

Scilab Manual for  
Control Theory  
by Prof Shweta Gaur  
Instrumentation Engineering  
Dharmsinh Desai University<sup>1</sup>

Solutions provided by  
Prof Shweta Gaur  
Instrumentation Engineering  
Dharmsinh Desai University

March 25, 2026

<sup>1</sup>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 Introduction to Control Systems	8
2 Transfer Function	12
3 Mathematical Modelling	18
4 Signal Flow Graph	20
5 Block Diagram Reduction Techniques	28
6 Feedback Characteristics	31
7 Time Response Analysis	37
8 Stability of Control Systems	43
9 Root Locus	49
10 Frequency Response Analysis	64
11 Polar Plots	69
12 Nyquist Plots	82
13 Bode Plots	97

# List of Experiments

Solution 1.5	To find Laplace Transform of given function . . . .	8
Solution 1.6	To find Inverse Laplace Transform of given function	9
Solution 1.7	To find Response of given system with respect to Step input . . . . .	9
Solution 1.8	To find Response of given system with respect to Ramp input . . . . .	10
Solution 2.3	To find Transfer function of given system . . . . .	12
Solution 2.4	To find Transfer function of given system . . . . .	13
Solution 2.5	To find Transfer function of given system . . . . .	14
Solution 2.6	To find Transfer function of given system . . . . .	15
Solution 2.7	To find Transfer function of given system . . . . .	16
Solution 3.4	To find Reflected Inertia and Reflected Coulomb Friction for given gear train system . . . . .	18
Solution 3.5	To find Transfer function of Mass spring system .	19
Solution 4.7	To obtain Transfer function using Masons gain for- mula . . . . .	20
Solution 4.8	To obtain Transfer function using Masons gain for- mula . . . . .	21
Solution 4.9	To obtain Transfer function using Masons gain for- mula . . . . .	22
Solution 4.10	To obtain Transfer function using Masons gain for- mula . . . . .	23
Solution 4.11	To obtain Transfer function using Masons gain for- mula . . . . .	24
Solution 4.12	To obtain Transfer function using Masons gain for- mula . . . . .	25
Solution 4.13	To obtain Transfer function using Masons gain for- mula . . . . .	26

Solution 4.14	To obtain Transfer function using Masons gain formula . . . . .	27
Solution 5.7	To solve Cascade of two functions . . . . .	28
Solution 5.8	To solve Parallel combination of two functions . . . . .	29
Solution 5.9	To find Transfer function for a closed loop . . . . .	29
Solution 6.6	To find sensitivity of given system . . . . .	31
Solution 6.7	To find sensitivity of given system . . . . .	32
Solution 6.8	To find sensitivity of given system . . . . .	32
Solution 6.9	To find sensitivity of given system . . . . .	34
Solution 6.10	To find sensitivity of given system . . . . .	34
Solution 6.11	To find sensitivity of given system . . . . .	35
Solution 7.6	To find Time response parameters for the given system . . . . .	37
Solution 7.7	To find Time response parameters for the given system . . . . .	38
Solution 7.8	To find Time response parameters for the given system . . . . .	39
Solution 7.9	To find Time response parameters for the given system . . . . .	41
Solution 8.6	To find stability of given system using Routh Hurwitz criteria . . . . .	43
Solution 8.7	To find stability of given system using Routh Hurwitz criteria . . . . .	44
Solution 8.8	To find value of K for marginal stability of given system . . . . .	45
Solution 8.9	To find value of K for marginal stability of given system . . . . .	46
Solution 8.10	To find value of K for marginal stability of given system . . . . .	47
Solution 9.3	To sketch Root locus of given system . . . . .	49
Solution 9.4	To sketch Root locus of given system . . . . .	52
Solution 9.5	To sketch Root locus of given system . . . . .	52
Solution 9.6	To sketch Root locus of given system . . . . .	54
Solution 9.7	To sketch Root locus of given system . . . . .	56
Solution 9.8	To sketch Root locus of given system . . . . .	56
Solution 9.9	To sketch Root locus of given system . . . . .	58
Solution 9.10	To sketch Root locus of given system . . . . .	60
Solution 9.11	To sketch Root locus of given system . . . . .	60

Solution 9.12	To sketch Root locus of given system . . . . .	62
Solution 10.5	To find Frequency domain specifications for the given system . . . . .	64
Solution 10.6	To find Frequency domain specifications for the given system . . . . .	65
Solution 10.7	To find Frequency domain specifications for the given system . . . . .	66
Solution 10.8	To find Frequency domain specifications for the given system . . . . .	67
Solution 11.3	To obtain Polar plot of given system . . . . .	69
Solution 11.4	To obtain Polar plot of given system . . . . .	72
Solution 11.5	To obtain Polar plot of given system . . . . .	72
Solution 11.6	To obtain Polar plot of given system . . . . .	74
Solution 11.7	To obtain Polar plot of given system . . . . .	75
Solution 11.8	To obtain Polar plot of given system . . . . .	76
Solution 11.9	To obtain Polar plot of given system . . . . .	78
Solution 11.10	To obtain Polar plot of given system . . . . .	79
Solution 11.11	To obtain Polar plot of given system . . . . .	80
Solution 12.3	To obtain Nyquist plot of given system . . . . .	82
Solution 12.4	To obtain Nyquist plot of given system . . . . .	82
Solution 12.5	To obtain Nyquist plot of given system . . . . .	85
Solution 12.6	To obtain Nyquist plot of given system . . . . .	88
Solution 12.7	To obtain Nyquist plot of given system . . . . .	88
Solution 12.8	To obtain Nyquist plot of given system . . . . .	90
Solution 12.9	To obtain Nyquist plot of given system . . . . .	92
Solution 12.10	To obtain Nyquist plot of given system . . . . .	92
Solution 12.11	To obtain Nyquist plot of given system . . . . .	94
Solution 12.12	To obtain Nyquist plot of given system . . . . .	95
Solution 13.3	To obtain Bode plot of given system . . . . .	97
Solution 13.4	To obtain Bode plot of given system . . . . .	99
Solution 13.5	To obtain Bode plot of given system . . . . .	100
Solution 13.6	To obtain Bode plot of given system . . . . .	101
Solution 13.7	To obtain Bode plot of given system . . . . .	103
Solution 13.8	To obtain Bode plot of given system . . . . .	104
Solution 13.9	To obtain Bode plot of given system . . . . .	106
Solution 13.10	To obtain Bode plot of given system . . . . .	107
Solution 13.11	To obtain Bode plot of given system . . . . .	109
Solution 13.12	To obtain Bode plot of given system . . . . .	110

# List of Figures

9.1	To sketch Root locus of given system . . . . .	50
9.2	To sketch Root locus of given system . . . . .	51
9.3	To sketch Root locus of given system . . . . .	53
9.4	To sketch Root locus of given system . . . . .	54
9.5	To sketch Root locus of given system . . . . .	55
9.6	To sketch Root locus of given system . . . . .	57
9.7	To sketch Root locus of given system . . . . .	58
9.8	To sketch Root locus of given system . . . . .	59
9.9	To sketch Root locus of given system . . . . .	61
9.10	To sketch Root locus of given system . . . . .	62
11.1	To obtain Polar plot of given system . . . . .	70
11.2	To obtain Polar plot of given system . . . . .	71
11.3	To obtain Polar plot of given system . . . . .	73
11.4	To obtain Polar plot of given system . . . . .	74
11.5	To obtain Polar plot of given system . . . . .	75
11.6	To obtain Polar plot of given system . . . . .	77
11.7	To obtain Polar plot of given system . . . . .	78
11.8	To obtain Polar plot of given system . . . . .	79
11.9	To obtain Polar plot of given system . . . . .	81
12.1	To obtain Nyquist plot of given system . . . . .	83
12.2	To obtain Nyquist plot of given system . . . . .	84
12.3	To obtain Nyquist plot of given system . . . . .	86
12.4	To obtain Nyquist plot of given system . . . . .	87
12.5	To obtain Nyquist plot of given system . . . . .	89
12.6	To obtain Nyquist plot of given system . . . . .	90
12.7	To obtain Nyquist plot of given system . . . . .	91
12.8	To obtain Nyquist plot of given system . . . . .	93

12.9 To obtain Nyquist plot of given system . . . . .	94
12.10 To obtain Nyquist plot of given system . . . . .	95
13.1 To obtain Bode plot of given system . . . . .	98
13.2 To obtain Bode plot of given system . . . . .	99
13.3 To obtain Bode plot of given system . . . . .	100
13.4 To obtain Bode plot of given system . . . . .	102
13.5 To obtain Bode plot of given system . . . . .	103
13.6 To obtain Bode plot of given system . . . . .	105
13.7 To obtain Bode plot of given system . . . . .	106
13.8 To obtain Bode plot of given system . . . . .	108
13.9 To obtain Bode plot of given system . . . . .	109
13.10 To obtain Bode plot of given system . . . . .	111

# Experiment: 1

## Introduction to Control Systems

Scilab code Solution 1.5 To find Laplace Transform of given function

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 5 of Lab 1
6 // To find Laplace Transform of given function f(t)
   = sin(2*t)*cos(3*t)
7
8 clc
9 close
10
11 syms t s
12 F=laplace (sin(2*t)*cos(2*t),t,s)
13 disp (F,"F( s )=")
```

---

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

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 6 of Lab 1
6 // To find Inverse Laplace Transform of given
   function F(s)=56*s/(s^4+106*s^2+2025)
7
8 clc
9 close
10
11 syms s t
12 F=56*s/(s^4+106*s^2+2025)
13 f=ilaplace(F,s,t)
14 disp (f," f ( t )=")
```

---

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

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 7 of Lab 1
```

```

6 // To find Response of given system w.r.t. Step
  input
7
8 clc
9 close
10
11 syms t s
12
13 TF=5/(s^2+6*s+101)
14 R=1/s
15
16 C=R*TF
17
18 disp(C,"Response = ")

```

---

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

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 8 of Lab 1
6 // To find Response of given system w.r.t. Ramp
  input
7
8 clc
9 close
10
11 syms t s
12
13 TF=10/((s+2)*(s+3))
14 R=1/s^2

```

```
15
16 C=R*TF
17
18 disp(C," Response = ")
```

---

# Experiment: 2

## Transfer Function

Scilab code **Solution 2.3** To find Transfer function of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 3 of Lab 2
6 // To find Transfer function of given system
7
8 clc
9 close
10
11 syms s
12
13 z1=-2
14 p1=-1
15 p2=0
16 p3=-3
17
18 tf=(s-z1)/((s-p1)*(s-p2)*(s-p3))
19 disp(tf,"Transfer function = ")
```

---

**Scilab code Solution 2.4** To find Transfer function of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 4 of Lab 2
6 // To find Transfer function of given system
7
8 clc
9 close
10
11 syms s R1 R2 C1 C2
12
13 //Preparing SFG for given circuit
14 //i1=(ei-V1)/R1
15 //V1=(i1-i2)*(1/(C1*s))
16 //i2=(V1-e0)/R2
17 //e0=i2*(1/(C2*s))
18
19 //Using Masons gain formula
20 P1=(1/R1)*(1/(C1*s))*(1/R2)*(1/(C2*s))
21 L1=(1/(C1*s))*(-1/R1)
22 L2=(1/R2)*(-1/(C1*s))
23 L3=(1/(C2*s))*(-1/R2)
24 L11=L1*L3
25 del1=1
26 del=1-(L1+L2+L3)+L11
27 tf=P1*del1/del
28 disp(tf,"Transfer function = ")
```

---

**Scilab code Solution 2.5** To find Transfer function of given system

```
1
2
3
4
5
6 // OS : Windows 7
7 // Scilab : 5.4.1
8 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
9
10 // Exercise 5 of Lab 2
11 // To find Transfer function of given system
12
13 clc
14 close
15
16 syms s R1 R2 C
17
18 //Preparing SFG for given circuit
19 //i1=(ei-V1)/R1
20 //ei=R1*i1+R2*i1+(1/(C*s))*i1
21 //V1=i1*(R2+(1/(C*s)))
22 //e2=V1
23
24 //Using Masons gain formula
25 P1=(1/R1)*(R2+(1/(C*s)))*1
26 L1=(R2+(1/(C*s)))*(-1/R1)
27 del=1-L1
28 del1=1
29
30 tf=P1*del1/del
```

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

---

**Scilab code Solution 2.6** To find Transfer function of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 6 of Lab 2
6 // To find Transfer function of given system
7
8 clc
9 close
10
11 syms s R1 R2 R3 R4 L1 L2 L3
12
13 //Preparing SFG for given circuit
14 //i1=(ei-V1)/(R1+L1*s)
15 //V1=(i1-i2)*(R2+L2*s)
16 //i2=(V1-e0)/(R3+L3*s)
17 //e0=R4*i2
18
19 //Using Masons gain formula
20 P1=R4*(1/(R1+L1*s))*(R2+L2*s)*(1/(R3+L3*s))
21 L1=-((R2+L2*s)/(R1+L1*s))
22 L2=-((R2+L2*s)/(R3+L3*s))
23 L3=-R4/(R3+L3*s)
24 L11=L1*L3
25 del1=1
26 del=1-(L1+L2+L3)+L11
27
28 tf=P1*del1/del
29 disp(tf," Transfer function = ")
```

---

**Scilab code Solution 2.7** To find Transfer function of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 7 of Lab 2
6 // To find Transfer function of given system
7
8 clc
9 close
10
11 syms s R1 R2 Rb C1 C2 L1 L2
12
13 //Preparing SFG for given circuit
14 //i1=(ei-V1)/Rb
15 //V1=(i1-i2)*(1/(C1*s))+V2
16 //i2=(V1-e0)/(R1+L1*s)
17 //V2=i1*(R2+L2*s)
18 //e0-V2=i2/(C2*s)
19
20 //Using Masons gain formula
21 P1=(1/Rb)*(1/(C1*s))*(1/(R1+L1*s))*(1/(C2*s))
22 P2=(1/Rb)*(R2+L2*s)*1
23 L1=(1/(C1*s))*(-1/Rb)
24 L2=(1/(R1+L1*s))*(-1/(C1*s))
25 L3=(1/(C2*s))*(-1/(R1+L1*s))
26 L4=(R2+L2*s)*1*(-1/Rb)
27 L5=(R2+L2*s)*1*(-1/(R1*L1*s))*(-1/(C1*s))*(-1/Rb)
28 L11=L1*L3
29
30 del1=1
```

```
31 del2=1-L2
32 del=1-(L1+L2+L3+L4+L5)+L11
33 tf=(P1*del1+P2*del2)/del
34 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

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 4 of Lab 3
6 // To find Reflected Inertia and Reflected Coulomb
   Friction for given gear train system
7
8 clc
9 close
10
11 J2=0.08;
12 T2=5;
13 teeth_ratio=1/10
14 J1=(teeth_ratio)^2*J2;
15 T1=(teeth_ratio)*T2
16
17 printf(" Reflected Inertia = %f oz-in.-sec3\n",J1)
```

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

---

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

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 5 of Lab 3
6 // To find Transfer function of Mass spring system
7
8 clc
9 close
10
11 syms K M B
12 s=%s
13 P1=(1/M)*(s^-2)
14 L11=-(B/M)*(s^-1)
15 L21=-(K/M)*(s^-2)
16 del=1-(L11+L21)
17 del1=1
18 tf=P1*del1/del
19 disp(tf," Transfer function = ")
```

---

# Experiment: 4

## Signal Flow Graph

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

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 7 of Lab 4
6 // To obtain Transfer function using Mason's gain
   formula
7
8 clc
9 close
10
11 syms g1 g2 g3 g4 h1 h2
12
13 p1=g1*g2*g3
14 p2=g4
15 l1=-g2*h1
16 l2=g1*g2*h1
17 l3=-(g3*g2*h2)
```

```

18 d=1-(l1+l2+l3)
19 d1=1
20 d2=d
21 tf=(p1*d1+p2*d2)/(d)
22
23 disp(tf,"Transfer funtion = ")

```

---

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

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 8 of Lab 4
6 // To obtain Transfer function using Mason's gain
   formula
7
8 clc
9 close
10
11 syms g1 g2 g3 g4 h1 h2
12
13 p1=g1*g3
14 p2=g1*g2
15 p3=g1*g3*g4*h2
16 p4=g1*g2*g4*h2
17 d1=1
18 d2=1
19 d3=1
20 d4=1
21 l1=-g1*g3*h1*h2
22 l2=-g1*g2*h1*h2

```

```

23 d=1-(l1+l2)
24 tf=(p1*d1+p2*d2+p3*d3+p4*d4)/d
25
26 disp(tf,"Transfer function = ")

```

---

**Scilab code Solution 4.9** To obtain Transfer function using Mason's gain formula

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 9 of Lab 4
6 // To obtain Transfer function using Mason's gain
  formula
7
8 clc
9 close
10
11 syms g1 g2 g3 g4 g5 g6 g7 g8 h1 h2
12
13 p1=g2*g4*g6
14 p2=g3*g5*g7
15 p3=g2*g1*g7
16 p4=g3*g6*g8
17 p5=-g2*g1*h2*g8*g6
18 p6=-g3*g8*h1*g1*g7
19 l1=-g4*h1
20 l2=-g5*h2
21 l3=g1*h2*g8*h1
22 d=1+g4*h1+g5*h2-g1*h2*g8*h1+g4*h1*g5*h2
23 d1=1+g5*h2
24 d2=1+g4*h1

```

```

25 d3=1
26 d4=1
27 d5=1
28 d6=1
29 tf=(p1*d1+p2*d2+p3*d3+p4*d4+p5*d5+p6*d6)/d
30
31 disp(tf,"Transfer function = ")

```

---

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

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 10 of Lab 4
6 // To obtain Transfer function using Mason's gain
  formula
7
8 clc
9 close
10
11 syms t1 t2 t3 q1 q2 s
12 p1=t1/(s*(s+q1))
13 p2=t2/(s+q1)
14 p3=t3
15 l1=q2/(s*(s+q1))
16 d=1+(q2/(s*(s+q1)))
17 d1=1
18 d2=1
19 d3=1+(q2/(s*(s+q1)))
20 tf=(p1*d1+p2*d2+p3*d3)/d
21

```

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

---

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

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 11 of Lab 4
6 // To obtain Transfer function using Mason's gain
   formula
7
8 clc
9 close
10
11 syms g1 g2 g3 g4 h1 h2
12
13 p1=g1*g2*g3
14 p2=g1*g4
15 l1=-g1*g2*h1
16 l2=-g2*g3*h2
17 l3=-g1*g2*g3
18 l4=-g4*h2
19 l5=-g4*g1
20 d=1+g1*g2*h1+g2*g3*h2+g1*g2*g3+g4*h2+g1*g4
21 d1=1
22 d2=1
23 tf=(p1*d1+p2*d2)/(d)
24
25 disp(tf,"Transfer function = ")
```

---

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

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 12 of Lab 4
6 // To obtain Transfer function using Mason's gain
   formula
7
8 clc
9 close
10
11 syms g1 g2 g3 g4 h1 h2
12
13 p1=g1*g2*g3
14 p2=g4
15 l1=-g1*g2*h1
16 l2=-g1*g2*g3
17 l3=-g2*g3*h2
18 l4-g4
19 l5=g4*h2*g2*h1
20 d=1-(l1+l2+l3+l4+l5)
21 d1=1
22 d2=1
23 tf=(p1*d1+p2*d2)/d
24
25 disp(tf," Transfer function = ")
```

---

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

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 13 of Lab 4
6 // To obtain Transfer function using Mason's gain
   formula
7
8 clc
9 close
10
11 syms g1 g2 g3 g4 h1 h2 h3 h4
12
13 p1=g1*g2*g3
14 p2=g4*g3
15 l1=g3*g4*h1*h2
16 l2=g1*g2*g3*h1*h2
17 l3=-g1*h2*h3
18 d=1+g1*h2*h3-g3*g4*h1*h2-g1*g2*g3*h1*h2
19 d1=1
20 d2=1
21 tf=(p1*d1+p2*d2)/d
22
23 disp(tf,"Transfer function = ")
```

---

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

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 14 of Lab 4
6 // To obtain Transfer function using Mason's gain
   formula
7
8 clc
9 close
10
11 syms g1 g2 g3 g4 g5 g6 g7 g8 g9
12
13 p1=g1*g2
14 p2=g4
15 p3=g7*g8
16 p4=g1*g5*g8
17 p5=g7*g6*g2
18 l1=g9
19 l2=g3
20 l3=g5*g6
21 d=1-(g3+g9+g5*g6)+g9*g3
22 d1=1-g9
23 d2=1-(g9+g3+g5*g6)+g9*g3
24 d3=1-g3
25 d4=1
26 d5=1
27 tf=(p1*d1+p2*d2+p3*d3+p4*d4+p5*d5)/d
28
29 disp(tf," Transfer function = ")
```

---

# Experiment: 5

## Block Diagram Reduction Techniques

Scilab code Solution 5.7 To solve Cascade of two functions

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 7 of Lab 1
6 // To solve Cascade of two functions
7 clc
8 close
9
10 syms s
11
12  $G1=5/(s*(s^2+s+2))$ 
13  $G2=(5*s+3)/((s+1)*(s+2))$ 
14  $G=G1*G2$ 
15
16 disp(G,"Resultant function G(s) = ")
```

---

**Scilab code Solution 5.8** To solve Parallel combination of two functions

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 8 of Lab 1
6 // To solve Parallel combination of two functions
7 clc
8 close
9
10 syms s
11
12 G1=19/((s+5)*(s+99))
13 G2=(10*s+9)/((s+3)*(s+5))
14 G=G1/G2
15
16 disp(G,"Resultant function G(s) = ")
```

---

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

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 9 of Lab 1
6 // To find Transfer function for a closed loop
7 clc
```

```
8 close
9
10 syms s
11
12 G=10/(s*(s+20))
13 H=2
14 TF=G/(1+G*H)
15
16 disp(TF,"Transfer function = ")
```

---

# Experiment: 6

## Feedback Characteristics

Scilab code Solution 6.6 To find sensitivity of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 6 of Lab 6
6 // To find sensitivity of given system
7
8 clc
9 close
10
11 funcprot(0);
12 s=%i;
13 h=.25;
14 g=25/(s2+s)
15 sg_complex=1/(1+g*h)
16 sg=abs(sg_complex)
17 disp(sg,"Sensitivity with respect to G=")
18 sh_complex=(-g*h/(1+g*h))
19 sh=abs(sh_complex)
```

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

---

**Scilab code Solution 6.7** To find sensitivity of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 7 of Lab 6
6 // To find sensitivity of given system
7
8 clc
9 close
10
11 funcprot(0);
12 s=%i*2;
13 h=1;
14 g=5/(s^2+s*2)
15 sg_complex=1/(1+g*h)
16 sg=abs(sg_complex)
17 disp(sg," Sensitivity with respect to G=")
18 sh_complex=(-g*h/(1+g*h))
19 sh=abs(sh_complex)
20 disp(sh," Sensitivity with respect to H=")
```

---

**Scilab code Solution 6.8** To find sensitivity of given system

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

```

3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 8 of Lab 6
6 // To find sensitivity of given system
7
8 clc
9 close
10
11 // FOR w=1 rad/sec;
12 funcprot(0);
13 s=1*%i;
14 h=10;
15 g=10/(s^2+s)
16 sg_complex=1/(1+g*h)
17 sg=abs(sg_complex)
18 disp("For w=1 rad/sec")
19 disp(sg, "Sensitivity with respect to G=")
20 sh_complex=(-g*h/(1+g*h))
21 sh=abs(sh_complex)
22 disp(sh, "Sensitivity with respect to H=")
23
24 // FOR w=2 rad/sec;
25 funcprot(0);
26 s=2*%i;
27 h=10;
28 g=10/(s^2+s)
29 sg_complex=1/(1+g*h)
30 sg=abs(sg_complex)
31 disp("For w=2 rad/sec")
32 disp(sg, "Sensitivity with respect to G=")
33 sh_complex=(-g*h/(1+g*h))
34 sh=abs(sh_complex)
35 disp(sh, "Sensitivity with respect to H=")

```

---

**Scilab code Solution 6.9** To find sensitivity of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 9 of Lab 6
6 // To find sensitivity of given system
7
8 clc
9 close
10
11 funcprot(0);
12 s=1*%i;
13 h=1;
14 k=1;
15 g=k/(s^2+15*s)
16 deff('t=f(k,s)', 't=k/(s^2+15*s+k)');
17 sg_complex=(derivative(f,k))*(k/(k/(s^2+15*s+k)))
18 sg=abs(sg_complex)
19 disp(sg, "Sensitivity with respect to K=")
```

---

**Scilab code Solution 6.10** To find sensitivity of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 10 of Lab 6
6 // To find sensitivity of given system
7
8 clc
9 close
```

```

10
11 funcprot(0);
12 s=%i;
13 T=input("enter the value of T=") // we will ask
    user to input the value of 'T'
14 h=1;
15 k=10;
16 g=k/(T*s+1)
17 sg_complex=1/(1+g*h)
18 sg=abs(sg_complex)
19 disp(sg,"Sensitivity with respect to G=")

```

---

**Scilab code Solution 6.11** To find sensitivity of given system

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 11 of Lab 6
6 // To find sensitivity of given system
7
8 clc
9 close
10
11 funcprot(0);
12 s=.5*i;
13 h=1;
14 k=1;
15 g=25*k/(s^2+5*s)
16 sg_complex=1/(1+g*h)
17 sg=abs(sg_complex)
18
19 disp(sg,"Sensitivity with respect to G=")

```

```
20 sh_complex=(-g*h/(1+g*h))
21 sh=abs(sh_complex)
22 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

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 6 of Lab 7
6 // To find Time response parameters for the given
   system
7
8 clc
9 close
10
11 s=poly(0, 's');
12 g=100/(s^2+10*s)
13 tf=g/(1+g)
14
15 // To compare tf with  $W_n^2/(s^2+2*\zeta*W_n+W_n^2)$ 
16 y=denom(tf)
17 q=numer(tf)
```

```

18 z=coeff(y)
19 r=coeff(q)
20 wn=sqrt(r)
21
22 zeta=z(1,2)/(2*wn)
23 wd=(wn*sqrt(1-zeta^2))
24 mp=((%e^(-%pi*zeta/(sqrt(1-zeta^2))))*100)
25 theta=atan(sqrt(1-zeta^2)/zeta);
26 tr=(%pi-theta)/wd)
27 tp=(%pi/wd)
28 ts=(4/(zeta*wn))
29
30 printf("Undamped natural frequency = %f rad/sec \n",
        wn)
31 printf("Damping ratio = %f \n",zeta)
32 printf("Damped frequency = %f rad/sec \n",wd)
33 printf("Maximum Peak Overshoot = %f percent \n",mp)
34 printf("Theta = %f rad \n",theta)
35 printf("Rise time = %f sec \n",tr)
36 printf("Peak Time = %f sec \n",tp)
37 printf("Settling time = %f sec \n",ts)

```

---

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

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 7 of Lab 7
6 // To find Time response parameters for the given
  system
7

```

```

8  clc
9  close
10
11 s=poly(0, 's');
12 tf=2025/(s^2+45*s+2025)
13
14 // To compare tf with  $W_n^2/(s^2+2*\zeta*W_n+W_n^2)$ 
15 y=denom(tf)
16 q=numer(tf)
17 z=coeff(y)
18 r=coeff(q)
19 wn=sqrt(r)
20
21 zeta=z(1,2)/(2*wn)
22 wd=(wn*sqrt(1-zeta^2))
23 mp=((%e^(-%pi*zeta/(sqrt(1-zeta^2)))))*100)
24 theta=atan(sqrt(1-zeta^2)/zeta);
25 tr=((%pi-theta)/wd)
26 tp=(%pi/wd)
27 ts=(4/(zeta*wn))
28
29 printf("Undamped natural frequency = %f rad/sec \n",
        wn)
30 printf("Damping ratio = %f \n",zeta)
31 printf("Damped frequency = %f rad/sec \n",wd)
32 printf("Maximum Peak Overshoot = %f percent \n",mp)
33 printf("Theta = %f rad \n",theta)
34 printf("Rise time = %f sec \n",tr)
35 printf("Peak Time = %f sec \n",tp)
36 printf("Settling time = %f sec \n",ts)

```

---

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

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 8 of Lab 7
6 // To find Time response parameters for the given
   system
7
8 clc
9 close
10
11 s=poly(0, 's');
12 g=6.25/(s^2+2*s)
13 tf=g/(1+g)
14
15 // To compare tf with  $Wn^2/(s^2+2*\zeta*Wn+Wn^2)$ 
16 y=denom(tf)
17 q=numer(tf)
18 z=coeff(y)
19 r=coeff(q)
20 wn=sqrt(r)
21
22 zeta=z(1,2)/(2*wn)
23 wd=(wn*sqrt(1-zeta^2))
24 mp=((%e^(-%pi*zeta/(sqrt(1-zeta^2))))*100)
25 theta=atan(sqrt(1-zeta^2)/zeta);
26 tr=((%pi-theta)/wd)
27 tp=(%pi/wd)
28 ts=(4/(zeta*wn))
29
30 printf("Undamped natural frequency = %f rad/sec \n",
   wn)
31 printf("Damping ratio = %f \n",zeta)
32 printf("Damped frequency = %f rad/sec \n",wd)
33 printf("Maximum Peak Overshoot = %f percent \n",mp)
34 printf("Theta = %f rad \n",theta)
35 printf("Rise time = %f sec \n",tr)
36 printf("Peak Time = %f sec \n",tp)

```

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

---

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

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 9 of Lab 7
6 // To find Time response parameters for the given
   system
7
8 clc
9 close
10
11 s=poly(0, 's');
12 tf=36481/(s^2+191*s+36481)
13
14 // To compare tf with  $W_n^2/(s^2+2*\zeta*W_n+W_n^2)$ 
15 y=denom(tf)
16 q=numer(tf)
17 z=coeff(y)
18 r=coeff(q)
19 wn=sqrt(r)
20
21 zeta=z(1,2)/(2*wn)
22 wd=(wn*sqrt(1-zeta^2))
23 mp=((%e^(-%pi*zeta/(sqrt(1-zeta^2))))*100)
24 theta=atan(sqrt(1-zeta^2)/zeta);
25 tr=((%pi-theta)/wd)
26 tp=(%pi/wd)
27 ts=(4/(zeta*wn))
```

```
28
29 printf("Undamped natural frequency = %f rad/sec \n",
        wn)
30 printf("Damping ratio = %f \n",zeta)
31 printf("Damped frequency = %f rad/sec \n",wd)
32 printf("Maximum Peak Overshoot = %f percent \n",mp)
33 printf("Theta = %f rad \n",theta)
34 printf("Rise time = %f sec \n",tr)
35 printf("Peak Time = %f sec \n",tp)
36 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

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 6 of Lab 8
6 // To find stability of given system using Routh
   Hurwitz criterion
7
8 clc
9 close
10
11 s=%s;
12 p=s^3-4*s^2+s+6;
13 disp(p,"Given system function = ")
14 c=coeff(p)
15 l=length(c)
16 routh=routh_t(p) //This Function generates the Routh
   table
```

```

17 disp(routh,"Rouths table = ")
18 x=0;
19 for i=1:l
20 if (routh(i,1)<0)
21 x=x+1;
22 end
23 end
24 if(x>=1)
25     printf("From Rouths table , it is clear that the
            system is unstable.")
26 else
27     printf("From Rouths table , it is clear that the
            system is stable.")
28 end

```

---

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

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 7 of Lab 8
6 // To find stability of given system using Routh
   Hurwitz criterion
7
8 clc
9 close
10
11 s=%s;
12 p=2*s^4+s^3+3*s^2+5*s+10;
13 disp(p,"Given system function = ")
14 c=coeff(p)

```

```

15 l=length(c)
16 routh=routh_t(p) //This Function generates the Routh
    table
17 disp(routh,"Rouths table = ")
18 x=0;
19 for i=1:l
20 if (routh(i,1)<0)
21 x=x+1;
22 end
23 end
24 if(x>=1)
25     printf("From Rouths table , it is clear that the
        system is unstable.")
26 else
27     printf("From Rouths table , it is clear that the
        system is stable.")
28 end

```

---

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

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 8 of Lab 8
6 // To find value of K for marginal stability of
    given system
7
8 clc
9 close
10
11 s=%s

```

```

12 syms K
13 p=s^3+2*s^2+(1-K)*s+K
14 coef_a0 = coeffs(p, 's',0);
15 coef_a1 = coeffs(p, 's',1);
16 coef_a2 = coeffs(p, 's',2);
17 coef_a3 = coeffs(p, 's',3);
18
19 c=[coef_a0 coef_a1 coef_a2 coef_a3]
20
21 l=length(c);
22 routh=[c([4,2]);c([3,1])];
23 routh=[routh;-det(routh)/routh(2,1),0];
24 a=routh(2:3,1:2); //Getting the square sub block of
    routh matrix
25 routh=[routh;-det(a)/a(2,1),0]
26 disp(routh,"Rouths table = ")
27 routh(3,1)=0 //For marginal stability
28 sys=syslin('c',(1-s)/(s^3+2*s^2+s))
29 k=kpure(sys)
30 disp(k,"K(marginal)=")

```

---

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

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 9 of Lab 8
6 // To find value of K for marginal stability of
    given system
7
8 clc

```

```

9  close
10
11 s=%s
12 syms K
13 p=s^3+2.5*s^2+20*s+10*K
14 coef_a0 = coeffs(p, 's',0);
15 coef_a1 = coeffs(p, 's',1);
16 coef_a2 = coeffs(p, 's',2);
17 coef_a3 = coeffs(p, 's',3);
18
19 c=[coef_a0 coef_a1 coef_a2 coef_a3]
20
21 l=length(c);
22 routh=[c([4,2]);c([3,1])];
23 routh=[routh;-det(routh)/routh(2,1),0];
24 a=routh(2:3,1:2); //Getting the square sub block of
    routh matrix
25 routh=[routh;-det(a)/a(2,1),0]
26 disp(routh,"Rouths table = ")
27 routh(3,1)=0 //For marginal stability
28 sys=symlin('c',10/(s^3+2.5*s^2+20*s))
29 k=kpure(sys)
30 disp(k,"K(marginal)=")

```

---

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

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 10 of Lab 8
6 // To find value of K for marginal stability of

```

```

        given system
7
8  clc
9  close
10
11 s=%s
12 syms K
13 p=s^3+10*s^2+(21+K)*s+13*K
14 coef_a0 = coeffs(p, 's',0);
15 coef_a1 = coeffs(p, 's',1);
16 coef_a2 = coeffs(p, 's',2);
17 coef_a3 = coeffs(p, 's',3);
18
19 c=[coef_a0 coef_a1 coef_a2 coef_a3]
20
21 l=length(c);
22 routh=[c([4,2]);c([3,1])];
23 routh=[routh;-det(routh)/routh(2,1),0];
24 a=routh(2:3,1:2); //Getting the square sub block of
        routh matrix
25 routh=[routh;-det(a)/a(2,1),0]
26 disp(routh,"Rouths table = ")
27 routh(3,1)=0 //For marginal stability
28 sys=syslin('c',(s+13)/(s^3+10*s^2+21*s))
29 k=kpure(sys)
30 disp(k,"K(marginal)=")

```

---

# Experiment: 9

## Root Locus

Scilab code **Solution 9.3** To sketch Root locus of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 3 of Lab 9
6 // To sketch Root locus of given system
7
8 clc
9 close
10 clf
11
12 s=%s
13 h=syslin('c',1/(s^3))
14 evans(h,100)
15 sgrid()
```

---

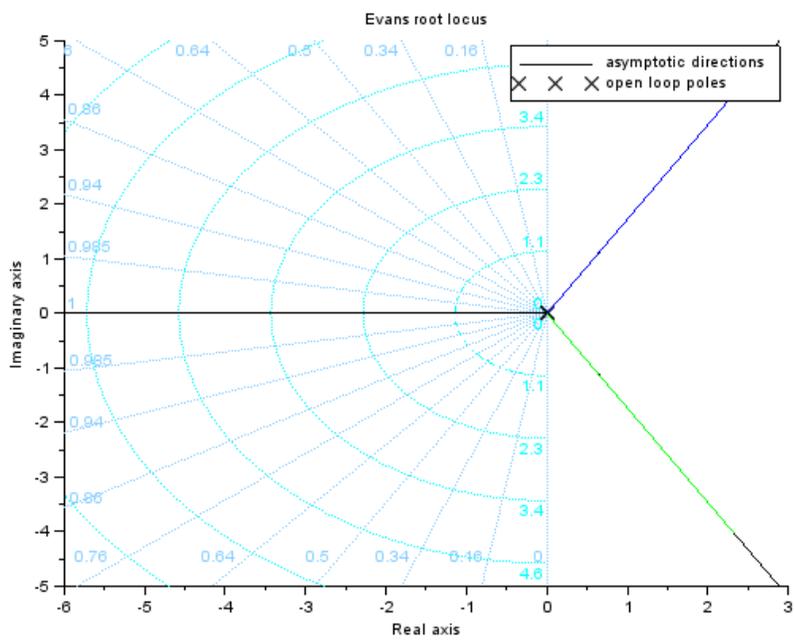


Figure 9.1: To sketch Root locus of given system

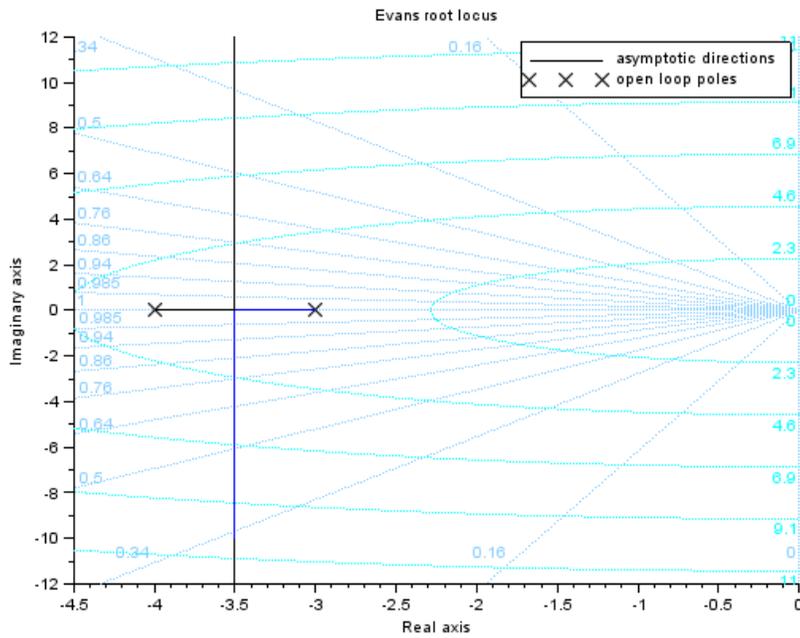


Figure 9.2: To sketch Root locus of given system

**Scilab code Solution 9.4** To sketch Root locus of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 4 of Lab 9
6 // To sketch Root locus of given system
7
8 clc
9 close
10 clf
11
12 s=%s
13 h=syslin('c',1/((s+4)*(s+3)))
14 evans(h,100)
15 sgrid()
```

---

**Scilab code Solution 9.5** To sketch Root locus of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 5 of Lab 9
6 // To sketch Root locus of given system
7
8 clc
9 close
10 clf
```

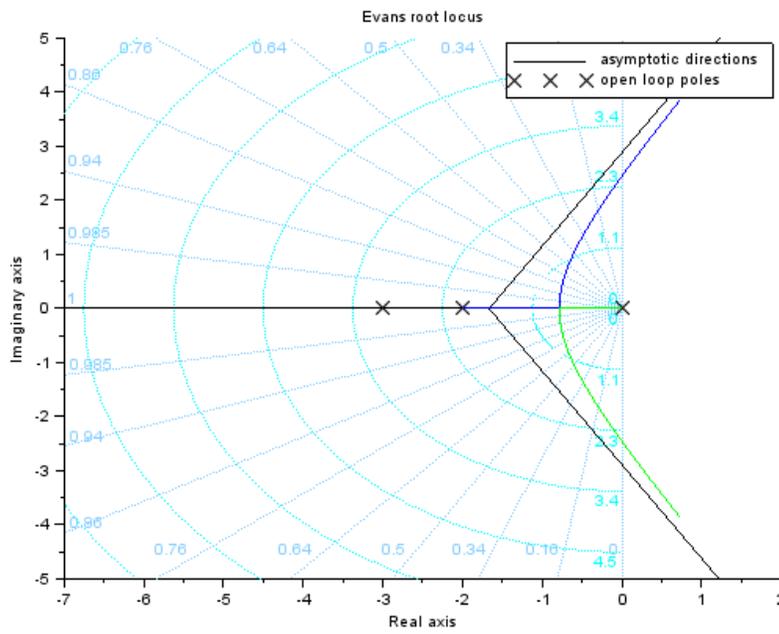


Figure 9.3: To sketch Root locus of given system

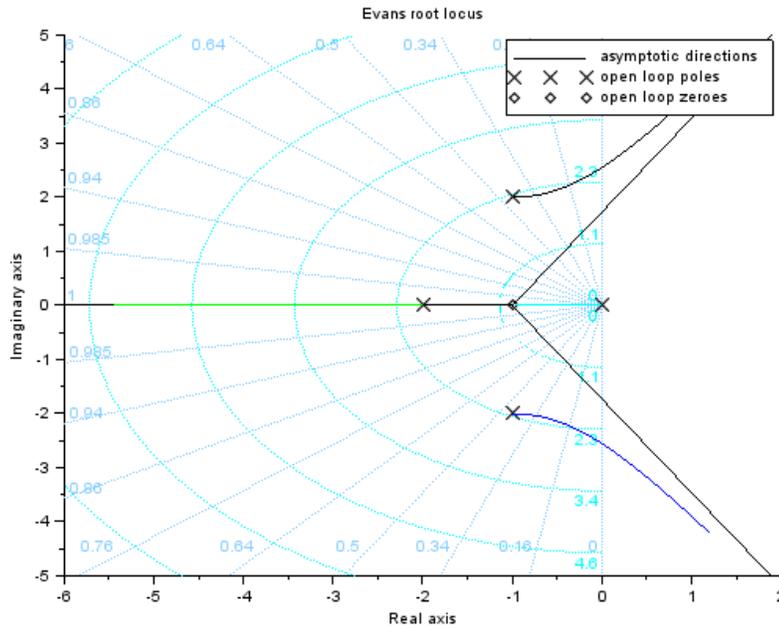


Figure 9.4: To sketch Root locus of given system

```

11
12 s=%s
13 h=syslin('c',1/((s)*(s+2)*(s+3)))
14 evans(h,100)
15 sgrid()

```

---

**Scilab code Solution 9.6** To sketch Root locus of given system

```

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

```

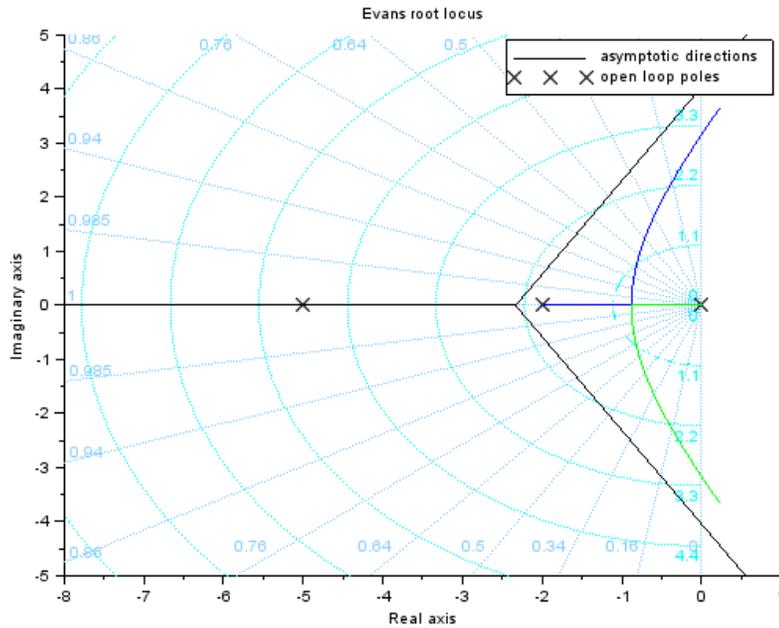


Figure 9.5: To sketch Root locus of given system

```

4
5 // Exercise 6 of Lab 9
6 // To sketch Root locus of given system
7
8 clc
9 close
10 clf
11
12 s=%s
13 h=syslin('c',(s+1)/(s*(s+2)*(s^2+2*s+5)))
14 evans(h,100)
15 sgrid()

```

---

**Scilab code Solution 9.7** To sketch Root locus of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 7 of Lab 9
6 // To sketch Root locus of given system
7
8 clc
9 close
10 clf
11
12 s=%s
13 h=syslin('c',1/(s*(s+2)*(s+5)))
14 evans(h,100)
15 sgrid()
```

---

**Scilab code Solution 9.8** To sketch Root locus of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 8 of Lab 9
6 // To sketch Root locus of given system
7
8 clc
9 close
10 clf
```

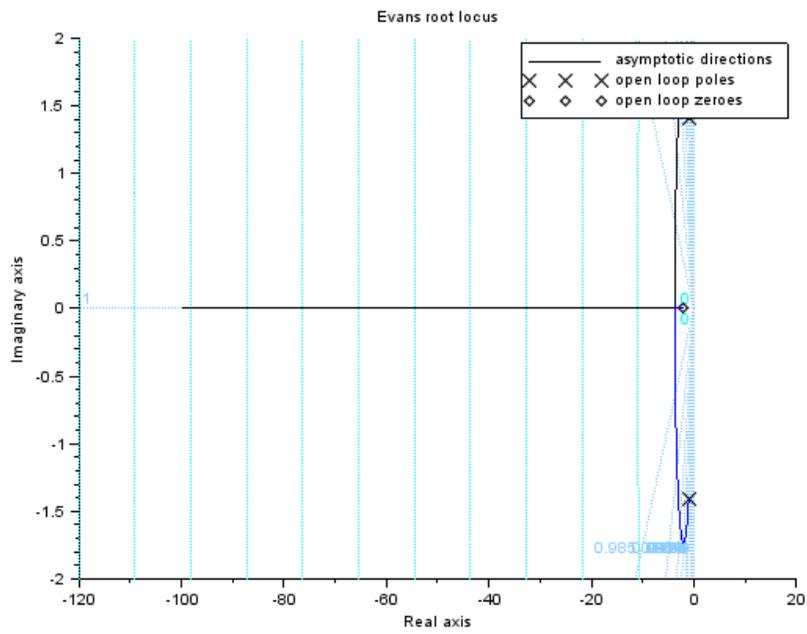


Figure 9.6: To sketch Root locus of given system

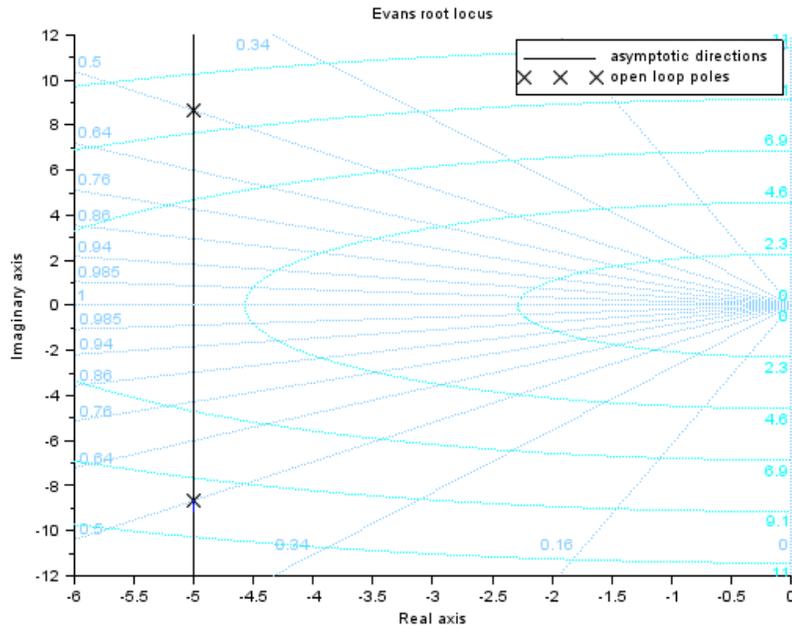


Figure 9.7: To sketch Root locus of given system

```

11
12 s=%s
13 h=syslin('c',(s+2)/(s^2+2*s+3))
14 evans(h,100)
15 sgrid()

```

---

**Scilab code Solution 9.9** To sketch Root locus of given system

```

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

```

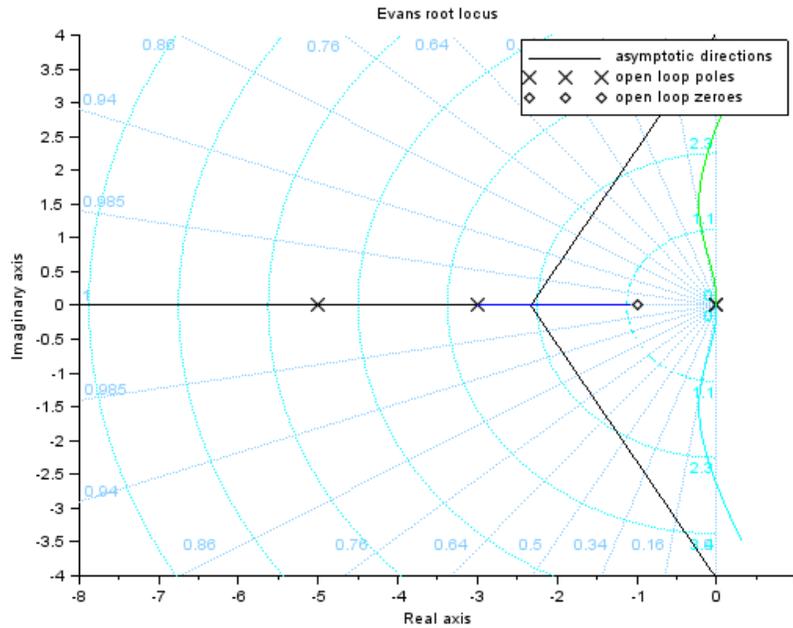


Figure 9.8: To sketch Root locus of given system

```

4
5 // Exercise 9 of Lab 9
6 // To sketch Root locus of given system
7
8 clc
9 close
10 clf
11
12 s=%s
13 h=syslin('c',1/(s^2+10*s+100))
14 evans(h,100)
15 sgrid()

```

---

**Scilab code Solution 9.10** To sketch Root locus of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 10 of Lab 9
6 // To sketch Root locus of given system
7
8 clc
9 close
10 clf
11
12 s=%s
13 h=syslin('c', (s+1)/(s^2*(s+3)*(s+5)))
14 evans(h,100)
15 sgrid()
```

---

**Scilab code Solution 9.11** To sketch Root locus of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 11 of Lab 9
6 // To sketch Root locus of given system
7
8 clc
9 close
10 clf
```

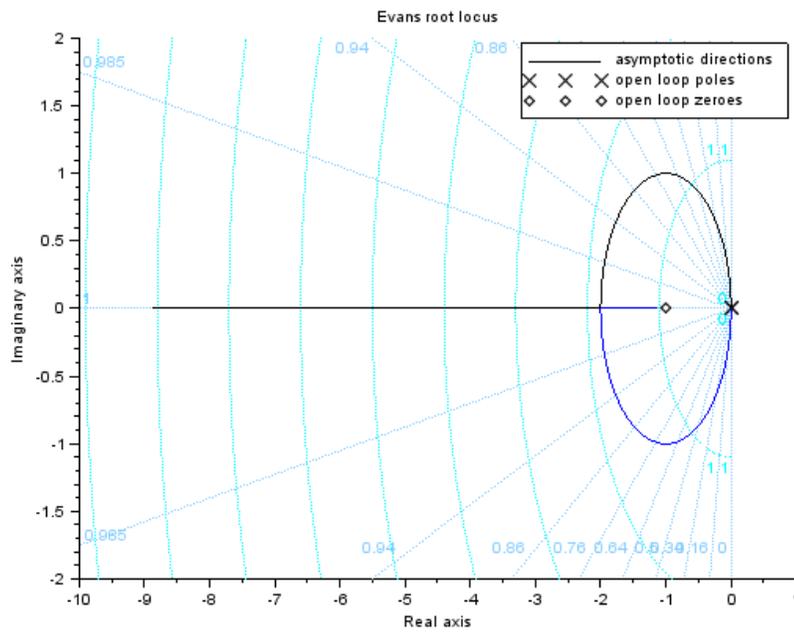


Figure 9.9: To sketch Root locus of given system



```
4
5 // Exercise 12 of Lab 9
6 // To sketch Root locus of given system
7
8 clc
9 close
10 clf
11
12 s=%s
13 h=syslin('c',(s+5)/(s^2*(s+2)))
14 evans(h,100)
15 sgrid()
```

---

# Experiment: 10

## Frequency Response Analysis

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

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 5 of Lab 10
6 // To find Frequency domain specifications for the
   given system
7
8 clc
9 close
10
11 s=poly(0, 's');
12 g1=3
13 g2=12/(s^2+3*s)
14 g=g1*g2
15 tf=g/(1+g)
16
17 // To compare tf with  $\frac{Wn^2}{(s^2+2*\zeta*Wn+Wn^2)}$ 
```

```

18 y=denom(tf)
19 q=numer(tf)
20 z=coeff(y)
21 r=coeff(q)
22 wn=sqrt(r)
23 zeta=z(1,2)/(2*wn)
24
25 mr=1/(2*zeta*sqrt(1-zeta^2))
26 phi_r=-atan(sqrt(1-2*zeta^2)/zeta)
27 wr=wn*sqrt(1-2*zeta^2)
28 BW=wn*sqrt(1-2*zeta^2+sqrt(2-4*zeta^2+4*zeta^4))
29
30 printf("Resonant Peak = %f \n", mr)
31 printf("Phase at resonant peak = %f rad \n", phi_r)
32 printf("Resonant Frequency = %f rad/sec \n", wr)
33 printf("Bandwidth = %f rad/sec \n", BW)

```

---

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

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 6 of Lab 10
6 // To find Frequency domain specifications for the
   given system
7
8 clc
9 close
10
11 s=poly(0, 's');
12 tf=5/(s^2+2*s+5)

```

```

13
14 // To compare tf with  $W_n^2/(s^2+2*\zeta*W_n+W_n^2)$ 
15 y=denom(tf)
16 q=numer(tf)
17 z=coeff(y)
18 r=coeff(q)
19 wn=sqrt(r)
20 zeta=z(1,2)/(2*wn)
21
22 mr=1/(2*zeta*sqrt(1-zeta^2))
23 phi_r=-atan(sqrt(1-2*zeta^2)/zeta)
24 wr=wn*sqrt(1-2*zeta^2)
25 BW=wn*sqrt(1-2*zeta^2+sqrt(2-4*zeta^2+4*zeta^4))
26
27 printf("Resonant Peak = %f \n", mr)
28 printf("Phase at resonant peak = %f rad \n", phi_r)
29 printf("Resonant Frequency = %f rad/sec \n", wr)
30 printf("Bandwidth = %f rad/sec \n", BW)

```

---

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

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 7 of Lab 10
6 // To find Frequency domain specifications for the
   given system
7
8 clc
9 close
10

```

```

11 s=poly(0, 's');
12 g=100/(s*(s+8))
13 tf=g/(1+g)
14
15 // To compare tf with  $W_n^2/(s^2+2*\zeta*W_n+W_n^2)$ 
16 y=denom(tf)
17 q=numer(tf)
18 z=coeff(y)
19 r=coeff(q)
20 wn=sqrt(r)
21 zeta=z(1,2)/(2*wn)
22
23 mr=1/(2*zeta*sqrt(1-zeta^2))
24 phi_r=-atan(sqrt(1-2*zeta^2)/zeta)
25 wr=wn*sqrt(1-2*zeta^2)
26 BW=wn*sqrt(1-2*zeta^2+sqrt(2-4*zeta^2+4*zeta^4))
27
28 printf("Resonant Peak = %f \n", mr)
29 printf("Phase at resonant peak = %f rad \n", phi_r)
30 printf("Resonant Frequency = %f rad/sec \n", wr)
31 printf("Bandwidth = %f rad/sec \n", BW)

```

---

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

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 8 of Lab 10
6 // To find Frequency domain specifications for the
  given system
7

```

```

8  clc
9  close
10
11 s=poly(0, 's');
12 tf=2006/(s^2+10*s+2006)
13
14 // To compare tf with  $W_n^2/(s^2+2*\zeta*W_n+W_n^2)$ 
15 y=denom(tf)
16 q=numer(tf)
17 z=coeff(y)
18 r=coeff(q)
19 wn=sqrt(r)
20 zeta=z(1,2)/(2*wn)
21
22 mr=1/(2*zeta*sqrt(1-zeta^2))
23 phi_r=-atan(sqrt(1-2*zeta^2)/zeta)
24 wr=wn*sqrt(1-2*zeta^2)
25 BW=wn*sqrt(1-2*zeta^2+sqrt(2-4*zeta^2+4*zeta^4))
26
27 printf("Resonant Peak = %f \n", mr)
28 printf("Phase at resonant peak = %f rad \n", phi_r)
29 printf("Resonant Frequency = %f rad/sec \n", wr)
30 printf("Bandwidth = %f rad/sec \n", BW)

```

---

# Experiment: 11

## Polar Plots

Scilab code Solution 11.3 To obtain Polar plot of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 3 of Lab 11
6 // To sketch Polar plot of given system
7
8 clc
9 close
10 clf
11
12 s=0:.1:2*%pi;
13 h=10/(1+5*s);
14 polarplot(s,h);
```

---

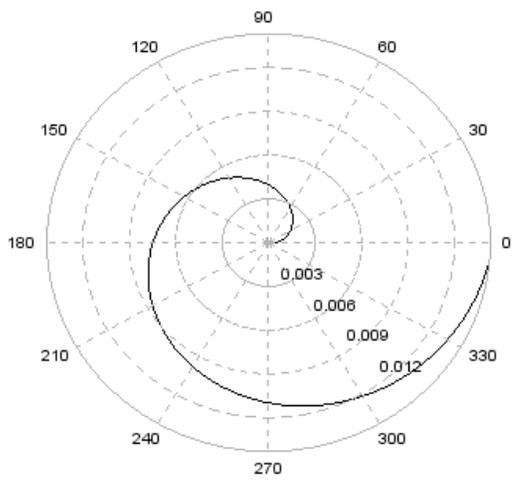


Figure 11.1: To obtain Polar plot of given system

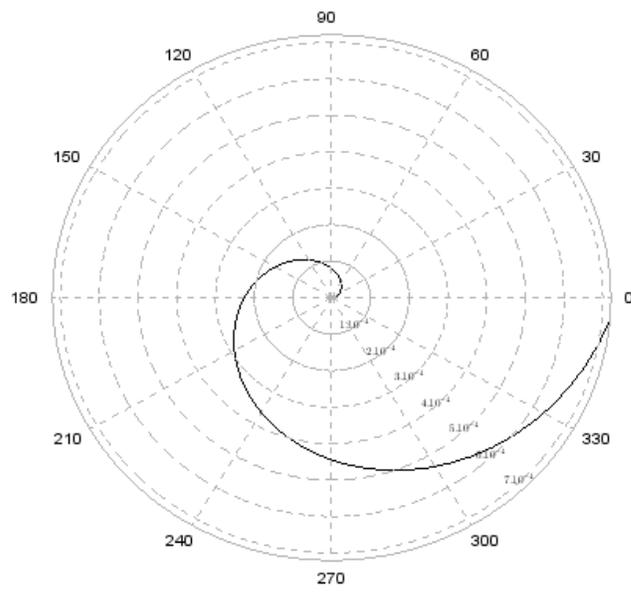


Figure 11.2: To obtain Polar plot of given system

**Scilab code Solution 11.4** To obtain Polar plot of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 4 of Lab 11
6 // To sketch Polar plot of given system
7
8 clc
9 close
10 clf
11
12 s=0:.1:2*%pi;
13 h=1/(1+3*s+2*s^2)
14 polarplot(s,h);
```

---

**Scilab code Solution 11.5** To obtain Polar plot of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 5 of Lab 11
6 // To sketch Polar plot of given system
7
8 clc
9 close
10 clf
11
12 s=0:.1:2*%pi*4;
13 h=1/(s+3*s^2+2*s^3)
14 polarplot(s,h);
```

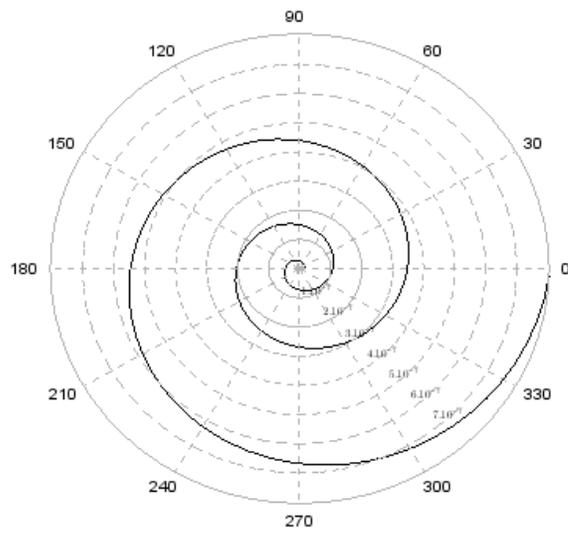


Figure 11.3: To obtain Polar plot of given system

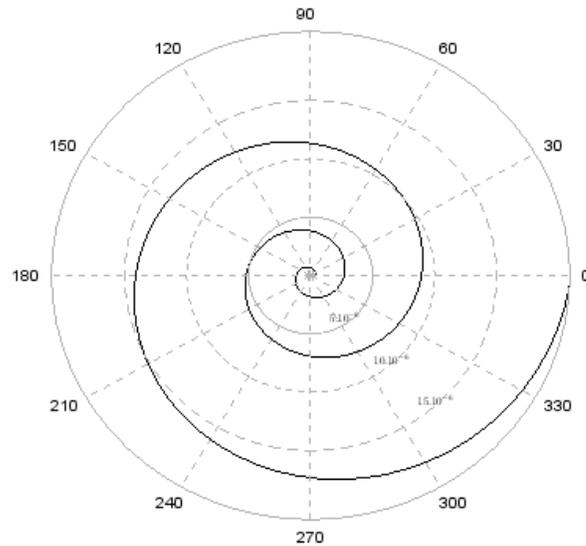


Figure 11.4: To obtain Polar plot of given system

**Scilab code Solution 11.6** To obtain Polar plot of given system

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 6 of Lab 11
6 // To sketch Polar plot of given system
7
8 clc

```

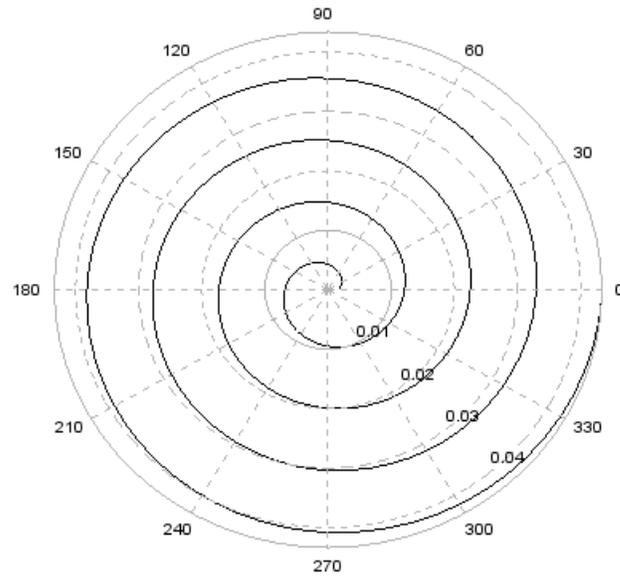


Figure 11.5: To obtain Polar plot of given system

```

9  close
10 clf
11
12 s=0:.1:2*%pi*4;
13 h=100/(s+6*s^2+8*s^3)
14 polarplot(s,h);

```

---

**Scilab code Solution 11.7** To obtain Polar plot of given system

```

1
2

```

```

3
4
5
6 // OS : Windows 7
7 // Scilab : 5.4.1
8 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
9
10 // Exercise 7 of Lab 11
11 // To sketch Polar plot of given system
12
13 clc
14 close
15 clf
16
17 s=0:.1:2*pi*4;
18 h=99/(s+1)
19 polarplot(s,h);

```

---

**Scilab code Solution 11.8** To obtain Polar plot of given system

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 8 of Lab 11
6 // To sketch Polar plot of given system
7
8 clc
9 close
10 clf
11
12 s=0:.1:2*pi*4;
13 h=(200/(s+20))

```

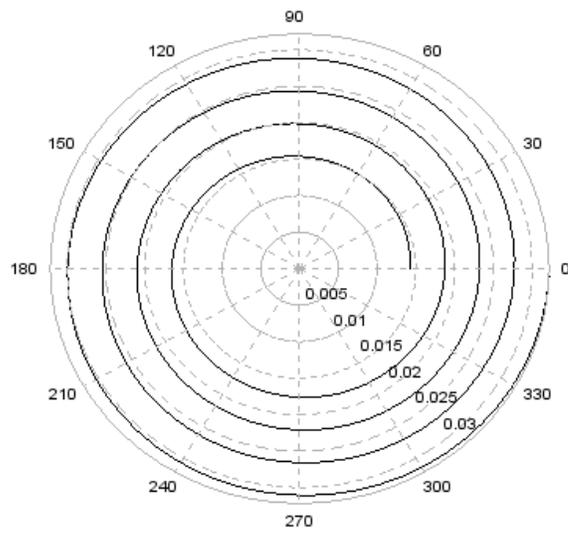


Figure 11.6: To obtain Polar plot of given system

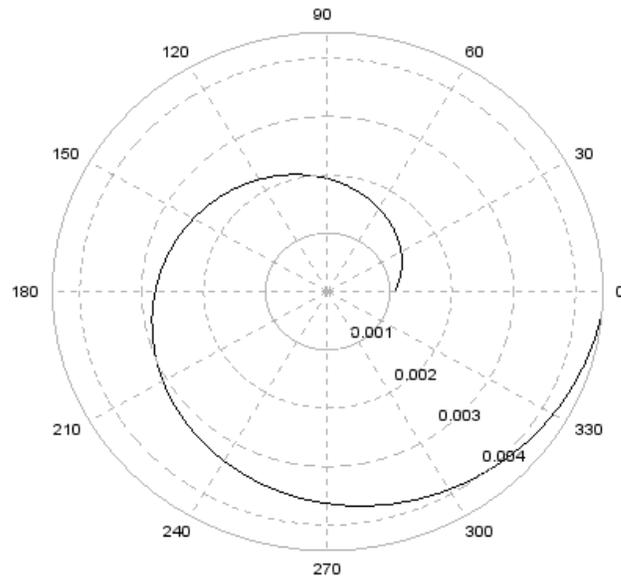


Figure 11.7: To obtain Polar plot of given system

14 `polarplot(s,h);`

---

**Scilab code Solution 11.9** To obtain Polar plot of given system

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 9 of Lab 11
6 // To sketch Polar plot of given system
7

```

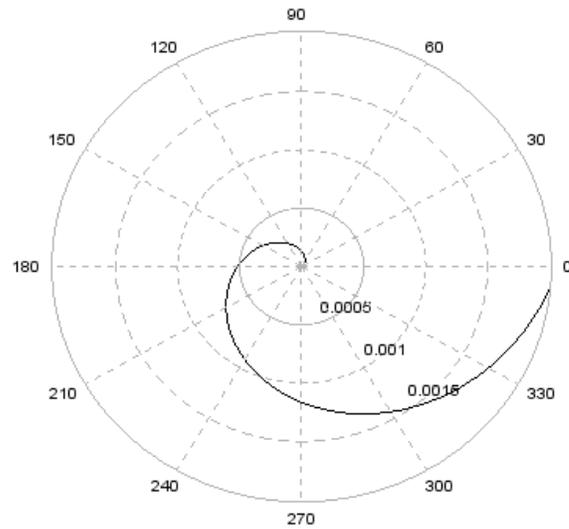


Figure 11.8: To obtain Polar plot of given system

```

8  clc
9  close
10 clf
11
12 s=0:.1:2*%pi;
13 h=(1/(s+2))
14 polarplot(s,h);

```

---

**Scilab code Solution 11.10** To obtain Polar plot of given system

```

1  // OS : Windows 7

```

```

2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 10 of Lab 11
6 // To sketch Polar plot of given system
7
8 clc
9 close
10 clf
11
12 s=0:.1:2*pi;
13 h=(1/(s^2))
14 polarplot(s,h);

```

---

**Scilab code Solution 11.11** To obtain Polar plot of given system

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 11 of Lab 11
6 // To sketch Polar plot of given system
7
8 clc
9 close
10 clf
11
12 s=0:.1:2*pi;
13 h=(1/(s^3))
14 polarplot(s,h);

```

---

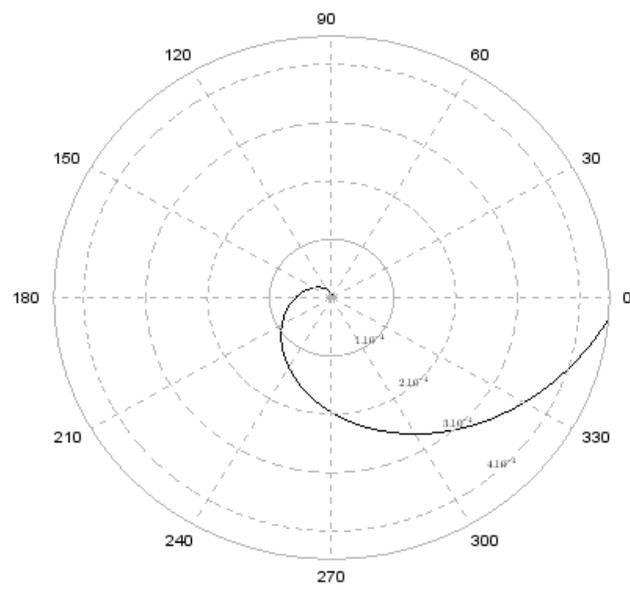


Figure 11.9: To obtain Polar plot of given system

# Experiment: 12

## Nyquist Plots

**Scilab code Solution 12.3** To obtain Nyquist plot of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 3 of Lab 12
6 // To obtain Nyquist plot for given system
7
8 clc
9 close
10
11 s=poly(0, 's')
12 h=syslin('c', (s*4+1)/((s^2*(s+1)*(2*s+1)))
13 nyquist(h);
```

---

**Scilab code Solution 12.4** To obtain Nyquist plot of given system



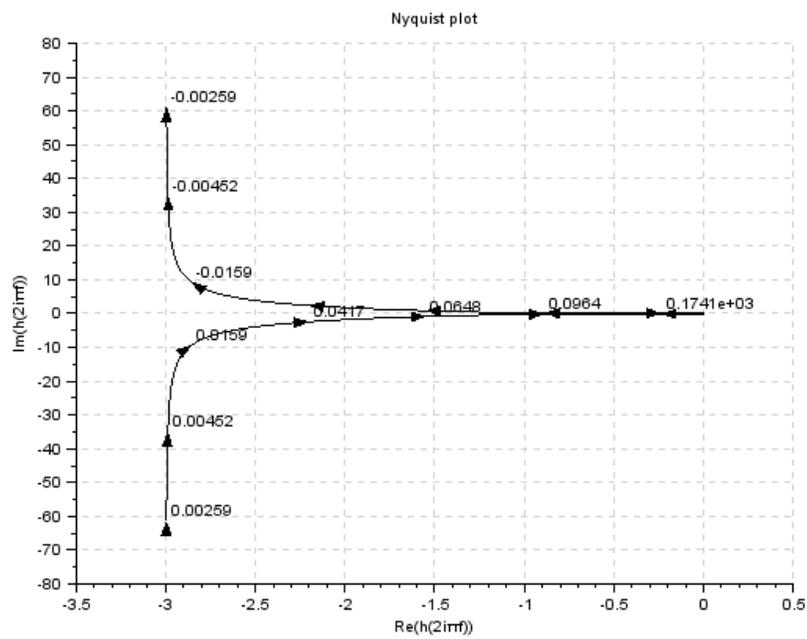


Figure 12.2: To obtain Nyquist plot of given system

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 4 of Lab 12
6 // To obtain Nyquist plot for given system
7
8 clc
9 close
10
11 s=poly(0, 's')
12 h=syslin('c', (1)/((s*(s+1)*(2*s+1))))
13 nyquist(h);

```

---

**Scilab code Solution 12.5** To obtain Nyquist plot of given system

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 5 of Lab 12
6 // To obtain Nyquist plot for given system
7
8 clc
9 close
10
11 s=poly(0, 's')
12 h=syslin('c', (s+2)/((s+1)*(s-1)))
13 nyquist(h);

```

---

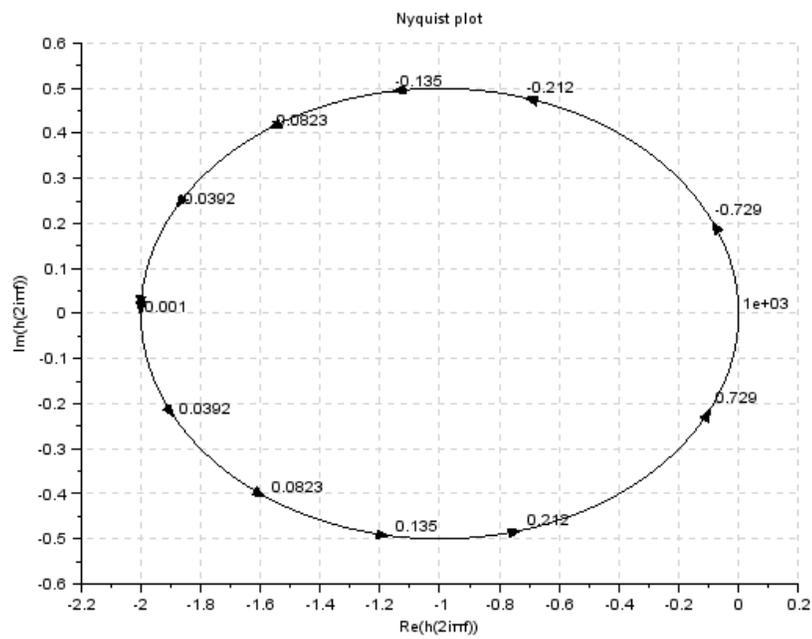


Figure 12.3: To obtain Nyquist plot of given system

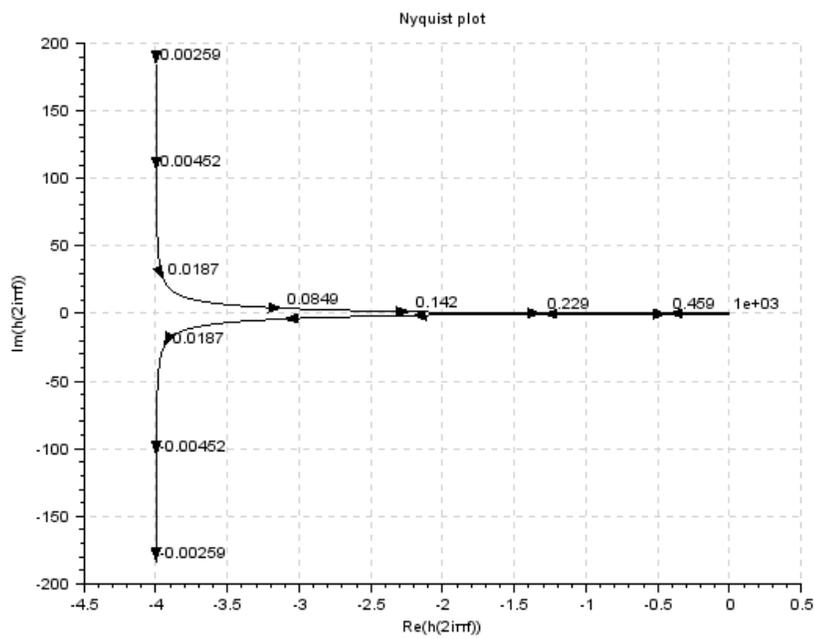


Figure 12.4: To obtain Nyquist plot of given system

**Scilab code Solution 12.6** To obtain Nyquist plot of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 6 of Lab 12
6 // To obtain Nyquist plot for given system
7
8 clc
9 close
10
11 s=poly(0, 's')
12 h=syslin('c', (s+3)/((s*(s-1))))
13 nyquist(h);
```

---

**Scilab code Solution 12.7** To obtain Nyquist plot of given system

```
1
2
3
4
5
6 // OS : Windows 7
7 // Scilab : 5.4.1
8 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
9
10 // Exercise 7 of Lab 12
11 // To obtain Nyquist plot for given system
12
```

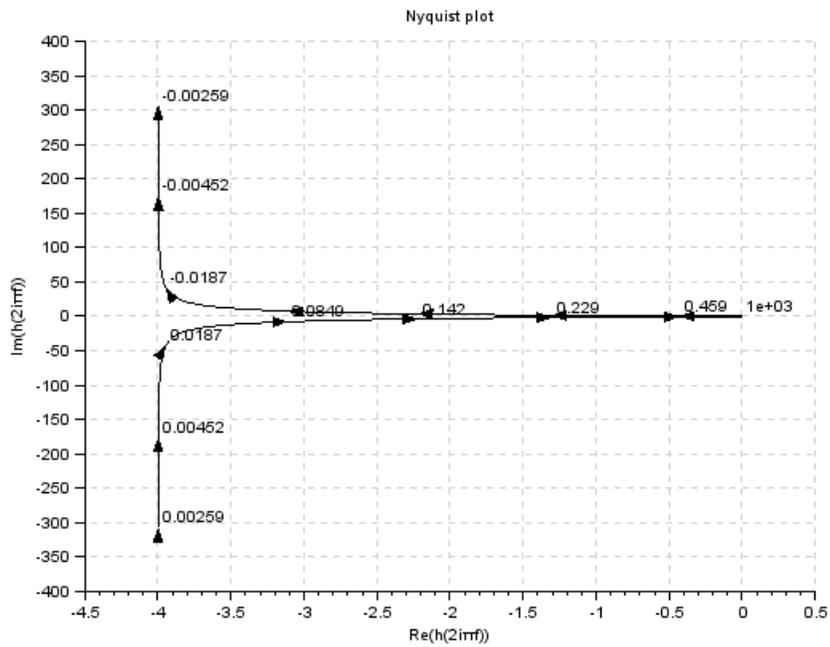


Figure 12.5: To obtain Nyquist plot of given system

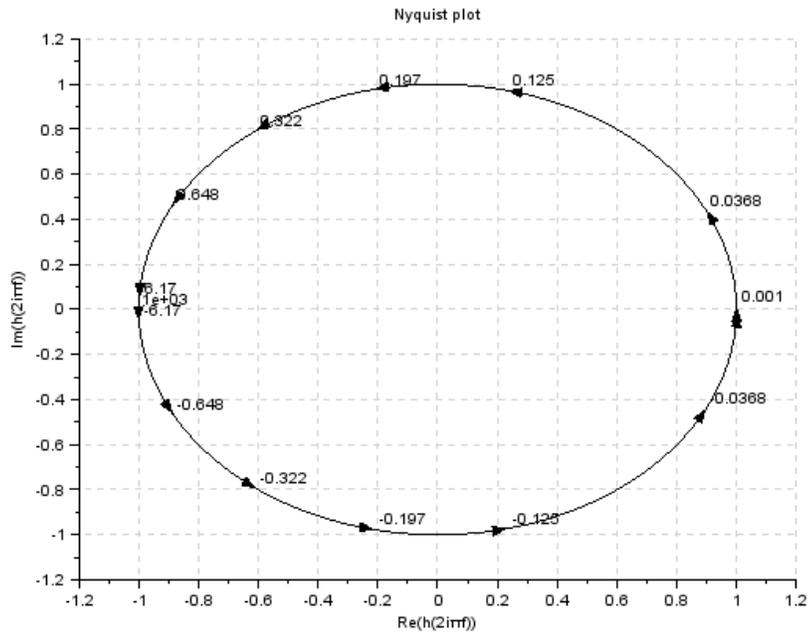


Figure 12.6: To obtain Nyquist plot of given system

```

13 clc
14 close
15
16 s=poly(0, 's')
17 h=syslin('c', (s+5)/((s*(s+1))))
18 nyquist(h);

```

---

**Scilab code Solution 12.8** To obtain Nyquist plot of given system

```

1 // OS : Windows 7
2 // Scilab : 5.4.1

```

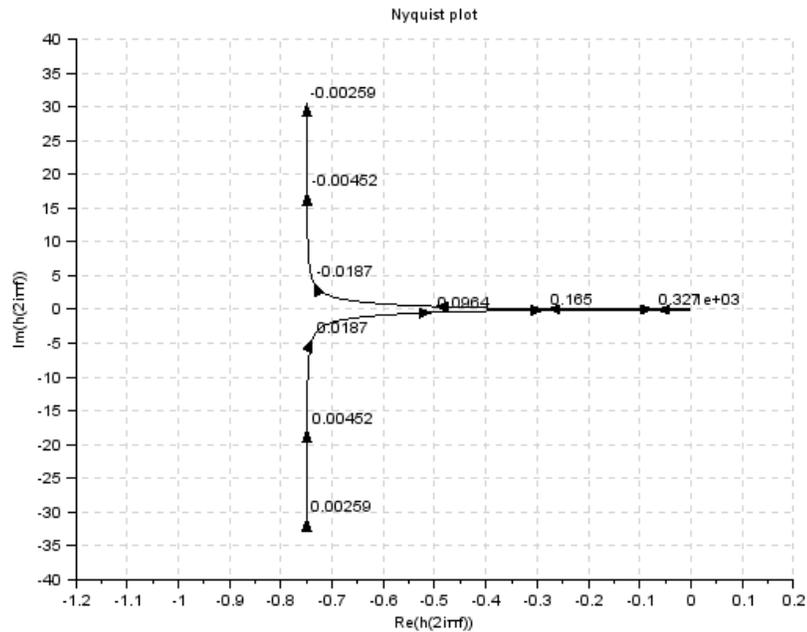


Figure 12.7: To obtain Nyquist plot of given system

```

3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 8 of Lab 12
6 // To obtain Nyquist plot for given system
7
8 clc
9 close
10
11 s=poly(0, 's')
12 h=syslin('c', (1+s)/(1-s))
13 nyquist(h)

```

---

**Scilab code Solution 12.9** To obtain Nyquist plot of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 9 of Lab 12
6 // To obtain Nyquist plot for given system
7
8 clc
9 close
10
11 s=poly(0, 's')
12 h=syslin('c', 1/((s*(s+1)*(s+2))))
13 nyquist(h);
```

---

**Scilab code Solution 12.10** To obtain Nyquist plot of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 10 of Lab 12
6 // To obtain Nyquist plot for given system
7
8 clc
9 close
10
11 s=poly(0, 's')
12 h=syslin('c', (s^2+2*s+3)/(s+2))
13 nyquist(h, 0.000001, 1)
```

---

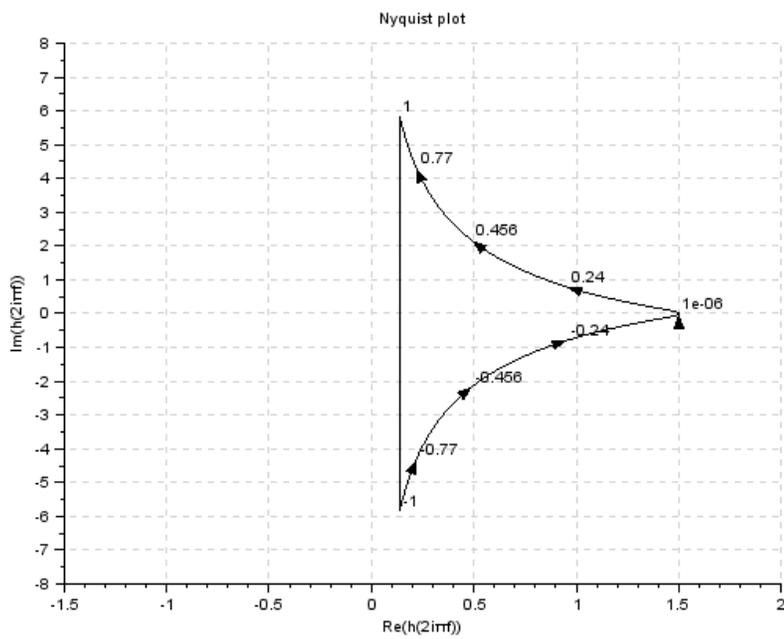


Figure 12.8: To obtain Nyquist plot of given system

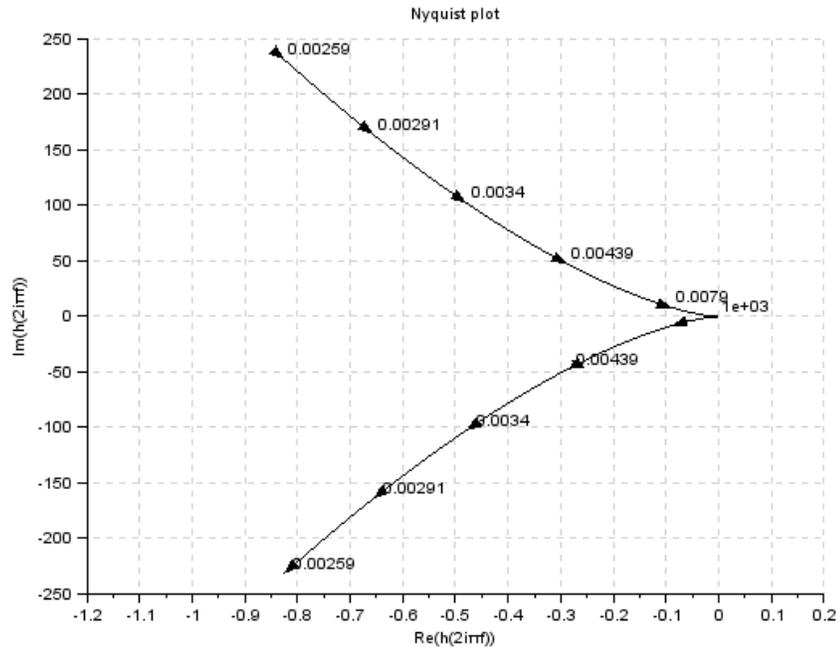


Figure 12.9: To obtain Nyquist plot of given system

**Scilab code Solution 12.11** To obtain Nyquist plot of given system

```

1
2
3
4
5
6 // OS : Windows 7
7 // Scilab : 5.4.1
8 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2

```

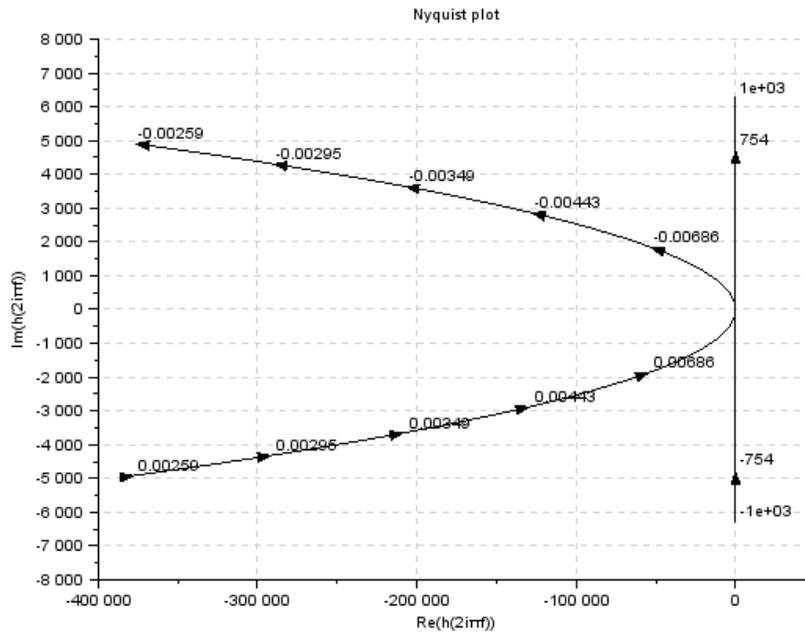


Figure 12.10: To obtain Nyquist plot of given system

```

9
10 // Exercise 11 of Lab 12
11 // To obtain Nyquist plot for given system
12
13 clc
14 close
15
16 s=poly(0, 's')
17 h=syslin('c', (5+s)*(s+40)/((s^3)*(s+200)*(s+1000)))
18 nyquist(h)

```

---

**Scilab code Solution 12.12** To obtain Nyquist plot of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 12 of Lab 12
6 // To obtain Nyquist plot for given system
7
8 clc
9 close
10
11 s=poly(0, 's')
12 h=syslin('c', (5+s)/(s^2)*(s+2)*(s+10))
13 nyquist(h)
```

---

# Experiment: 13

## Bode Plots

Scilab code Solution 13.3 To obtain Bode plot of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 3 of Lab 13
6 // To obtain Bode plot of given system
7
8 clc
9 close
10
11 s=poly(0, 's')
12 h=syslin('c',31.62*(s*0.125+1)/((s)*(s*2+1)*(s
    *0.044+1)))
13 clf();
14 bode(h,0.1,100)
15 g_margin(h)
16 show_margins(h)
17 p_margin(h)
18 show_margins(h)
```

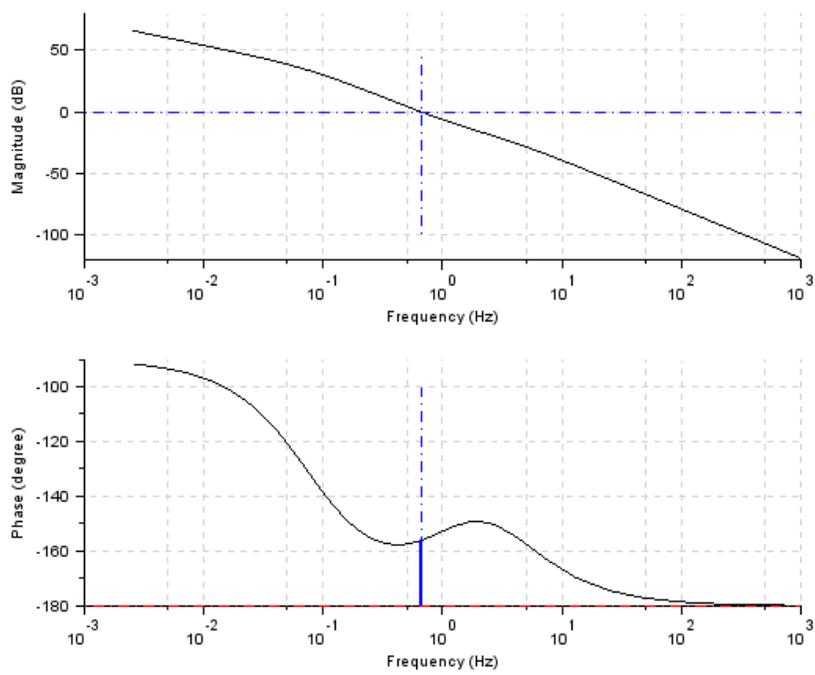


Figure 13.1: To obtain Bode plot of given system

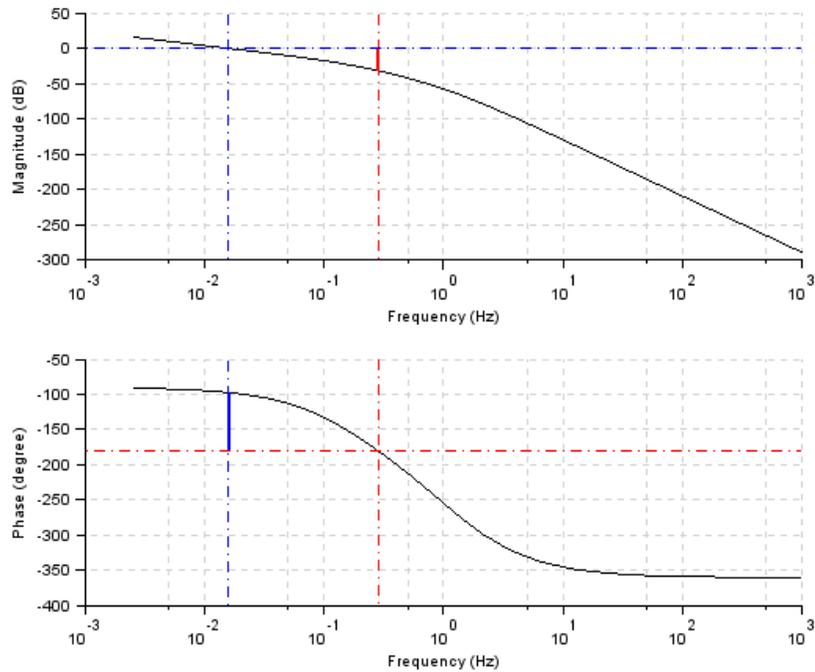


Figure 13.2: To obtain Bode plot of given system

---

**Scilab code Solution 13.4** To obtain Bode plot of given system

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 4 of Lab 13
6 // To obtain Bode plot of given system
7
8 clc

```

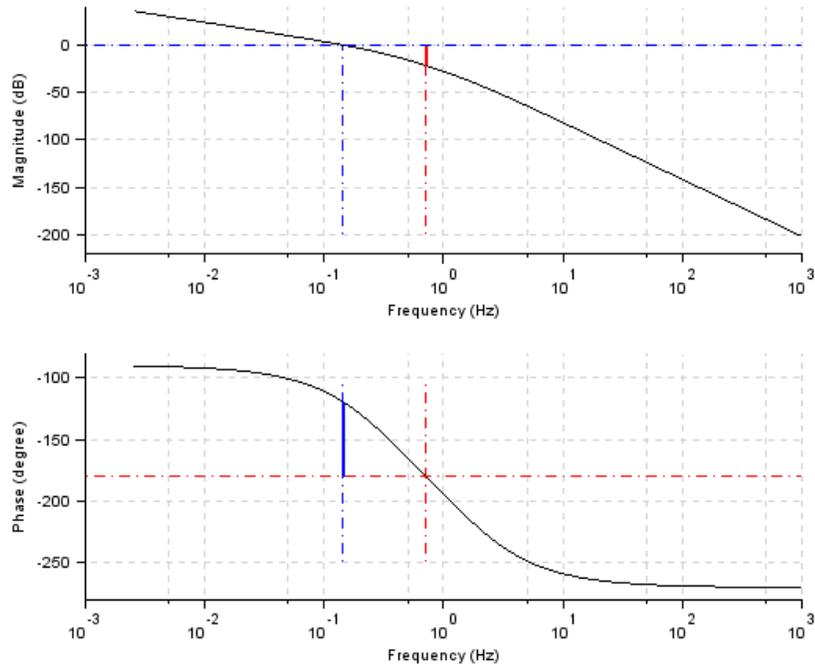


Figure 13.3: To obtain Bode plot of given system

```

9  close
10
11  s=poly(0, 's')
12  h=syslin('c',5/((s)*(s+10)*(s+5)*(s+1)))
13  clf();
14  bode(h,0.1,100)
15  g_margin(h)
16  show_margins(h)
17  p_margin(h)
18  show_margins(h)

```

---

**Scilab code Solution 13.5** To obtain Bode plot of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 5 of Lab 13
6 // To obtain Bode plot of given system
7
8 clc
9 close
10
11 s=poly(0, 's')
12 h=syslin('c', 1/((s)*(s*0.5+1)*(s*0.1+1)))
13 clf();
14 bode(h, 0.1, 100)
15 g_margin(h)
16 show_margins(h)
17 p_margin(h)
18 show_margins(h)
```

---

**Scilab code Solution 13.6** To obtain Bode plot of given system

```
1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 6 of Lab 13
6 // To obtain Bode plot of given system
7
8 clc
9 close
10
```

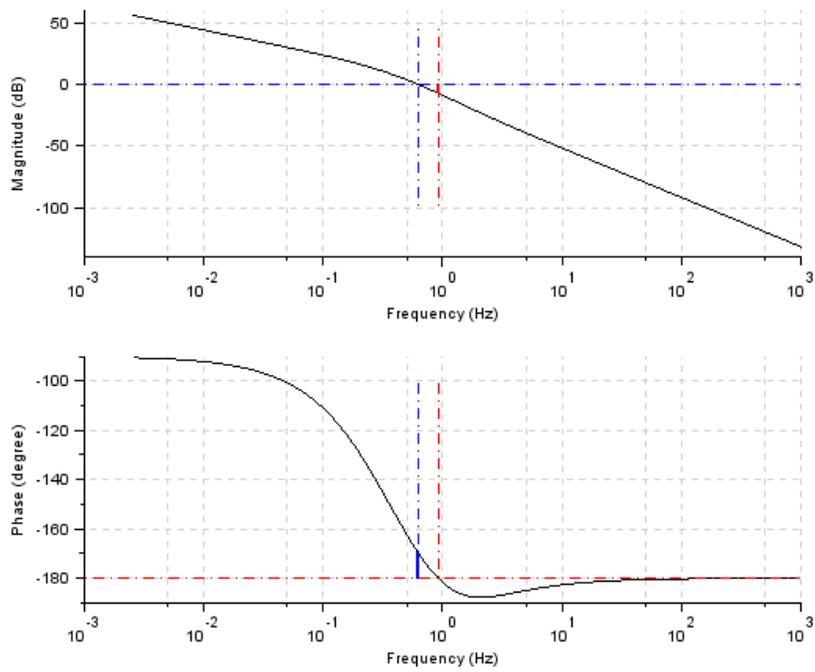


Figure 13.4: To obtain Bode plot of given system

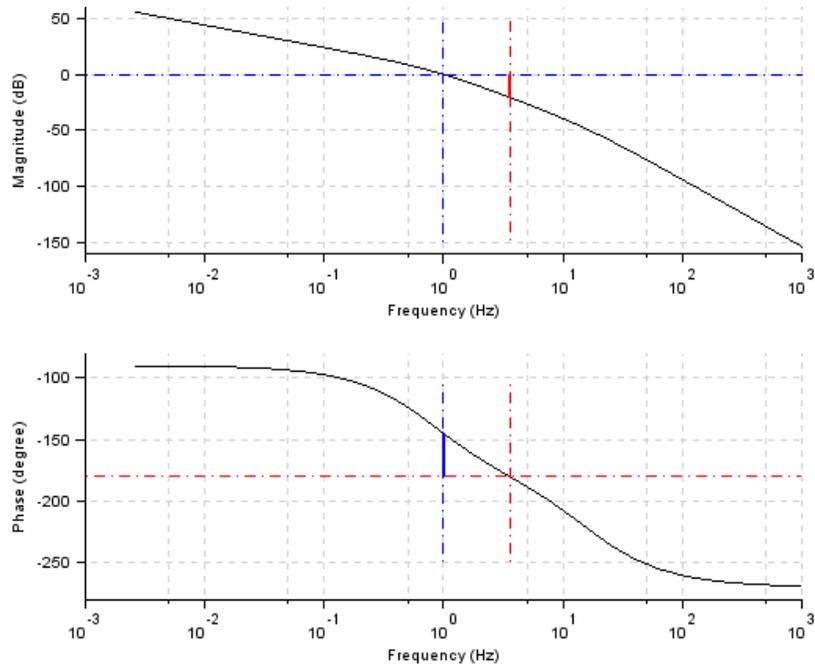


Figure 13.5: To obtain Bode plot of given system

```

11 s=poly(0, 's')
12 h=syslin('c', 10*(s+10)/((s)*(s+2)*(s+5)))
13 clf();
14 bode(h, 0.1, 100)
15 g_margin(h)
16 show_margins(h)
17 p_margin(h)
18 show_margins(h)

```

---

**Scilab code Solution 13.7** To obtain Bode plot of given system

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 7 of Lab 13
6 // To obtain Bode plot of given system
7
8 clc
9 close
10
11 s=poly(0, 's')
12 h=syslin('c', 10/((s)*(s*0.2+1)*(s*0.01+1)))
13 clf();
14 bode(h, 0.1, 100)
15 g_margin(h)
16 show_margins(h)
17 p_margin(h)
18 show_margins(h)

```

---

**Scilab code Solution 13.8** To obtain Bode plot of given system

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 8 of Lab 13
6 // To obtain Bode plot of given system
7
8 clc
9 close
10
11 s=poly(0, 's')
12 h=syslin('c', 10/((s)*(s*0.01+1)*(s*0.1+1)))

```

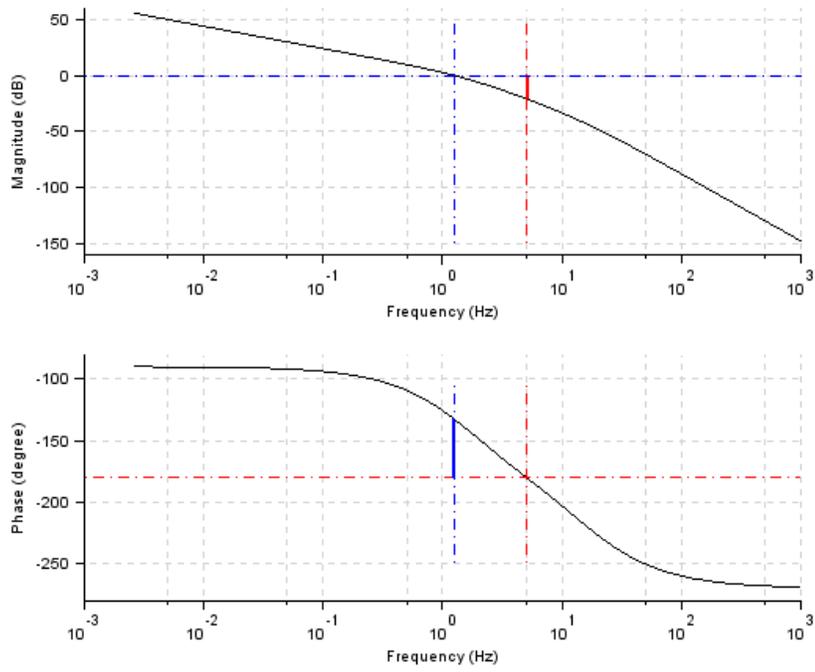


Figure 13.6: To obtain Bode plot of given system

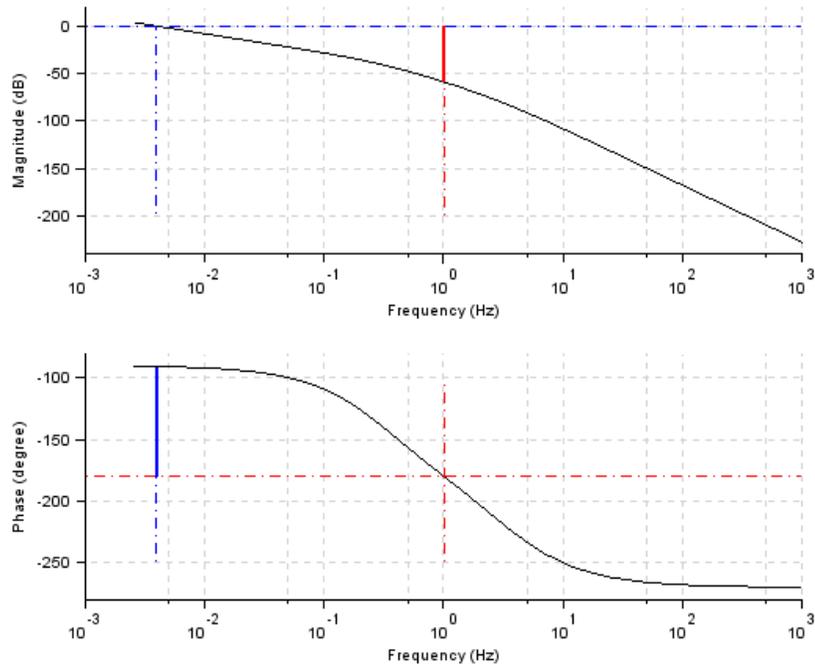


Figure 13.7: To obtain Bode plot of given system

```

13 clf();
14 bode(h,0.1,100)
15 g_margin(h)
16 show_margins(h)
17 p_margin(h)
18 show_margins(h)

```

---

**Scilab code Solution 13.9** To obtain Bode plot of given system

```

1 // OS : Windows 7
2 // Scilab : 5.4.1

```

```

3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 9 of Lab 13
6 // To obtain Bode plot of given system
7
8 clc
9 close
10
11 s=poly(0, 's')
12 h=syslin('c', 0.025/((s)*(s*0.05+1)*(s*0.5+1)))
13 clf();
14 bode(h, 0.1, 100)
15 g_margin(h)
16 show_margins(h)
17 p_margin(h)
18 show_margins(h)

```

---

**Scilab code Solution 13.10** To obtain Bode plot of given system

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 10 of Lab 13
6 // To obtain Bode plot of given system
7
8 clc
9 close
10
11 s=poly(0, 's')
12 h=syslin('c', 50/((s)*(s+1)*(s*0.5+1)))
13 clf();
14 bode(h, 0.1, 100)

```

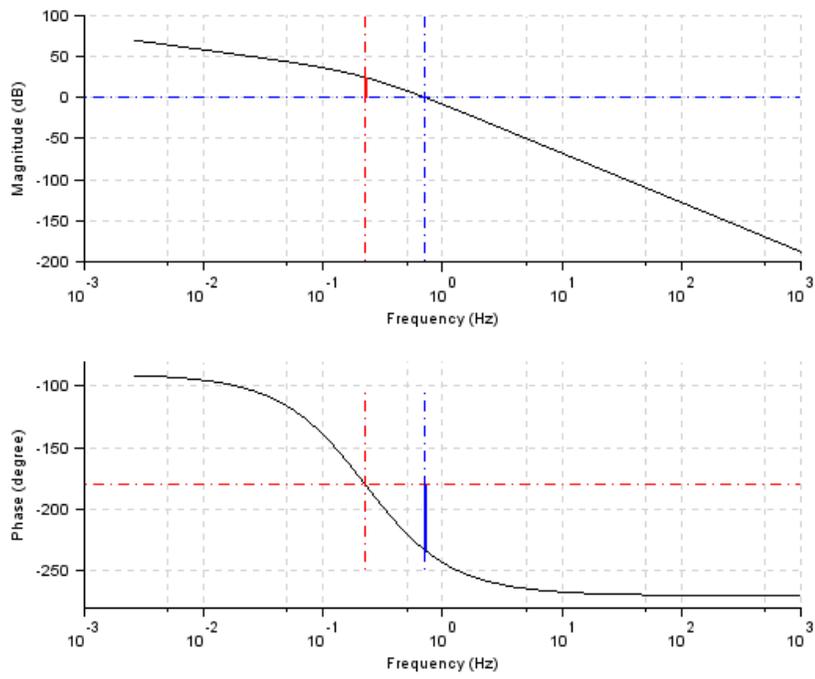


Figure 13.8: To obtain Bode plot of given system

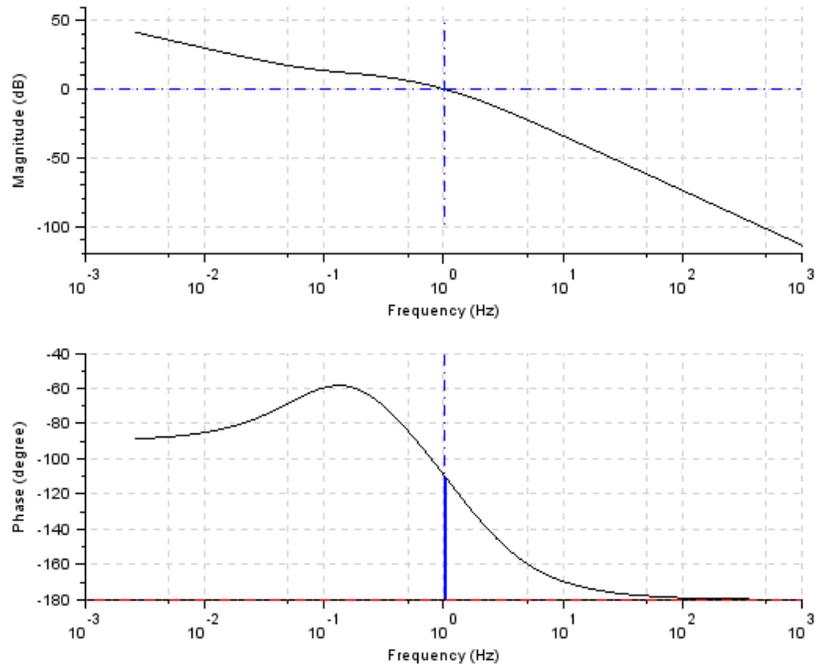


Figure 13.9: To obtain Bode plot of given system

```

15 g_margin(h)
16 show_margins(h)
17 p_margin(h)
18 show_margins(h)

```

---

**Scilab code Solution 13.11** To obtain Bode plot of given system

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4

```

```

5 // Exercise 11 of Lab 13
6 // To obtain Bode plot of given system
7
8 clc
9 close
10
11 s=poly(0, 's')
12 h=syslin('c', (2+4*s)/((s)*(s*0.5+1)*(s*0.1+1)))
13 clf();
14 bode(h,0.1,100)
15 g_margin(h)
16 show_margins(h)
17 p_margin(h)
18 show_margins(h)

```

---

**Scilab code Solution 13.12** To obtain Bode plot of given system

```

1 // OS : Windows 7
2 // Scilab : 5.4.1
3 // Toolboxes : Maxima 5.20.1 and ActivePerl 5.20.2
4
5 // Exercise 12 of Lab 13
6 // To obtain Bode plot of given system
7
8 clc
9 close
10
11 s=poly(0, 's')
12 h=syslin('c', 1/s^3)
13 clf();
14 bode(h,0.1,100)
15 g_margin(h)
16 show_margins(h)

```

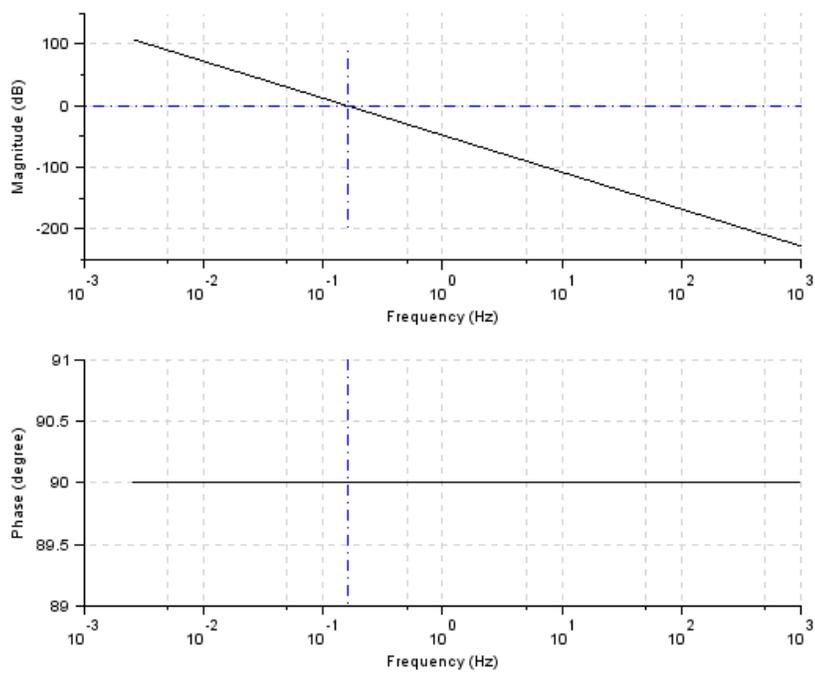


Figure 13.10: To obtain Bode plot of given system

```
17 p_margin(h)
18 show_margins(h)
```

---