T=input("enter the value of T=")  // we will ask
   user to input the value of 'T'
h=1;
k=10;
g=k/(T*s+1)
sg_complex=1/(1+g*h)
sg=abs(sg_complex)
disp(sg,"Sensitivity with respect to G=")
```

Scilab code Solution 6.11 To find sensitivity of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 11 of Lab 6
// To find sensitivity of given system

clc
close

funcprot(0);
s=.5*%i;
h=1;
k=1;
g=25*k/(s^2+5*s)
sg_complex=1/(1+g*h)
sg=abs(sg_complex)

disp(sg,"    Sensitivity with respect to G=")
```

```
sh_complex=(-g*h/(1+g*h))
sh=abs(sh_complex)
disp(sh,"    Sensitivity with respect to H=")
```

Experiment: 7

Time Response Analysis

Scilab code Solution 7.6 To find Time response parameters for the given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 6 of Lab 7
// To find Time response parameters for the given
   system

clc
close

s=poly(0,'s');
g=100/(s^2+10*s)
tf=g/(1+g)

// To compare tf with Wn^2/(s^2+2*zeta*Wn+Wn^2)
y=denom(tf)
q=numer(tf)
```

```
z=coeff(y)
r=coeff(q)
wn=sqrt(r)

zeta=z(1,2)/(2*wn)
wd=(wn*sqrt(1-zeta^2))
mp=((%e^(-%pi*zeta/(sqrt(1-zeta^2))))*100)
theta=atan(sqrt(1-zeta^2)/zeta);
tr=((%pi-theta)/wd)
tp=(%pi/wd)
ts=(4/(zeta*wn))

printf("Undamped natural frequency = %f rad/sec \n",
    wn)
printf("Damping ratio = %f \n",zeta)
printf("Damped frequency = %f rad/sec \n",wd)
printf("Maximum Peak Overshoot = %f percent \n",mp)
printf("Theta = %f rad \n",theta)
printf("Rise time = %f sec \n",tr)
printf("Peak Time = %f sec \n",tp)
printf("Settling time = %f sec \n",ts)
```

Scilab code Solution 7.7 To find Time response parameters for the given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 7 of Lab 7
// To find Time response parameters for the given
    system

```

```
clc
close

s=poly(0,'s');
tf=2025/(s^2+45*s+2025)

// To compare tf with Wn^2/(s^2+2*zeta*Wn+Wn^2)
y=denom(tf)
q=numer(tf)
z=coeff(y)
r=coeff(q)
wn=sqrt(r)

zeta=z(1,2)/(2*wn)
wd=(wn*sqrt(1-zeta^2))
mp=((%e^(-%pi*zeta/(sqrt(1-zeta^2))))*100)
theta=atan(sqrt(1-zeta^2)/zeta);
tr=((%pi-theta)/wd)
tp=(%pi/wd)
ts=(4/(zeta*wn))

printf("Undamped natural frequency = %f rad/sec \n",
    wn)
printf("Damping ratio = %f \n",zeta)
printf("Damped frequency = %f rad/sec \n",wd)
printf("Maximum Peak Overshoot = %f percent \n",mp)
printf("Theta = %f rad \n",theta)
printf("Rise time = %f sec \n",tr)
printf("Peak Time = %f sec \n",tp)
printf("Settling time = %f sec \n",ts)
```

Scilab code Solution 7.8 To find Time response parameters for the given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 8 of Lab 7
// To find Time response parameters for the given system

clc
close

s=poly(0,'s');
g=6.25/(s^2+2*s)
tf=g/(1+g)

// To compare tf with Wn^2/(s^2+2*zeta*Wn+Wn^2)
y=denom(tf)
q=numer(tf)
z=coeff(y)
r=coeff(q)
wn=sqrt(r)

zeta=z(1,2)/(2*wn)
wd=(wn*sqrt(1-zeta^2))
mp=((%e^(-%pi*zeta/(sqrt(1-zeta^2))))*100)
theta=atan(sqrt(1-zeta^2)/zeta);
tr=((%pi-theta)/wd)
tp=(%pi/wd)
ts=(4/(zeta*wn))

printf("Undamped natural frequency = %f rad/sec \n",wn)
printf("Damping ratio = %f \n",zeta)
printf("Damped frequency = %f rad/sec \n",wd)
printf("Maximum Peak Overshoot = %f percent \n",mp)
printf("Theta = %f rad \n",theta)
printf("Rise time = %f sec \n",tr)
printf("Peak Time = %f sec \n",tp)
```

```
printf("Settling time = %f sec \n",ts)
```

Scilab code Solution 7.9 To find Time response parameters for the given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 9 of Lab 7
// To find Time response parameters for the given
   system

clc
close

s=poly(0,'s');
tf=36481/(s^2+191*s+36481)

// To compare tf with Wn^2/(s^2+2*zeta*Wn+Wn^2)
y=denom(tf)
q=numer(tf)
z=coeff(y)
r=coeff(q)
wn=sqrt(r)

zeta=z(1,2)/(2*wn)
wd=(wn*sqrt(1-zeta^2))
mp=((%e^(-%pi*zeta/(sqrt(1-zeta^2))))*100)
theta=atan(sqrt(1-zeta^2)/zeta);
tr=((%pi-theta)/wd)
tp=(%pi/wd)
ts=(4/(zeta*wn))
```

```
printf("Undamped natural frequency = %f rad/sec \n",
   wn)
printf("Damping ratio = %f \n",zeta)
printf("Damped frequency = %f rad/sec \n",wd)
printf("Maximum Peak Overshoot = %f percent \n",mp)
printf("Theta = %f rad \n",theta)
printf("Rise time = %f sec \n",tr)
printf("Peak Time = %f sec \n",tp)
printf("Settling time = %f sec \n",ts)
```

Experiment: 8

Stability of Control Systems

Scilab code Solution 8.6 To find stability of given system using Routh Hurwitz criteria

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 6 of Lab 8
// To find stability of given system using Routh
   Hurwitz criterion

clc
close

s=%s;
p=s^3-4*s^2+s+6;
disp(p,"Given system function = ")
c=coeff(p)
l=length(c)
routh=routh_t(p) //This Function generates the Routh
    table
```

```
disp(routh,"Rouths table = ")
x=0;
for i=1:l
if (routh(i,1)<0)
x=x+1;
end
end
if(x>=1)
    printf("From Rouths table, it is clear that the
       system is unstable.")
else
    printf("From Rouths table, it is clear that the
       system is stable.")
end
```

Scilab code Solution 8.7 To find stability of given system using Routh Hurwitz criteria

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 7 of Lab 8
// To find stability of given system using Routh
   Hurwitz criterion

clc
close

s=%s;
p=2*s^4+s^3+3*s^2+5*s+10;
disp(p,"Given system function = ")
c=coeff(p)
```

```
l=length(c)
routh=routh_t(p) //This Function generates the Routh
    table
disp(routh,"Rouths table = ")
x=0;
for i=1:l
if (routh(i,1)<0)
x=x+1;
end
end
if(x>=1)
    printf("From Rouths table, it is clear that the
       system is unstable.")
else
    printf("From Rouths table, it is clear that the
       system is stable.")
end
```

Scilab code Solution 8.8 To find value of K for marginal stability of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 8 of Lab 8
// To find value of K for marginal stability of
   given system

clc
close

s=%s
```

```
syms K
p=s^3+2*s^2+(1-K)*s+K
coef_a0 = coeffs(p,'s',0);
coef_a1 = coeffs(p,'s',1);
coef_a2 = coeffs(p,'s',2);
coef_a3 = coeffs(p,'s',3);

c=[coef_a0 coef_a1 coef_a2 coef_a3]

l=length(c);
routh=[c([4,2]);c([3,1])];
routh=[routh;-det(routh)/routh(2,1),0];
a=routh(2:3,1:2); //Getting the square sub block of
    routh matrix
routh=[routh;-det(a)/a(2,1),0]
disp(routh,"Rouths table = ")
routh(3,1)=0  //For marginal stability
sys=syslin('c',(1-s)/(s^3+2*s^2+s))
k=kpure(sys)
disp(k,"K(marginal)=")
```

Scilab code Solution 8.9 To find value of K for marginal stability of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 9 of Lab 8
// To find value of K for marginal stability of
    given system

clc
```

```
close

s=%s
syms K
p=s^3+2.5*s^2+20*s+10*K
coef_a0 = coeffs(p,'s',0);
coef_a1 = coeffs(p,'s',1);
coef_a2 = coeffs(p,'s',2);
coef_a3 = coeffs(p,'s',3);

c=[coef_a0 coef_a1 coef_a2 coef_a3]

l=length(c);
routh=[c([4,2]);c([3,1])];
routh=[routh;-det(routh)/routh(2,1),0];
a=routh(2:3,1:2); //Getting the square sub block of
   routh matrix
routh=[routh;-det(a)/a(2,1),0]
disp(routh,"Rouths table = ")
routh(3,1)=0  //For marginal stability
sys=syslin('c',10/(s^3+2.5*s^2+20*s))
k=kpure(sys)
disp(k,"K(marginal)=")
```

Scilab code Solution 8.10 To find value of K for marginal stability of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 10 of Lab 8
// To find value of K for marginal stability of
```

```
    given system

clc
close

s=%s
syms K
p=s^3+10*s^2+(21+K)*s+13*K
coef_a0 = coeffs(p,'s',0);
coef_a1 = coeffs(p,'s',1);
coef_a2 = coeffs(p,'s',2);
coef_a3 = coeffs(p,'s',3);

c=[coef_a0 coef_a1 coef_a2 coef_a3]

l=length(c);
routh=[c([4,2]);c([3,1])];
routh=[routh;-det(routh)/routh(2,1),0];
a=routh(2:3,1:2); //Getting the square sub block of
    routh matrix
routh=[routh;-det(a)/a(2,1),0]
disp(routh,"Rouths table = ")
routh(3,1)=0  //For marginal stability
sys=syslin('c',(s+13)/(s^3+10*s^2+21*s))
k=kpure(sys)
disp(k,"K(marginal)=")
```

Experiment: 9

Root Locus

Scilab code Solution 9.3 To sketch Root locus of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 3 of Lab 9
// To sketch Root locus of given system

clc
close
clf

s=%s
h=syslin('c',1/(s^3))
evans(h,100)
sgrid()
```

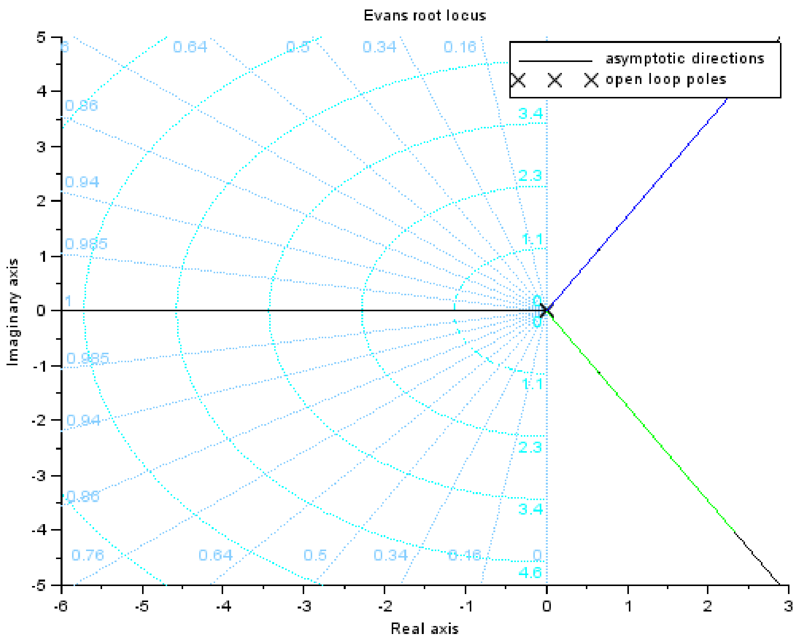

Figure 9.1: To sketch Root locus of given system

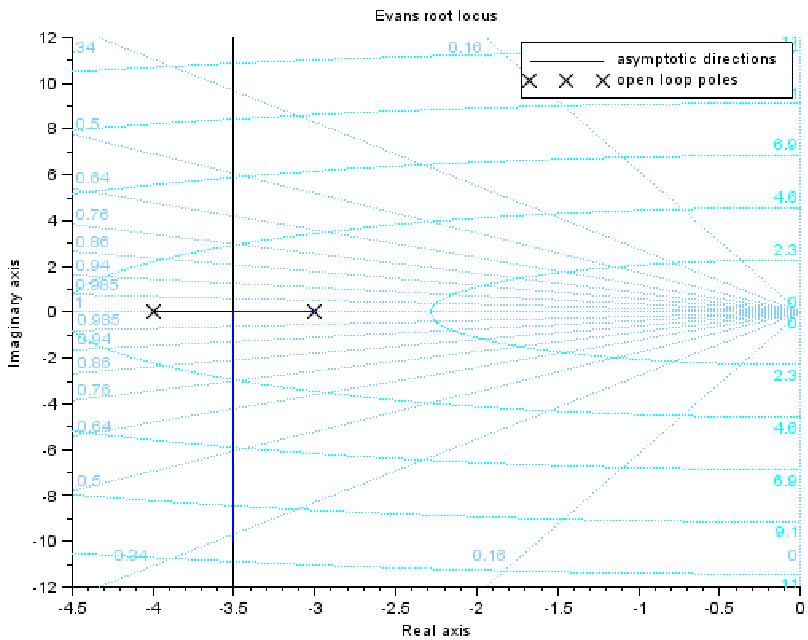

Figure 9.2: To sketch Root locus of given system

Scilab code Solution 9.4 To sketch Root locus of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 4 of Lab 9
// To sketch Root locus of given system

clc
close
clf

s=%s
h=syslin('c',1/((s+4)*(s+3)))
evans(h,100)
sgrid()
```

Scilab code Solution 9.5 To sketch Root locus of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 5 of Lab 9
// To sketch Root locus of given system

clc
close
clf
```

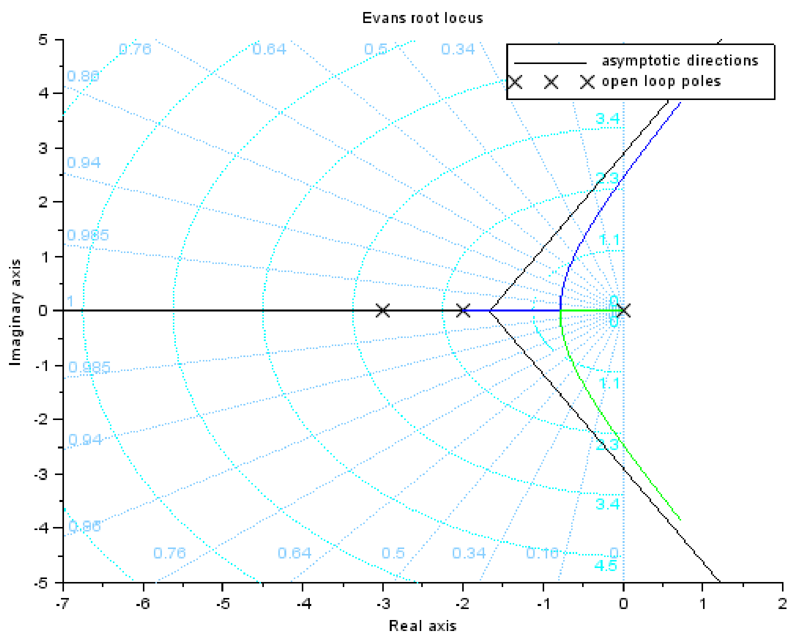

Figure 9.3: To sketch Root locus of given system

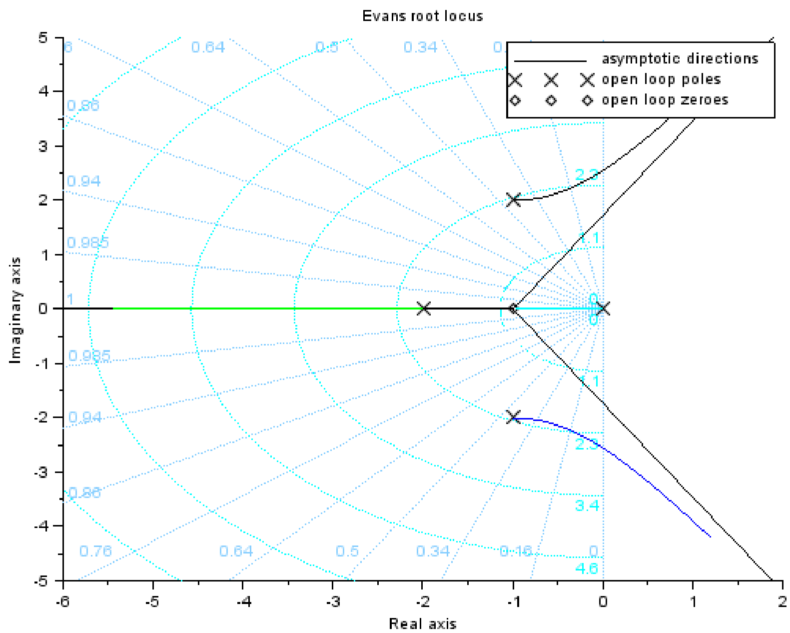

Figure 9.4: To sketch Root locus of given system

```

s=%s
h=syslin('c',1/((s)*(s+2)*(s+3)))
evans(h,100)
sgrid()
```

Scilab code Solution 9.6 To sketch Root locus of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
```

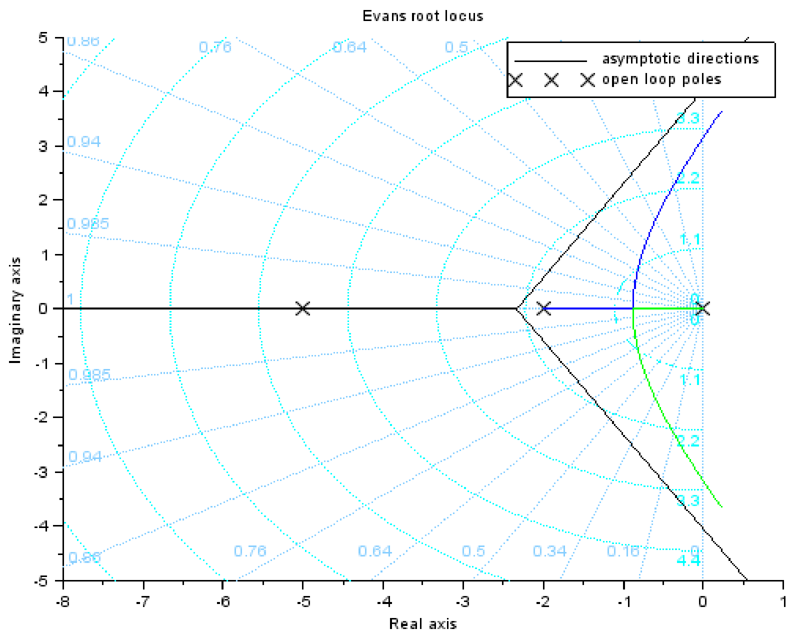

Figure 9.5: To sketch Root locus of given system

```

// Exercise 6 of Lab 9
// To sketch Root locus of given system

clc
close
clf

s=%s
h=syslin('c',(s+1)/(s*(s+2)*(s^2+2*s+5)))
evans(h,100)
sgrid()
```

Scilab code Solution 9.7 To sketch Root locus of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 7 of Lab 9
// To sketch Root locus of given system

clc
close
clf

s=%s
h=syslin('c',1/(s*(s+2)*(s+5)))
evans(h,100)
sgrid()
```

Scilab code Solution 9.8 To sketch Root locus of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 8 of Lab 9
// To sketch Root locus of given system

clc
close
clf
```

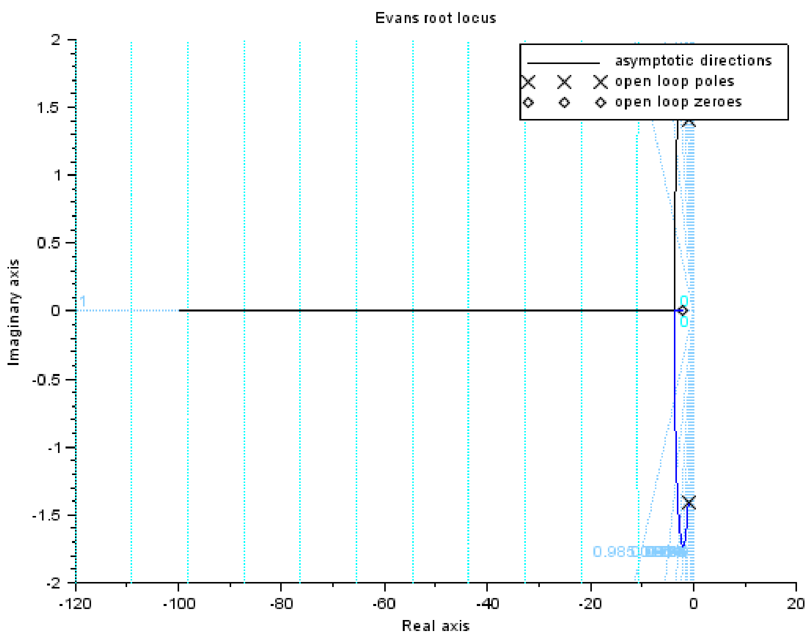

Figure 9.6: To sketch Root locus of given system

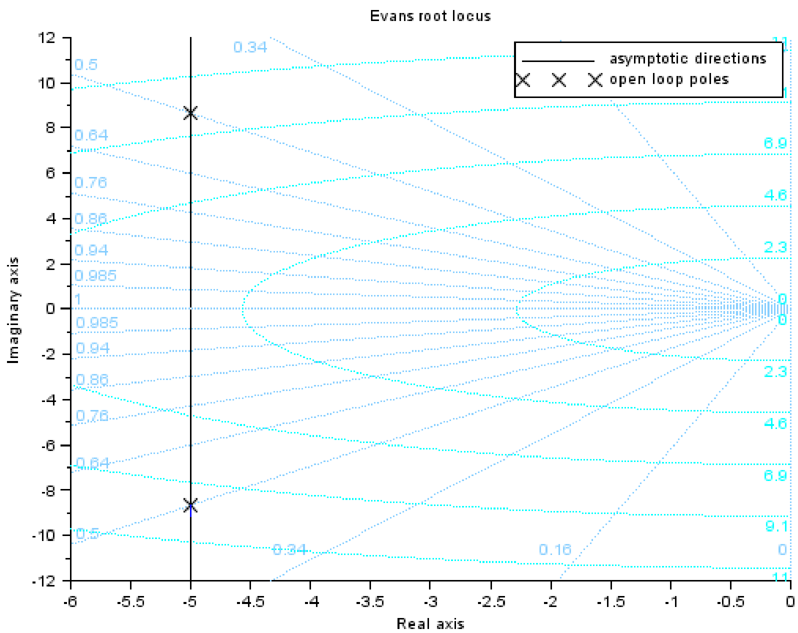

Figure 9.7: To sketch Root locus of given system

```

s=%s
h=syslin('c',(s+2)/(s^2+2*s+3))
evans(h,100)
sgrid()
```

Scilab code Solution 9.9 To sketch Root locus of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
```

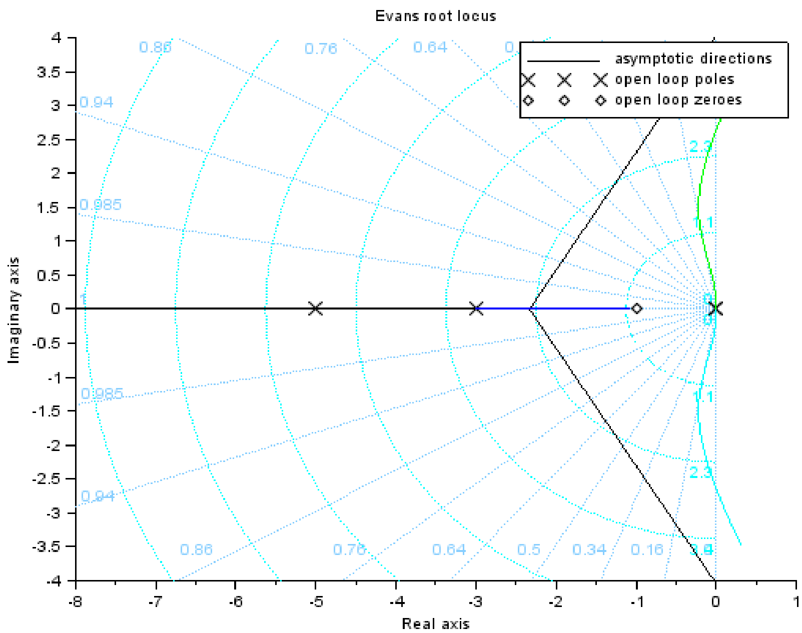

Figure 9.8: To sketch Root locus of given system

```

// Exercise 9 of Lab 9
// To sketch Root locus of given system

clc
close
clf

s=%s
h=syslin('c',1/(s^2+10*s+100))
evans(h,100)
sgrid()
```

Scilab code Solution 9.10 To sketch Root locus of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 10 of Lab 9
// To sketch Root locus of given system

clc
close
clf

s=%s
h=syslin('c',(s+1)/(s^2*(s+3)*(s+5)))
evans(h,100)
sgrid()
```

Scilab code Solution 9.11 To sketch Root locus of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 11 of Lab 9
// To sketch Root locus of given system

clc
close
clf
```

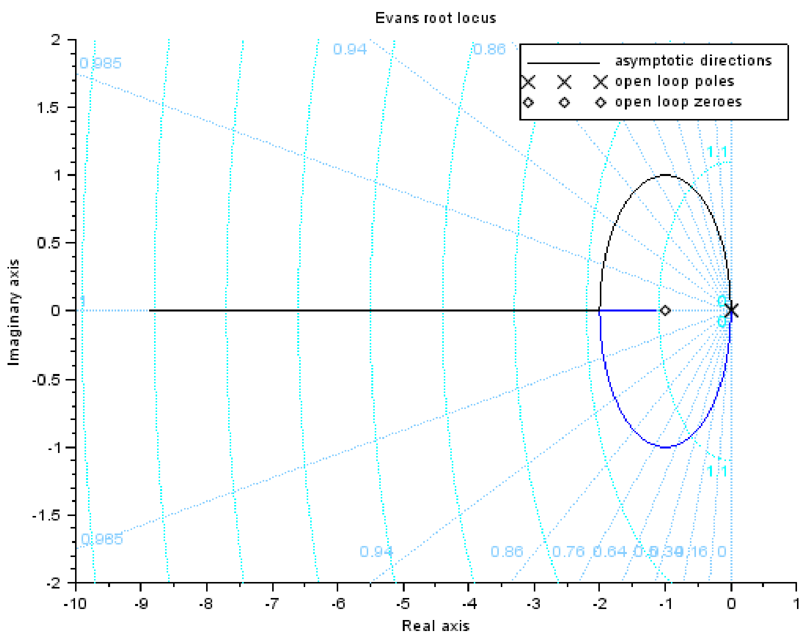

Figure 9.9: To sketch Root locus of given system

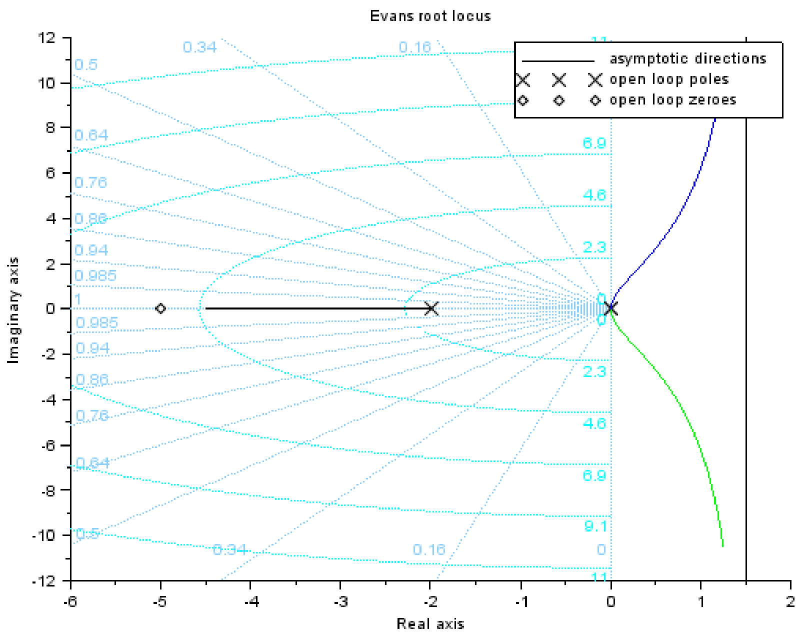

Figure 9.10: To sketch Root locus of given system

```

s=%s
h=syslin('c',(s+1)/(s^2))
evans(h,10)
sgrid()
```

Scilab code Solution 9.12 To sketch Root locus of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
```

```

// Exercise 12 of Lab 9
// To sketch Root locus of given system

clc
close
clf

s=%s
h=syslin('c',(s+5)/(s^2*(s+2)))
evans(h,100)
sgrid()
```

Experiment: 10

Frequency Response Analysis

Scilab code Solution 10.5 To find Frequency domain specifications for the given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 5 of Lab 10
// To find Frequency domain specifications for the
   given system

clc
close

s=poly(0,'s');
g1=3
g2=12/(s^2+3*s)
g=g1*g2
tf=g/(1+g)

// To compare tf with Wn^2/(s^2+2*zeta*Wn+Wn^2)
```

```
y=denom(tf)
q=numer(tf)
z=coeff(y)
r=coeff(q)
wn=sqrt(r)
zeta=z(1,2)/(2*wn)

mr=1/(2*zeta*sqrt(1-zeta^2))
phi_r=-atan(sqrt(1-2*zeta^2)/zeta)
wr=wn*sqrt(1-2*zeta^2)
BW=wn*sqrt(1-2*zeta^2+sqrt(2-4*zeta^2+4*zeta^4))

printf("Resonant Peak = %f \n", mr)
printf("Phase at resonant peak = %f rad \n",phi_r)
printf("Resonant Frequency = %f rad/sec \n",wr)
printf("Bandwidth = %f rad/sec \n",BW)
```

Scilab code Solution 10.6 To find Frequency domain specifications for the given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 6 of Lab 10
// To find Frequency domain specifications for the
   given system

clc
close

s=poly(0,'s');
tf=5/(s^2+2*s+5)
```

```
// To compare tf with Wn^2/(s^2+2*zeta*Wn+Wn^2)
y=denom(tf)
q=numer(tf)
z=coeff(y)
r=coeff(q)
wn=sqrt(r)
zeta=z(1,2)/(2*wn)

mr=1/(2*zeta*sqrt(1-zeta^2))
phi_r=-atan(sqrt(1-2*zeta^2)/zeta)
wr=wn*sqrt(1-2*zeta^2)
BW=wn*sqrt(1-2*zeta^2+sqrt(2-4*zeta^2+4*zeta^4))

printf("Resonant Peak = %f \n", mr)
printf("Phase at resonant peak = %f rad \n",phi_r)
printf("Resonant Frequency = %f rad/sec \n",wr)
printf("Bandwidth = %f rad/sec \n",BW)
```

Scilab code Solution 10.7 To find Frequency domain specifications for the given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 7 of Lab 10
// To find Frequency domain specifications for the given system

clc
close

```

```
s=poly(0,'s');
g=100/(s*(s+8))
tf=g/(1+g)

// To compare tf with Wn^2/(s^2+2*zeta*Wn+Wn^2)
y=denom(tf)
q=numer(tf)
z=coeff(y)
r=coeff(q)
wn=sqrt(r)
zeta=z(1,2)/(2*wn)

mr=1/(2*zeta*sqrt(1-zeta^2))
phi_r=-atan(sqrt(1-2*zeta^2)/zeta)
wr=wn*sqrt(1-2*zeta^2)
BW=wn*sqrt(1-2*zeta^2+sqrt(2-4*zeta^2+4*zeta^4))

printf("Resonant Peak = %f \n", mr)
printf("Phase at resonant peak = %f rad \n",phi_r)
printf("Resonant Frequency = %f rad/sec \n",wr)
printf("Bandwidth = %f rad/sec \n",BW)
```

Scilab code Solution 10.8 To find Frequency domain specifications for the given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 8 of Lab 10
// To find Frequency domain specifications for the
   given system

```

```
clc
close

s=poly(0,'s');
tf=2006/(s^2+10*s+2006)

// To compare tf with Wn^2/(s^2+2*zeta*Wn+Wn^2)
y=denom(tf)
q=numer(tf)
z=coeff(y)
r=coeff(q)
wn=sqrt(r)
zeta=z(1,2)/(2*wn)

mr=1/(2*zeta*sqrt(1-zeta^2))
phi_r=-atan(sqrt(1-2*zeta^2)/zeta)
wr=wn*sqrt(1-2*zeta^2)
BW=wn*sqrt(1-2*zeta^2+sqrt(2-4*zeta^2+4*zeta^4))

printf("Resonant Peak = %f \n", mr)
printf("Phase at resonant peak = %f rad \n",phi_r)
printf("Resonant Frequency = %f rad/sec \n",wr)
printf("Bandwidth = %f rad/sec \n",BW)
```

Experiment: 11

Polar Plots

Scilab code Solution 11.3 To obtain Polar plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 3 of Lab 11
// To sketch Polar plot of given system

clc
close
clf

s=0:.1:2*%pi;
h=10/(1+5*s);
polarplot(s,h);
```

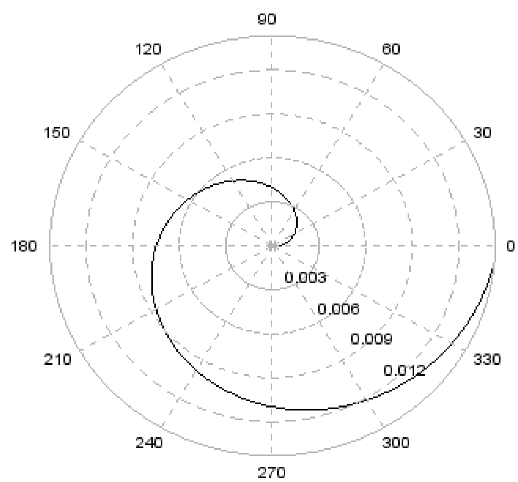

Figure 11.1: To obtain Polar plot of given system

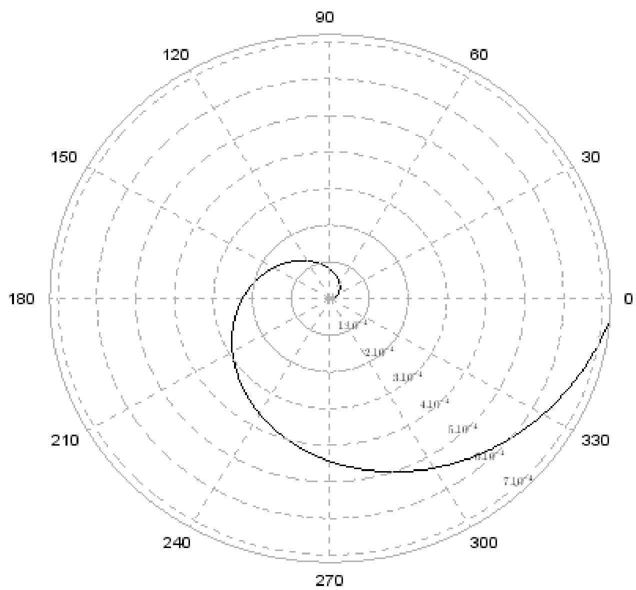

Figure 11.2: To obtain Polar plot of given system

Scilab code Solution 11.4 To obtain Polar plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 4 of Lab 11
// To sketch Polar plot of given system

clc
close
clf

s=0:.1:2*%pi;
h=1/(1+3*s+2*s^2)
polarplot(s,h);
```

Scilab code Solution 11.5 To obtain Polar plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 5 of Lab 11
// To sketch Polar plot of given system

clc
close
clf

s=0:.1:2*%pi*4;
h=1/(s+3*s^2+2*s^3)
polarplot(s,h);
```

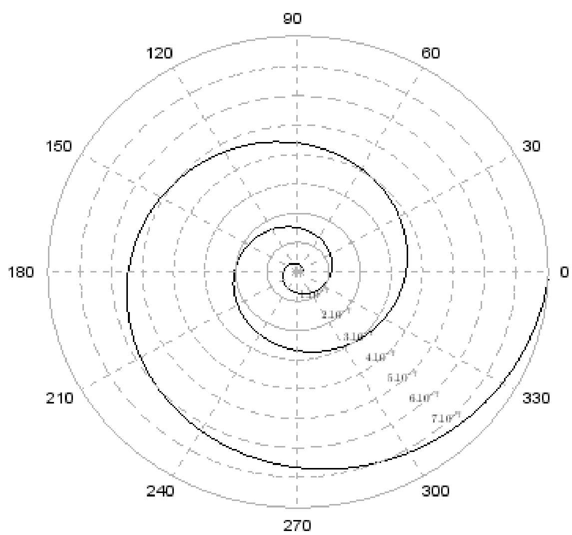

Figure 11.3: To obtain Polar plot of given system

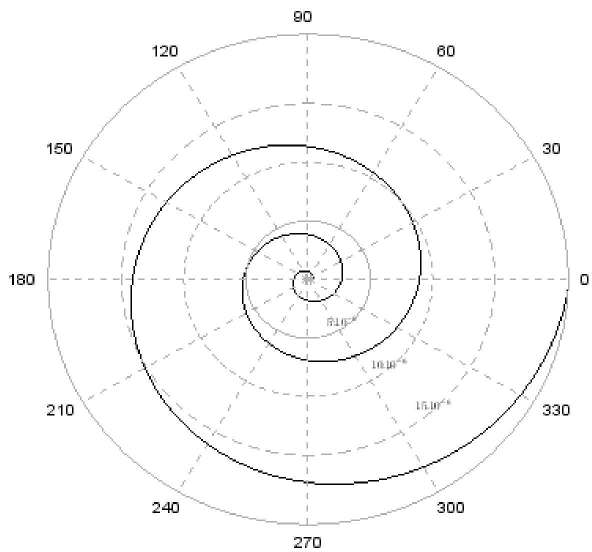

Figure 11.4: To obtain Polar plot of given system

Scilab code Solution 11.6 To obtain Polar plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 6 of Lab 11
// To sketch Polar plot of given system

clc
```

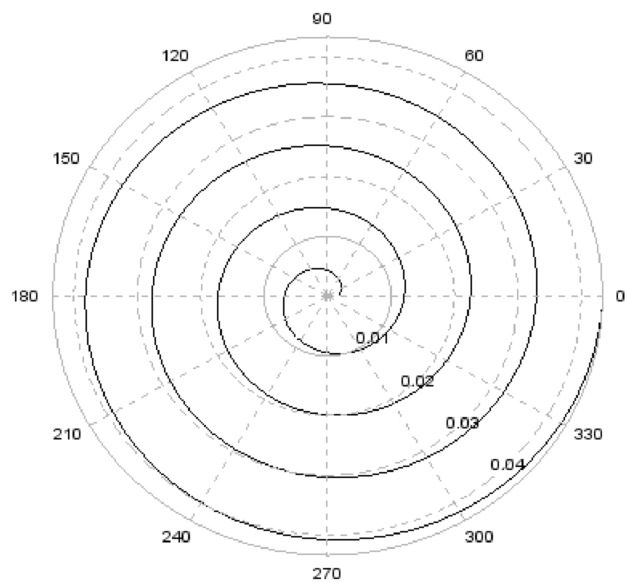

Figure 11.5: To obtain Polar plot of given system

```
close
clf

s=0:.1:2*%pi*4;
h=100/(s+6*s^2+8*s^3)
polarplot(s,h);
```

Scilab code Solution 11.7 To obtain Polar plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 7 of Lab 11
// To sketch Polar plot of given system

clc
close
clf

s=0:.1:2*%pi*4;
h=99/(s+1)
polarplot(s,h);
```

Scilab code Solution 11.8 To obtain Polar plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 8 of Lab 11
// To sketch Polar plot of given system

clc
close
clf

s=0:.1:2*%pi*4;
h=(200/(s+20))
```

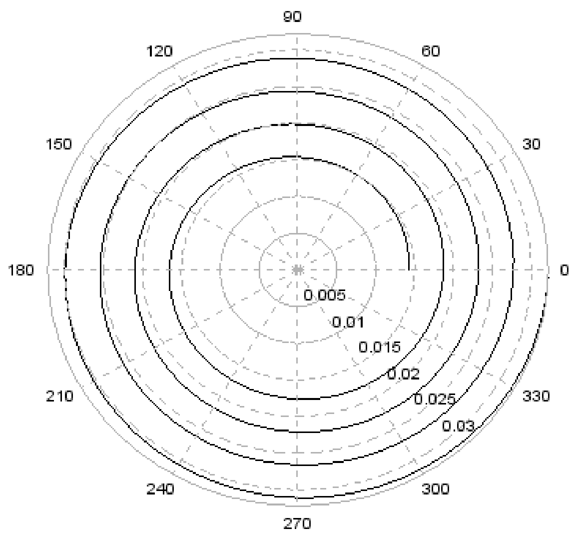

Figure 11.6: To obtain Polar plot of given system

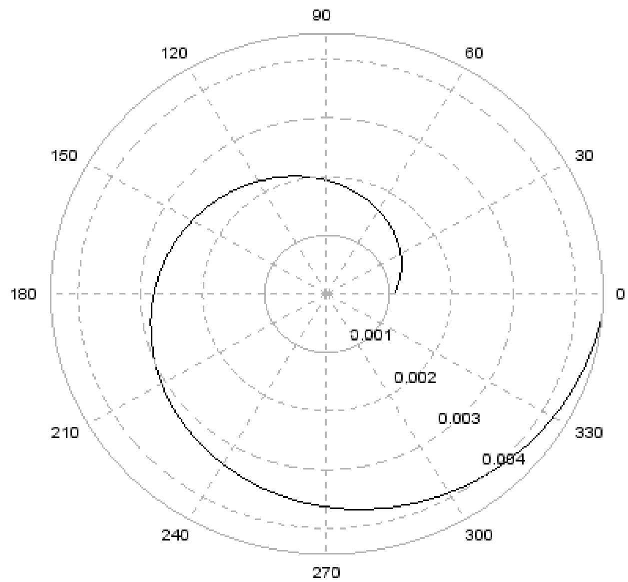

Figure 11.7: To obtain Polar plot of given system

```
polarplot(s,h);
```

Scilab code Solution 11.9 To obtain Polar plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 9 of Lab 11
// To sketch Polar plot of given system

```

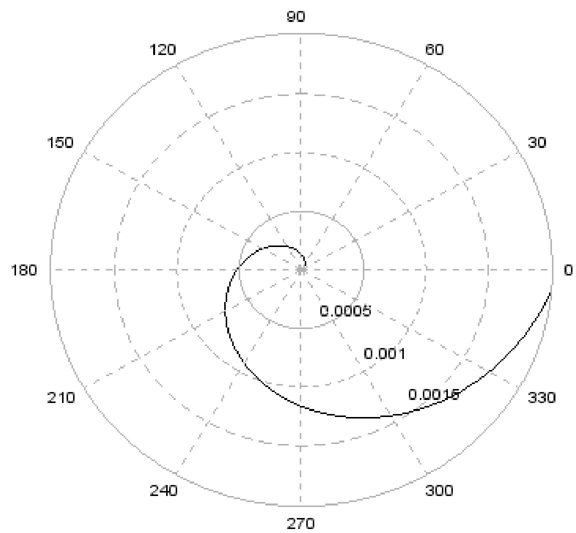

Figure 11.8: To obtain Polar plot of given system

```
clc
close
clf

s=0:.1:2*%pi;
h=(1/(s+2))
polarplot(s,h);
```

Scilab code Solution 11.10 To obtain Polar plot of given system

```
// OS : Windows 7
```

```
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 10 of Lab 11
// To sketch Polar plot of given system

clc
close
clf

s=0:.1:2*%pi;
h=(1/(s^2))
polarplot(s,h);
```

Scilab code Solution 11.11 To obtain Polar plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 11 of Lab 11
// To sketch Polar plot of given system

clc
close
clf

s=0:.1:2*%pi;
h=(1/(s^3))
polarplot(s,h);
```

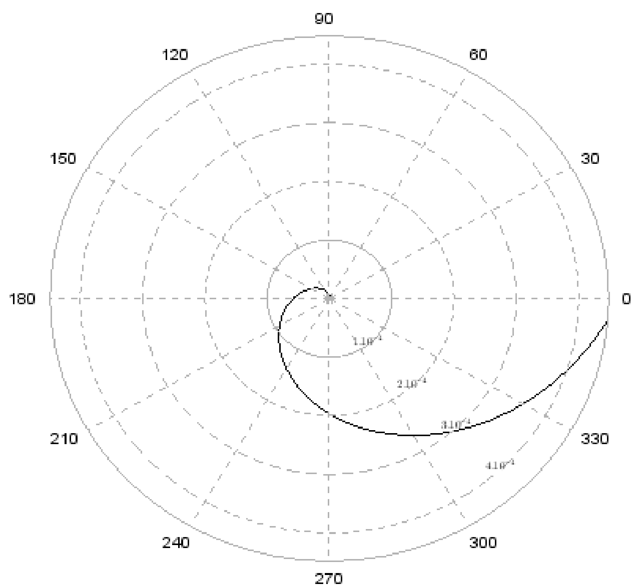

Figure 11.9: To obtain Polar plot of given system

Experiment: 12

Nyquist Plots

Scilab code Solution 12.3 To obtain Nyquist plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 3 of Lab 12
// To obtain Nyquist plot for given system

clc
close

s=poly(0,'s')
h=syslin('c',(s*4+1)/((s^2*(s+1)*(2*s+1))))
nyquist(h);
```

Scilab code Solution 12.4 To obtain Nyquist plot of given system

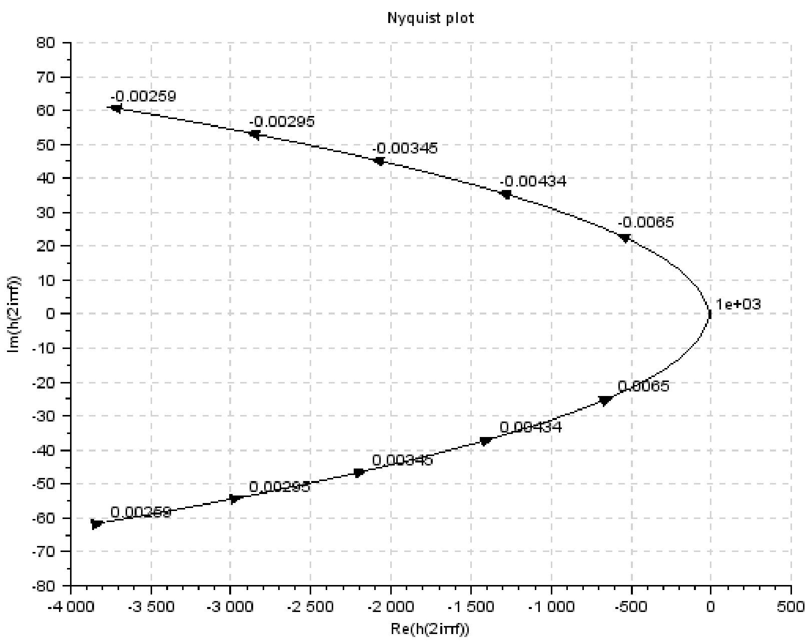

Figure 12.1: To obtain Nyquist plot of given system

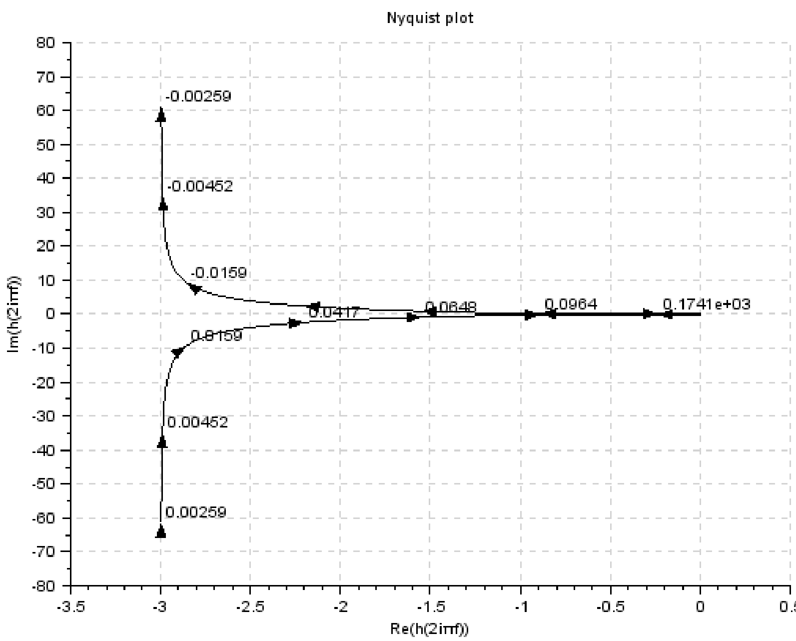

Figure 12.2: To obtain Nyquist plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 4 of Lab 12
// To obtain Nyquist plot for given system

clc
close

s=poly(0,'s')
h=syslin('c',(1)/((s*(s+1)*(2*s+1))))
nyquist(h);
```

Scilab code Solution 12.5 To obtain Nyquist plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 5 of Lab 12
// To obtain Nyquist plot for given system

clc
close

s=poly(0,'s')
h=syslin('c',(s+2)/((s+1)*(s-1)))
nyquist(h);
```

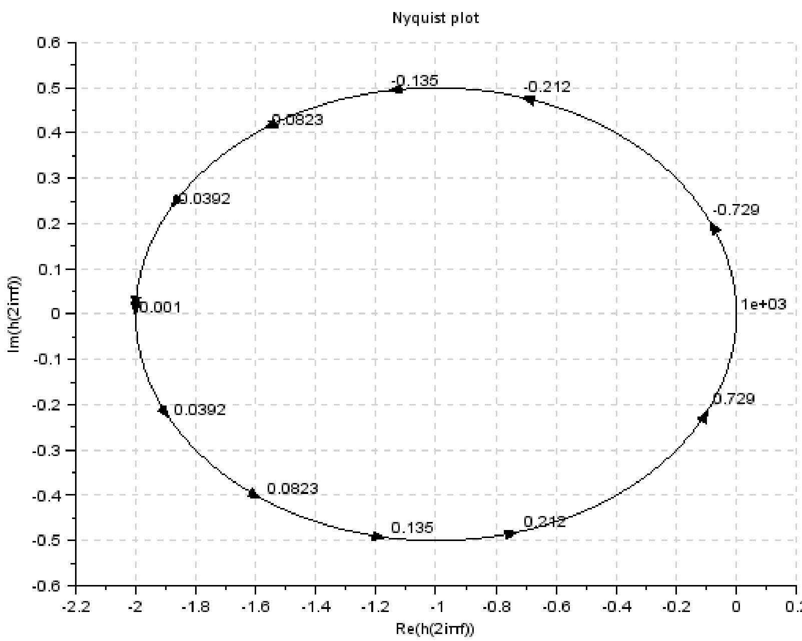

Figure 12.3: To obtain Nyquist plot of given system

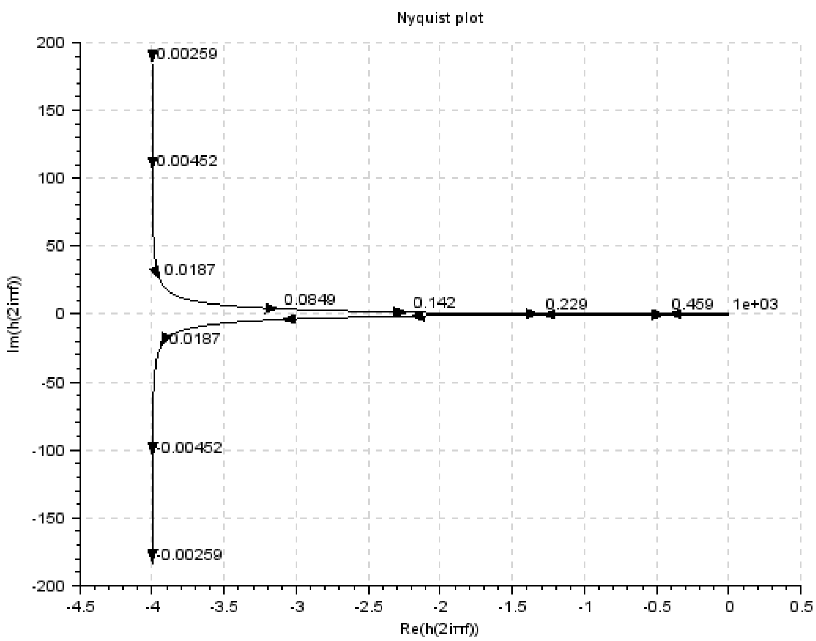

Figure 12.4: To obtain Nyquist plot of given system

Scilab code Solution 12.6 To obtain Nyquist plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 6 of Lab 12
// To obtain Nyquist plot for given system

clc
close

s=poly(0,'s')
h=syslin('c',(s+3)/((s*(s-1))))
nyquist(h);
```

Scilab code Solution 12.7 To obtain Nyquist plot of given system

```





// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 7 of Lab 12
// To obtain Nyquist plot for given system

```

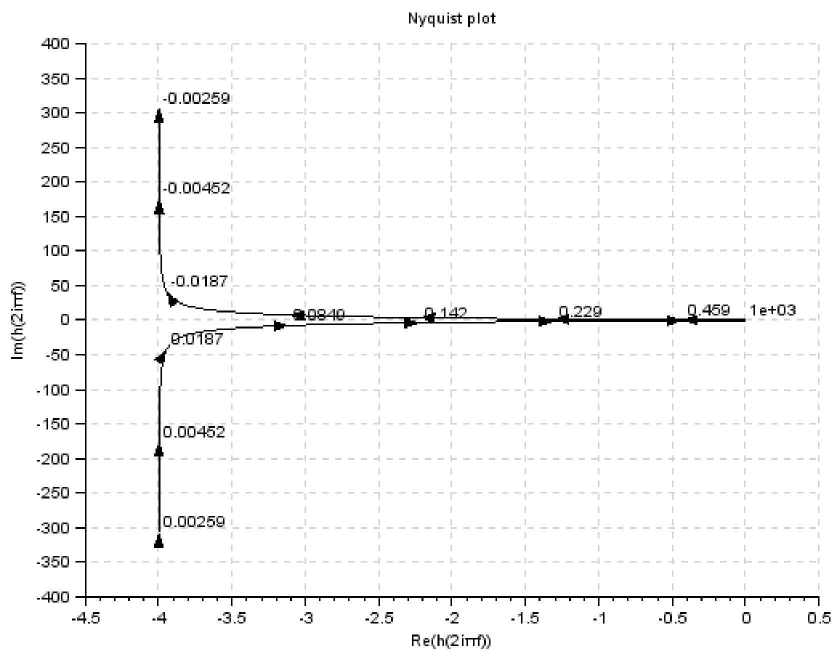

Figure 12.5: To obtain Nyquist plot of given system

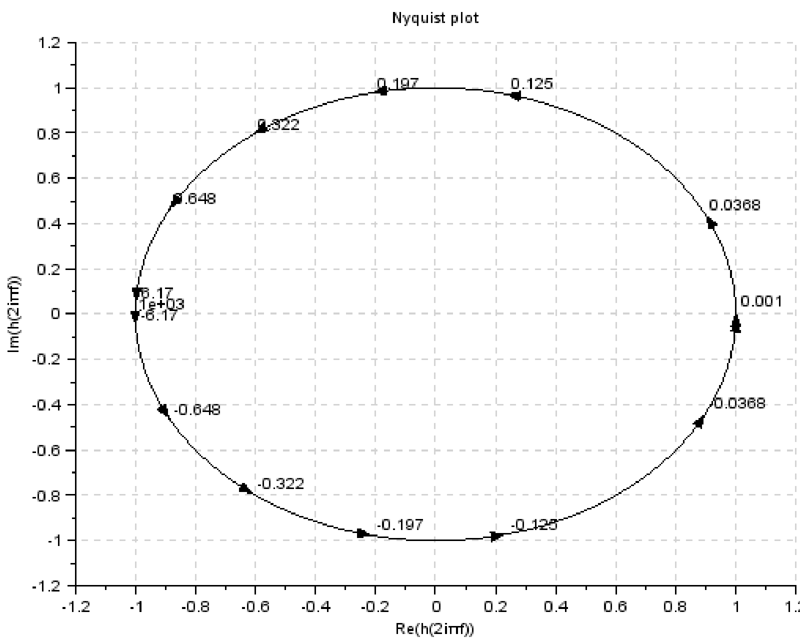

Figure 12.6: To obtain Nyquist plot of given system

```
clc
close

s=poly(0,'s')
h=syslin('c',(s+5)/((s*(s+1))))
nyquist(h);
```

Scilab code Solution 12.8 To obtain Nyquist plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
```

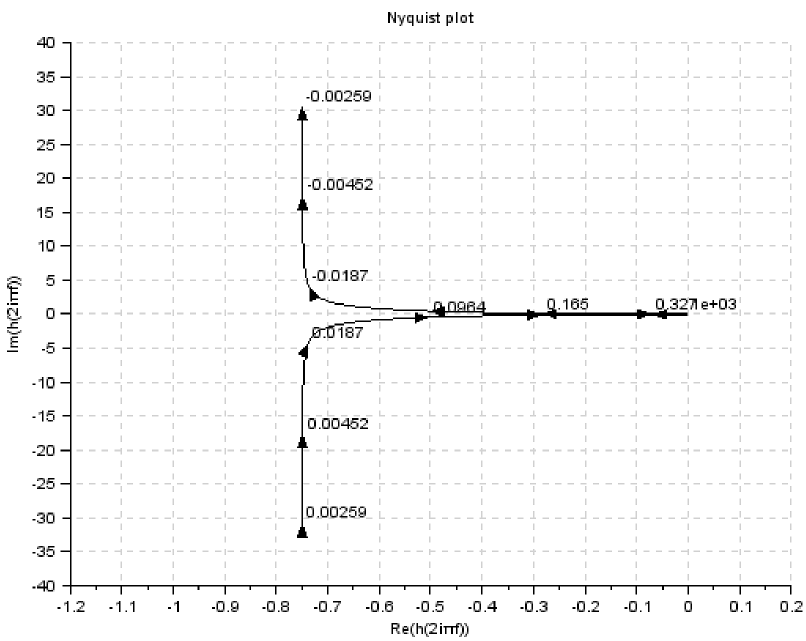

Figure 12.7: To obtain Nyquist plot of given system

```
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 8 of Lab 12
// To obtain Nyquist plot for given system

clc
close

s=poly(0,'s')
h=syslin('c',(1+s)/(1-s))
nyquist(h)
```

Scilab code Solution 12.9 To obtain Nyquist plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 9 of Lab 12
// To obtain Nyquist plot for given system

clc
close

s=poly(0,'s')
h=syslin('c',1/((s*(s+1)*(s+2))))
nyquist(h);
```

Scilab code Solution 12.10 To obtain Nyquist plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 10 of Lab 12
// To obtain Nyquist plot for given system

clc
close

s=poly(0,'s')
h=syslin('c',(s^2+2*s+3)/(s+2))
nyquist(h,0.000001,1)
```

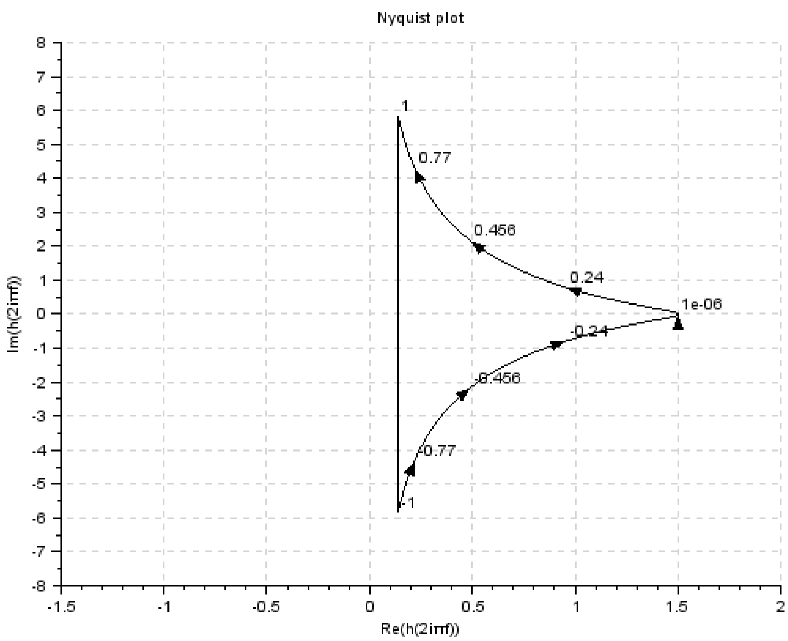

Figure 12.8: To obtain Nyquist plot of given system

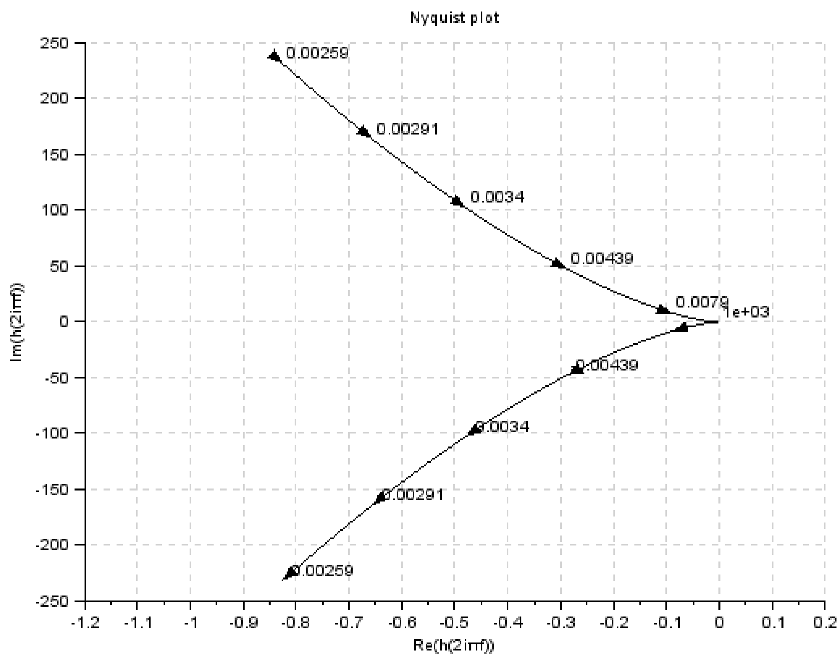

Figure 12.9: To obtain Nyquist plot of given system

Scilab code Solution 12.11 To obtain Nyquist plot of given system

```





// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
```

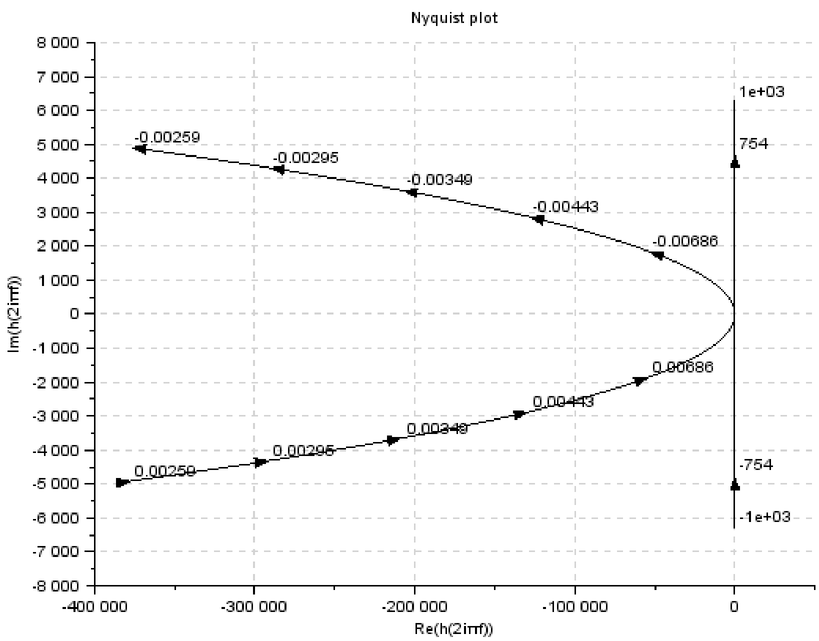

Figure 12.10: To obtain Nyquist plot of given system

```

// Exercise 11 of Lab 12
// To obtain Nyquist plot for given system

clc
close

s=poly(0,'s')
h=syslin('c',(5+s)*(s+40)/((s^3)*(s+200)*(s+1000)))
nyquist(h)
```

Scilab code Solution 12.12 To obtain Nyquist plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 12 of Lab 12
// To obtain Nyquist plot for given system

clc
close

s=poly(0,'s')
h=syslin('c',(5+s)/(s^2)*(s+2)*(s+10))
nyquist(h)
```

Experiment: 13

Bode Plots

Scilab code Solution 13.3 To obtain Bode plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 3 of Lab 13
// To obtain Bode plot of given system

clc
close

s=poly(0,'s')
h=syslin('c',31.62*(s*0.125+1)/((s)*(s*2+1)*(s
    *0.044+1)))
clf();
bode(h,0.1,100)
g_margin(h)
show_margins(h)
p_margin(h)
show_margins(h)
```

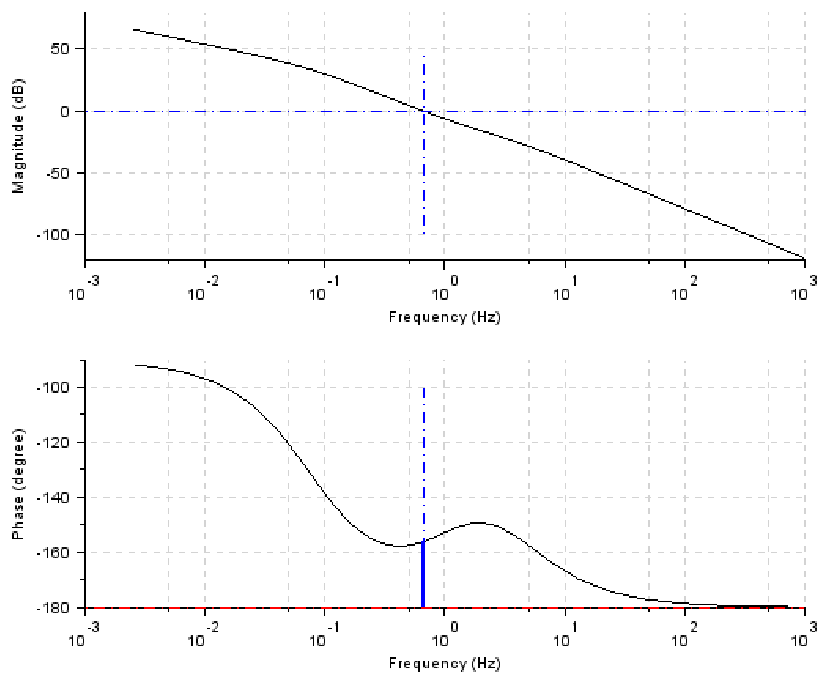

Figure 13.1: To obtain Bode plot of given system

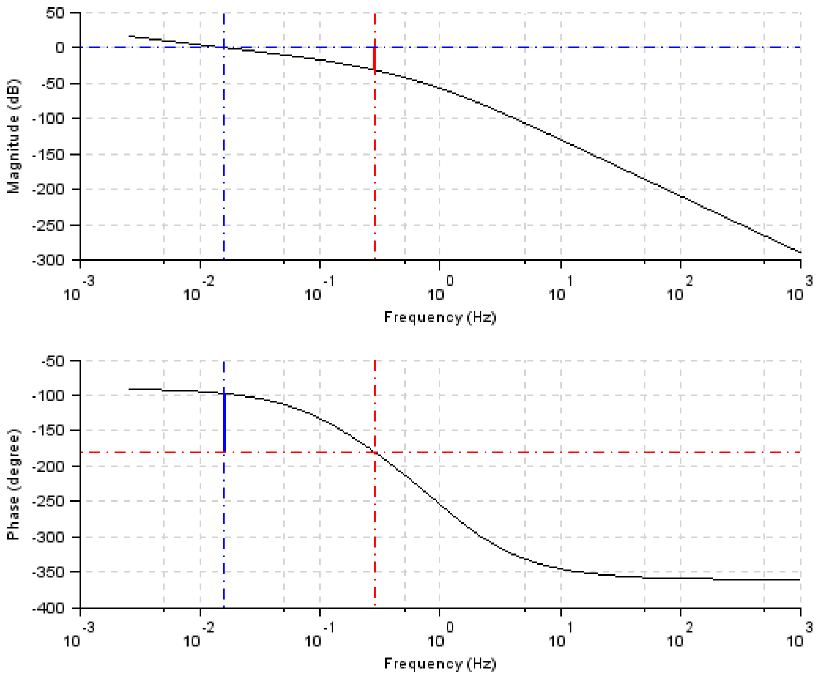

Figure 13.2: To obtain Bode plot of given system

Scilab code Solution 13.4 To obtain Bode plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 4 of Lab 13
// To obtain Bode plot of given system

clc
```

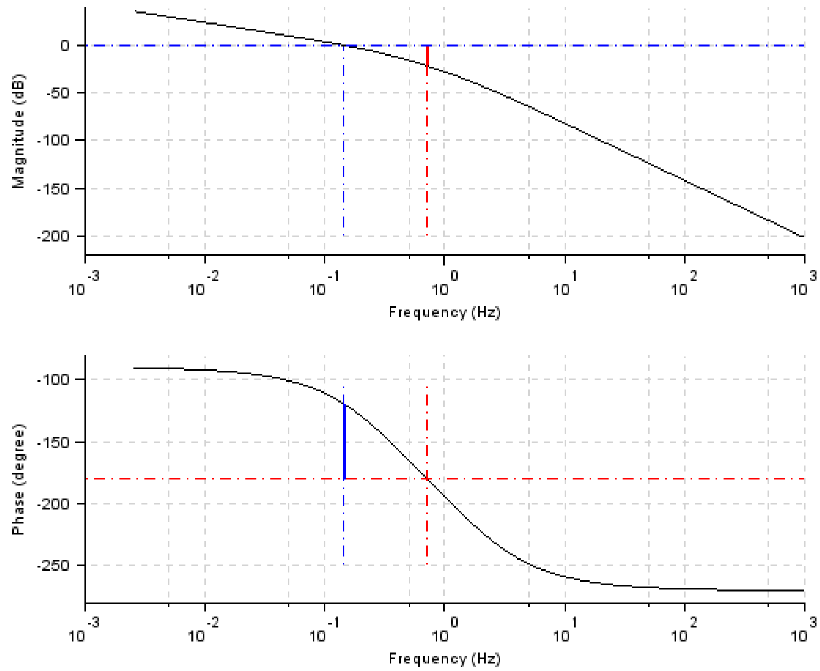

Figure 13.3: To obtain Bode plot of given system

```
close

s=poly(0,'s')
h=syslin('c',5/((s)*(s+10)*(s+5)*(s+1)))
clf();
bode(h,0.1,100)
g_margin(h)
show_margins(h)
p_margin(h)
show_margins(h)
```

Scilab code Solution 13.5 To obtain Bode plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 5 of Lab 13
// To obtain Bode plot of given system

clc
close

s=poly(0,'s')
h=syslin('c',1/((s)*(s*0.5+1)*(s*0.1+1)))
clf();
bode(h,0.1,100)
g_margin(h)
show_margins(h)
p_margin(h)
show_margins(h)
```

Scilab code Solution 13.6 To obtain Bode plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 6 of Lab 13
// To obtain Bode plot of given system

clc
close

```

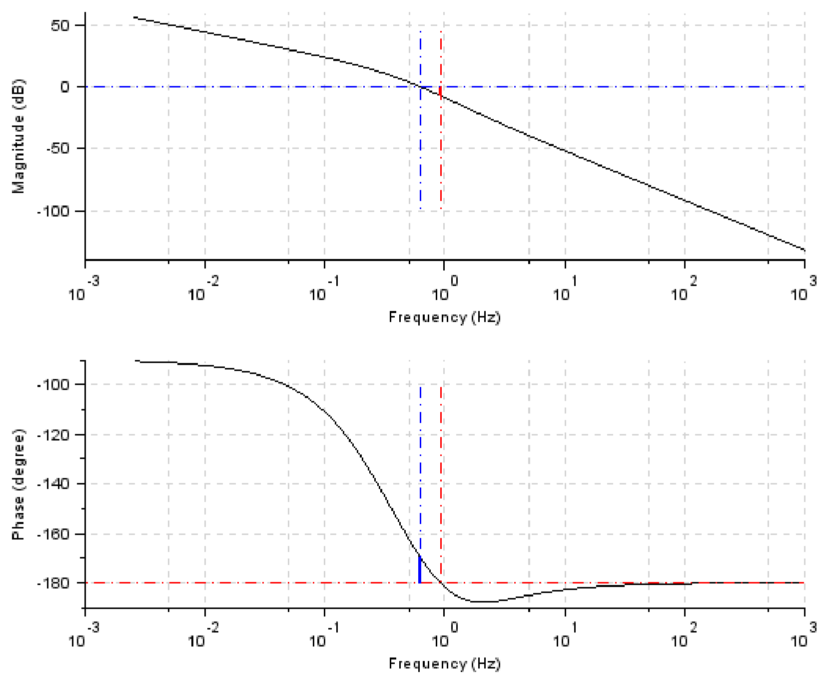

Figure 13.4: To obtain Bode plot of given system

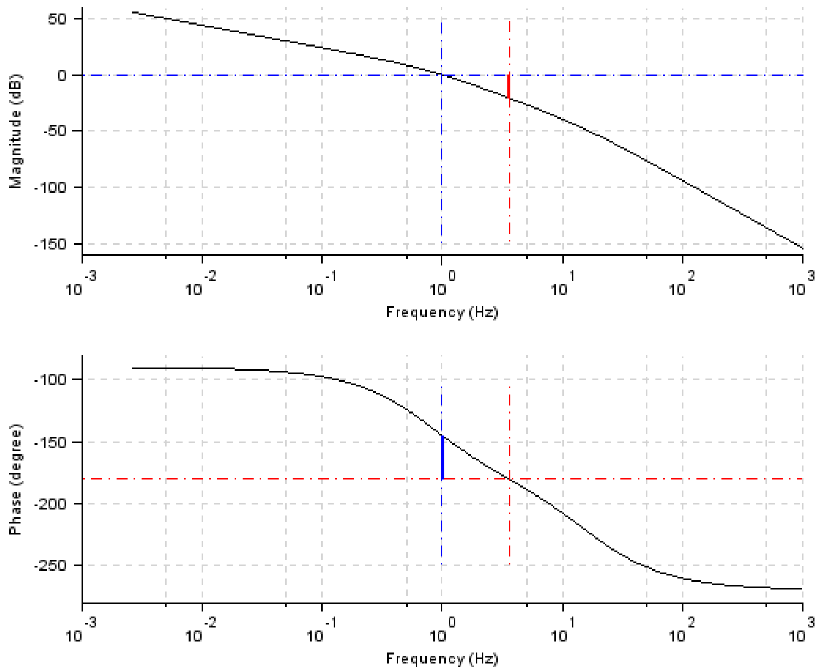

Figure 13.5: To obtain Bode plot of given system

```
s=poly(0,'s')
h=syslin('c',10*(s+10)/((s)*(s+2)*(s+5)))
clf();
bode(h,0.1,100)
g_margin(h)
show_margins(h)
p_margin(h)
show_margins(h)
```

Scilab code Solution 13.7 To obtain Bode plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 7 of Lab 13
// To obtain Bode plot of given system

clc
close

s=poly(0,'s')
h=syslin('c',10/((s)*(s*0.2+1)*(s*0.01+1)))
clf();
bode(h,0.1,100)
g_margin(h)
show_margins(h)
p_margin(h)
show_margins(h)
```

Scilab code Solution 13.8 To obtain Bode plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 8 of Lab 13
// To obtain Bode plot of given system

clc
close

s=poly(0,'s')
h=syslin('c',10/((s)*(s*0.01+1)*(s*0.1+1)))
```

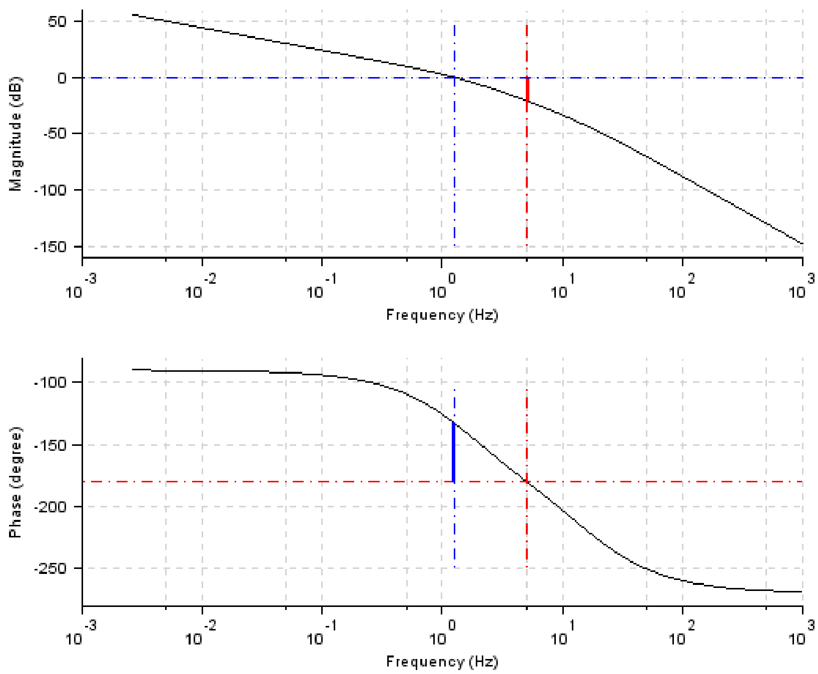

Figure 13.6: To obtain Bode plot of given system

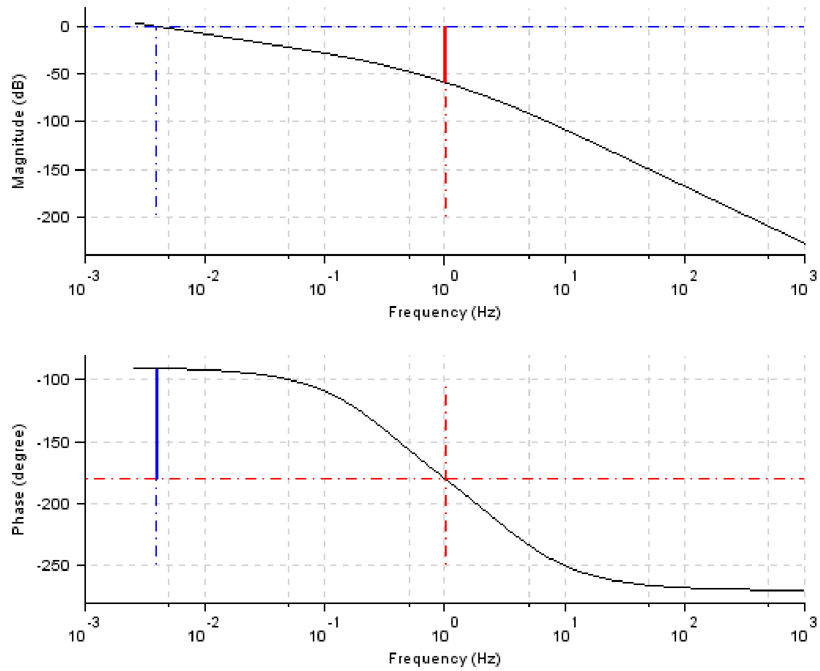

Figure 13.7: To obtain Bode plot of given system

```
clf();
bode(h,0.1,100)
g_margin(h)
show_margins(h)
p_margin(h)
show_margins(h)
```

Scilab code Solution 13.9 To obtain Bode plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
```

```
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 9 of Lab 13
// To obtain Bode plot of given system

clc
close

s=poly(0,'s')
h=syslin('c',0.025/((s)*(s*0.05+1)*(s*0.5+1)))
clf();
bode(h,0.1,100)
g_margin(h)
show_margins(h)
p_margin(h)
show_margins(h)
```

Scilab code Solution 13.10 To obtain Bode plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 10 of Lab 13
// To obtain Bode plot of given system

clc
close

s=poly(0,'s')
h=syslin('c',50/((s)*(s+1)*(s*0.5+1)))
clf();
bode(h,0.1,100)
```

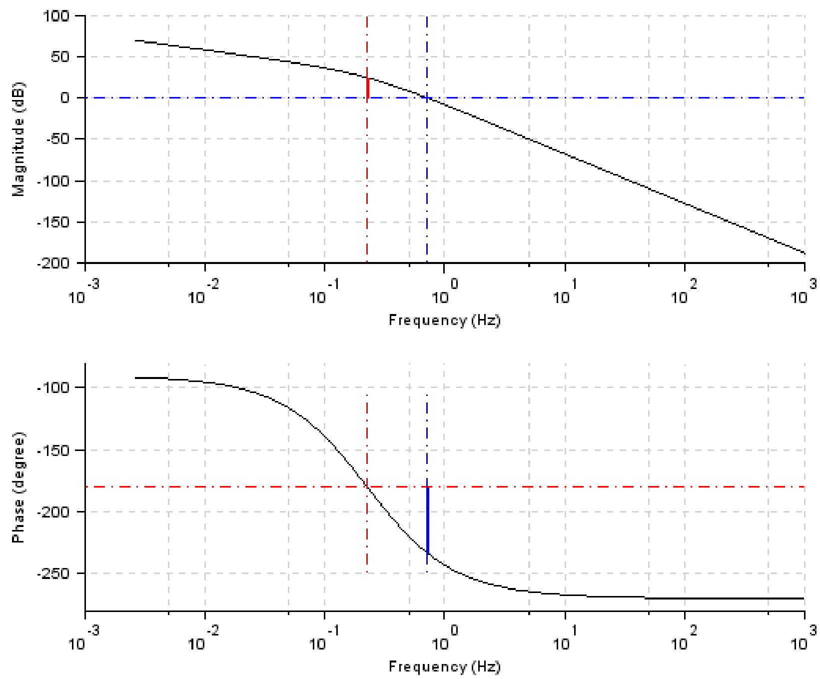

Figure 13.8: To obtain Bode plot of given system

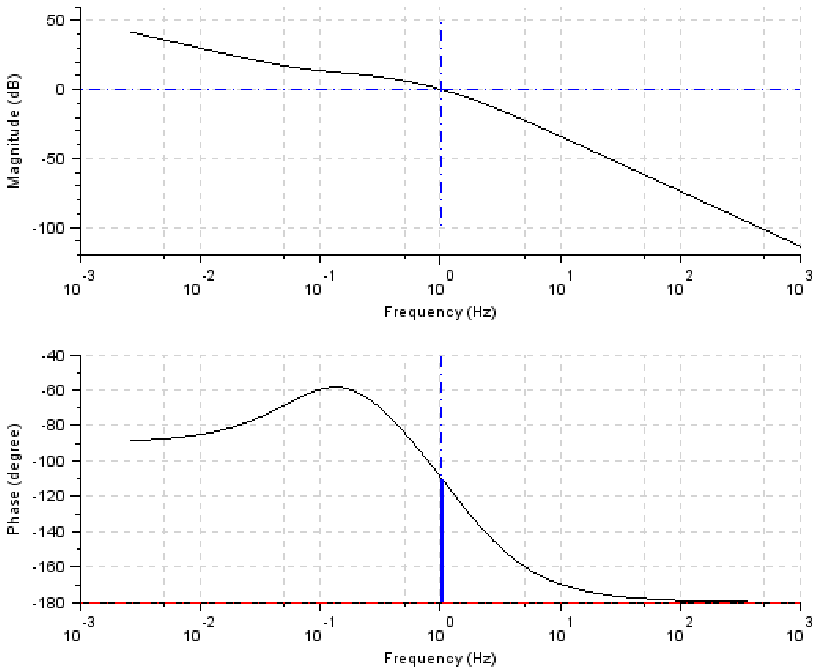

Figure 13.9: To obtain Bode plot of given system

```
g_margin(h)
show_margins(h)
p_margin(h)
show_margins(h)
```

Scilab code Solution 13.11 To obtain Bode plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

```

```
// Exercise 11 of Lab 13
// To obtain Bode plot of given system

clc
close

s=poly(0,'s')
h=syslin('c',(2+4*s)/((s)*(s*0.5+1)*(s*0.1+1)))
clf();
bode(h,0.1,100)
g_margin(h)
show_margins(h)
p_margin(h)
show_margins(h)
```

Scilab code Solution 13.12 To obtain Bode plot of given system

```
// OS : Windows 7
// Scilab : 5.4.1
// Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

// Exercise 12 of Lab 13
// To obtain Bode plot of given system

clc
close

s=poly(0,'s')
h=syslin('c',1/s^3)
clf();
bode(h,0.1,100)
g_margin(h)
show_margins(h)
```

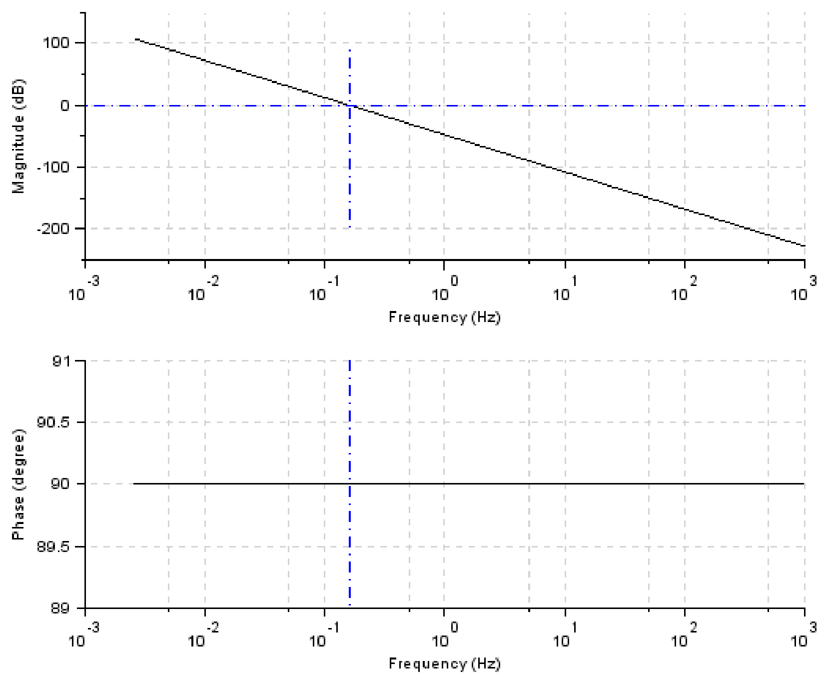

Figure 13.10: To obtain Bode plot of given system

```
p_margin(h)
show_margins(h)
```