

Scilab Manual for  
Advanced Control Systems  
by Prof Deepti Khimani  
Instrumentation Engineering  
VESIT<sup>1</sup>

Solutions provided by  
Prof Mrs. Deepti Khimani  
Instrumentation Engineering  
Mumbai University/V. E. S. Institute of Technology

June 4, 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   | 4  |
| 1 To plot the phase portrait of systems having stable and unstable nodes.      | 6  |
| 2 To plot the phase portrait of systems having stable and unstable focus.      | 11 |
| 3 To plot the phase portrait of systems having vortex point.                   | 15 |
| 4 To plot the phase portrait of systems having saddle point.                   | 19 |
| 5 To demonstrate limit cycles for vander pol's equation.                       | 23 |
| 6 To demonstrate the effect of the static nonlinearities.                      | 25 |
| 7 To demonstrate the stability of the system using Describing function.        | 27 |
| 8 To demonstrate the stability of the system using Lyapunov equation.          | 30 |
| 9 Stabilization of double integrator system using variable structure control.  | 32 |
| 10 Design the exact feedback linearizing controller for the non linear system. | 34 |
| 11 To design sliding mode controller for a linear system.                      | 36 |

12 To demonstrate model reference adaptive control system. 38

# List of Experiments

|               |       |       |    |
|---------------|-------|-------|----|
| Solution 1.01 | Lab01 | ..... | 6  |
| Solution 2.1  | Lab02 | ..... | 11 |
| Solution 3.01 | Lab03 | ..... | 15 |
| Solution 4.01 | Lab04 | ..... | 19 |
| Solution 7.01 | Lab07 | ..... | 27 |
| Solution 8.01 | Lab08 | ..... | 30 |

# List of Figures

|      |       |       |    |
|------|-------|-------|----|
| 1.1  | Lab01 | ..... | 9  |
| 1.2  | Lab01 | ..... | 10 |
| 2.1  | Lab02 | ..... | 14 |
| 2.2  | Lab02 | ..... | 14 |
| 3.1  | Lab03 | ..... | 17 |
| 3.2  | Lab03 | ..... | 18 |
| 4.1  | Lab04 | ..... | 21 |
| 4.2  | Lab04 | ..... | 22 |
| 5.1  | Lab05 | ..... | 24 |
| 6.1  | Lab06 | ..... | 26 |
| 6.2  | Lab06 | ..... | 26 |
| 7.1  | Lab07 | ..... | 29 |
| 9.1  | Lab09 | ..... | 33 |
| 10.1 | Lab10 | ..... | 35 |
| 11.1 | Lab11 | ..... | 37 |
| 11.2 | Lab11 | ..... | 37 |
| 12.1 | Lab12 | ..... | 39 |

# Experiment: 1

To plot the phase portrait of systems having stable and unstable nodes.

Scilab code Solution 1.01 Lab01

```
1 //Lab. 01: To plot the phase portrait of systems
   having stable and unstable nodes.
2
3 //scilab - 5.5.0
4 // Operating System : Windows 7, 32-bit
5
6 clc;
7 clear all;
8 clf;
9
10 //System transfer function
11 s=poly(0, 's');
12 g=1/(s^2+3*s+2);
13
14 //Draw pole zero map of the system
15 plzr(g);
16 title('Pole-zero map of the system with real stable
```

```

    eigen values ', 'fontsize',3)
17 //Convert the given transfer function into state
    space form
18 sys=tf2ss(g);
19
20 //Plot of system phase trajectory
21 sys.c=[1,0;0,1];
22 sys.d=[0 0]';
23
24 t=0:0.2:10;
25 a1=size(t);
26 u=zeros(a1(1),a1(2));
27 figure
28 for i=-2.0:0.5:2;
29     for j=-2:0.5:2;
30         y1=csim(u,t,sys,[i,j]');
31         plot(y1(1,:),y1(2,:));
32     end
33 end
34
35 set(gca(),"grid",[0.3 0.3])
36 title('Phase portrait of the system with stable node
    ', 'fontsize',3)
37 xlabel('x1(t)', 'fontsize',2)
38 ylabel('x2(t)', 'fontsize',2)
39 f=get("current_figure") //Current figure handle
40 f.background=8
41 //System transfer function
42 s=poly(0,'s');
43 g=1/(s^2-3*s+2);
44
45 // Draw pole zero map of the system
46 figure;
47 plzr(g);
48 title('Pole-zero map of the system with real
    unstable eigen values ', 'fontsize',3)
49 f=get("current_figure") //Current figure handle
50 f.background=8

```

```

51 //Convert the given transfer function into state
    space form
52 sys=tf2ss(g);
53
54 //Plot of system phase trajectory
55 sys.c=[1,0;0,1];
56 sys.d=[0 0]';
57
58 a1=size(t);
59 u=zeros(a1(1),a1(2));
60 figure
61 for i=-2.0:0.5:2;
62     for j=-2:0.5:2;
63         y1=csim(u,t,sys,[i,j]');
64         plot(y1(1,:),y1(2,:));
65     end
66 end
67 set(gca(),"grid",[0.3 0.3])
68 f=get("current_figure") //Current figure handle
69 f.background=8
70 zoom_rect([-5,-5,5,5])
71 title('Phase portrait of the system with unstable
        node','fontsize',3)
72 xlabel('x1(t)','fontsize',2)
73 ylabel('x2(t)','fontsize',2)

```

---

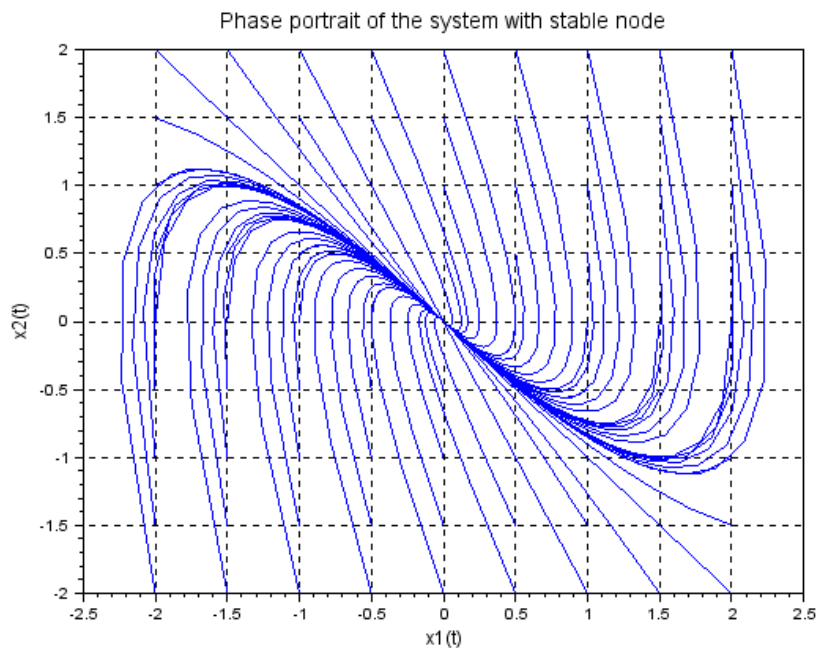


Figure 1.1: Lab01

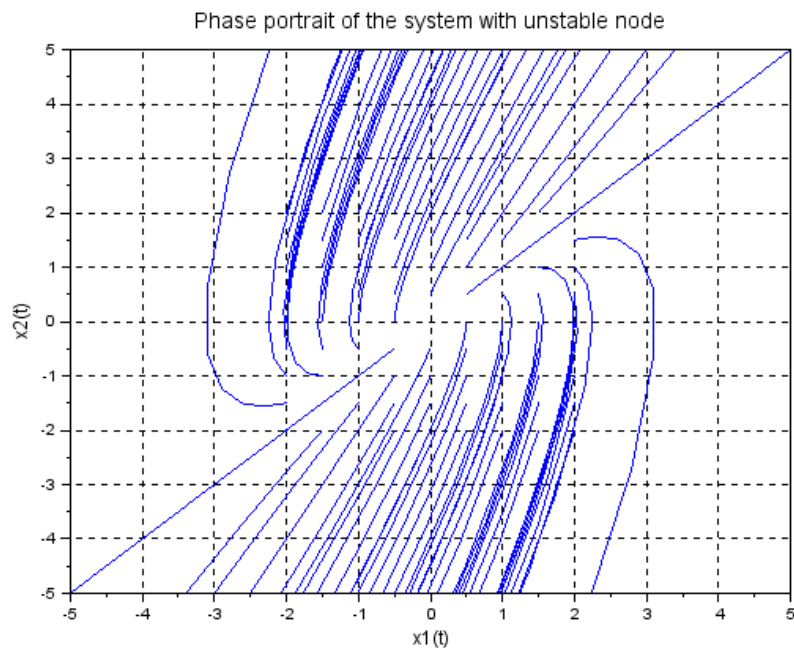


Figure 1.2: Lab01

## Experiment: 2

To plot the phase portrait of systems having stable and unstable focus.

Scilab code Solution 2.1 Lab02

```
1 //Lab. 02: To plot the phase portrait of systems
   having stable and unstable focus point.
2
3 //scilab - 5.5.0
4 // Operating System : Windows 7, 32-bit
5
6 clc;
7 clear all;
8 clf;
9
10 //System transfer function
11 s=poly(0, 's');
12 g=1/(s^2+s+1);
13
14 // Draw pole zero map of the system
15 plzr(g);
16 title('Pole-zero map of the system with stable
```

```

    underdamped eigen values ', 'fontsize', 3)
17 //Convert the given transfer function into state
    space form
18 sys=tf2ss(g);
19
20 //Plot of system phase trajectory
21 sys.c=[1,0;0,1];
22 sys.d=[0 0]';
23
24 t=0:0.2:10;
25 a1=size(t);
26 u=zeros(a1(1),a1(2));
27 figure
28 for i=-2.0:0.5:2;
29     for j=-2:0.5:2;
30 y1=csim(u,t,sys,[i,j]');
31 plot(y1(1,:),y1(2,:));
32 end
33 end
34 set(gca(),'grid',[0.3 0.3])
35 f=get("current_figure") //Current figure handle
36 f.background=8
37 title('Phase portrait of the system with stable
    focus ', 'fontsize', 3)
38 xlabel('x1(t)', 'fontsize', 2)
39 ylabel('x2(t)', 'fontsize', 2)
40
41 //System transfer function
42 s=poly(0, 's');
43 g=1/(s^2-s+1);
44
45 //Convert the given transfer function into state
    space form
46 sys=tf2ss(g);
47
48 // Draw pole zero map of the system
49 figure;
50 plzr(g);

```

```

51 f=get("current_figure") //Current figure handle
52 f.background=8
53 title('Pole-zero map of the system with negatively
        damped eigen values','fontsize',3)
54 //Plot of system phase trajectory
55 sys.c=[1,0;0,1];
56 sys.d=[0 0]';
57
58 a1=size(t);
59 u=zeros(a1(1),a1(2));
60 figure
61 for i=-2.0:0.5:2;
62     for j=-2:0.5:2;
63 y1=csim(u,t,sys,[i,j]');
64 plot(y1(1,:),y1(2,:));
65 end
66 end
67 set(gca(),"grid",[0.3 0.3])
68 f=get("current_figure") //Current figure handle
69 f.background=8
70 zoom_rect([-5,-5,5,5])
71 title('Phase portrait of the system with unstable
        focus ','fontsize',3)
72 xlabel('x1(t)','fontsize',2)
73 ylabel('x2(t)','fontsize',2)

```

---

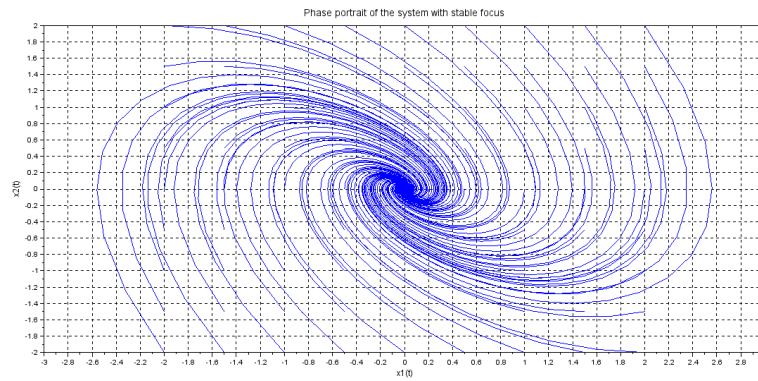


Figure 2.1: Lab02

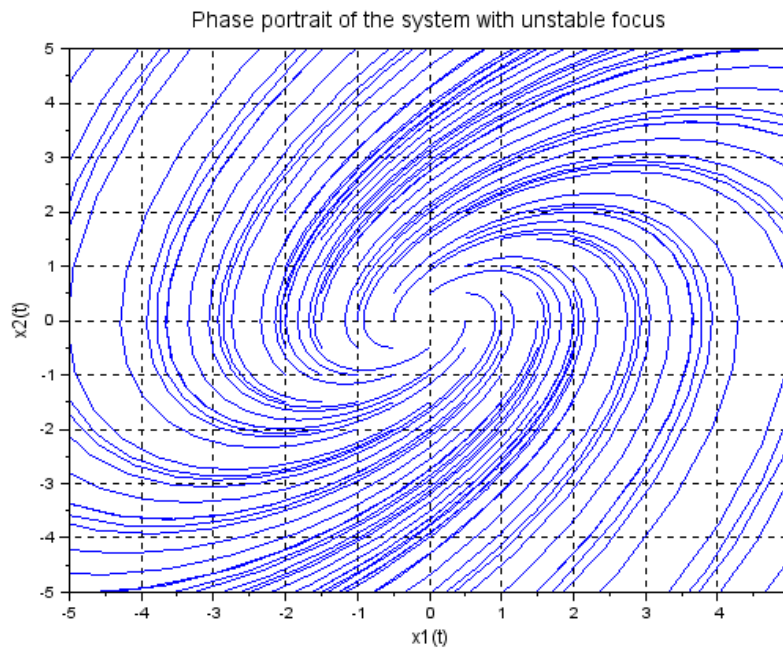


Figure 2.2: Lab02

## Experiment: 3

### To plot the phase portrait of systems having vortex point.

Scilab code Solution 3.01 Lab03

```
1 //Lab. 03: To plot the phase portrait of systems
   having vortex point.
2
3 //scilab - 5.5.0
4 // Operating System : Windows 7, 32-bit
5
6 clc;
7 clear all;
8 clf;
9
10 //System transfer function
11 s=poly(0, 's');
12 g=1/(s^2+4);
13
14 // Draw pole zero map of the system
15 plzr(g);
16 title('Pole-zero map of the system with critically
   damped eigen values ','fontsize',3)
17 //Convert the given transfer function into state
```

```

        space form
18 sys=tf2ss(g);
19
20 //Plot of system phase trajectory
21 sys.c=[1,0;0,1];
22 sys.d=[0 0]';
23
24 t=0:0.2:10;
25 a1=size(t);
26 u=zeros(a1(1),a1(2));
27 figure
28 for i=-2.0:0.5:2;
29     for j=-2:0.5:2;
30 y1=csim(u,t,sys,[i,j]');
31 plot(y1(1,:),y1(2,:));
32 end
33 end
34 set(gca(),"grid",[0.3 0.3])
35 f=get("current_figure") //Current figure handle
36 f.background=8
37 title('Phase portrait of the system with vortex
        point','fontsize',3)
38 xlabel('x1(t)','fontsize',2)
39 ylabel('x2(t)','fontsize',2)

```

---

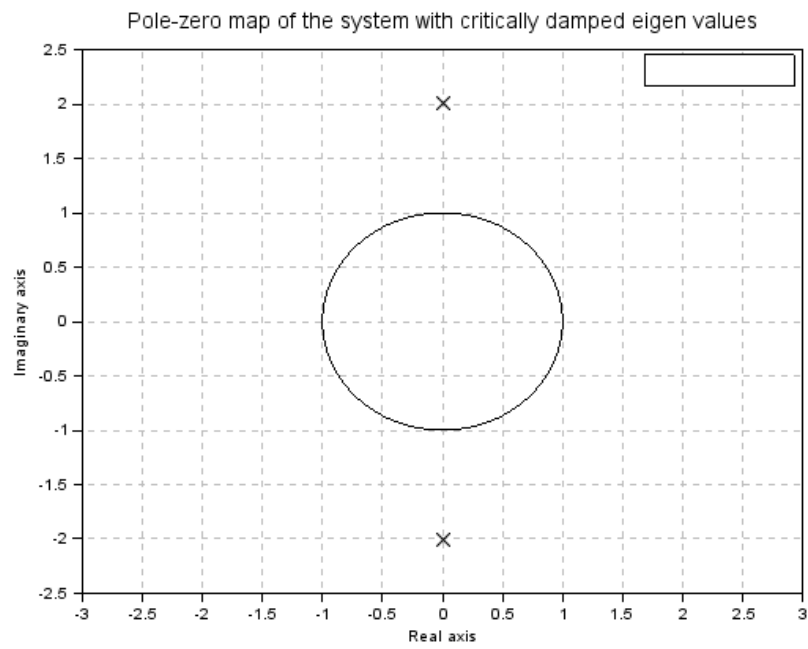


Figure 3.1: Lab03

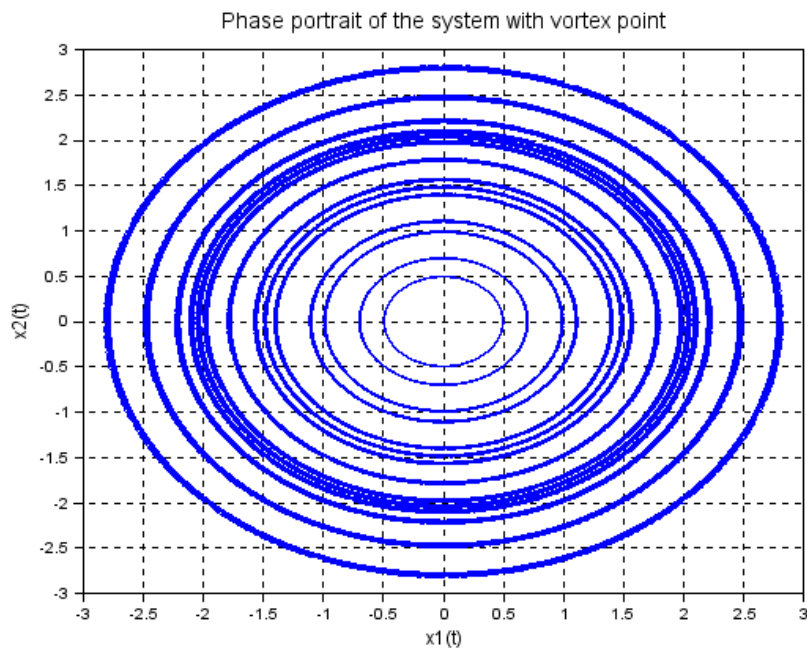


Figure 3.2: Lab03

## Experiment: 4

To plot the phase portrait of systems having saddle point.

Scilab code Solution 4.01 Lab04

```
1 //Lab. 04: To plot the Phase portrait of systems
   having saddle point.
2
3 //scilab - 5.5.0
4 // Operating System : Windows 7, 32-bit
5
6 clc;
7 clear all;
8 clf;
9
10 //System transfer function
11 s=poly(0, 's');
12 g=1/(s^2+1*s-2);
13
14 //Convert the given transfer function into state
   space form
15 sys=tf2ss(g);
16
17 // Draw pole zero map of the system
```

```

18 plzr(sys);
19 title('Pole-zero map of the system with real stable
        and unstable eigen values','fontsize',3)
20 //Plot of system phase trajectory
21 sys.c=[1,0;0,1];
22 sys.d=[0 0]';
23
24 t=0:0.2:5;
25 a1=size(t);
26 u=zeros(a1(1),a1(2));
27 figure;
28 for i=-2.0:0.5:2;
29     for j=-2:0.5:2;
30 y1=csim(u,t,sys,[i,j]');
31 plot(y1(1,:),y1(2,:));
32 end
33 end
34 set(gca(),"grid",[0.3 0.3])
35 f=get("current_figure") //Current figure handle
36 f.background=8
37 zoom_rect([-3,-3,3,3])
38 title('Phase portrait of the system with saddle
        point','fontsize',3)
39 xlabel('x1(t)','fontsize',2)
40 ylabel('x2(t)','fontsize',2)

```

---

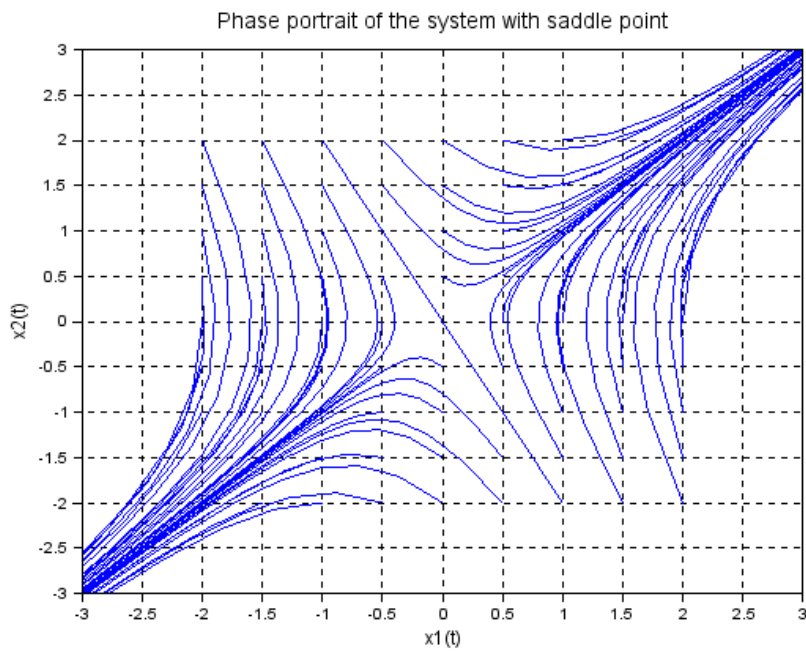


Figure 4.1: Lab04

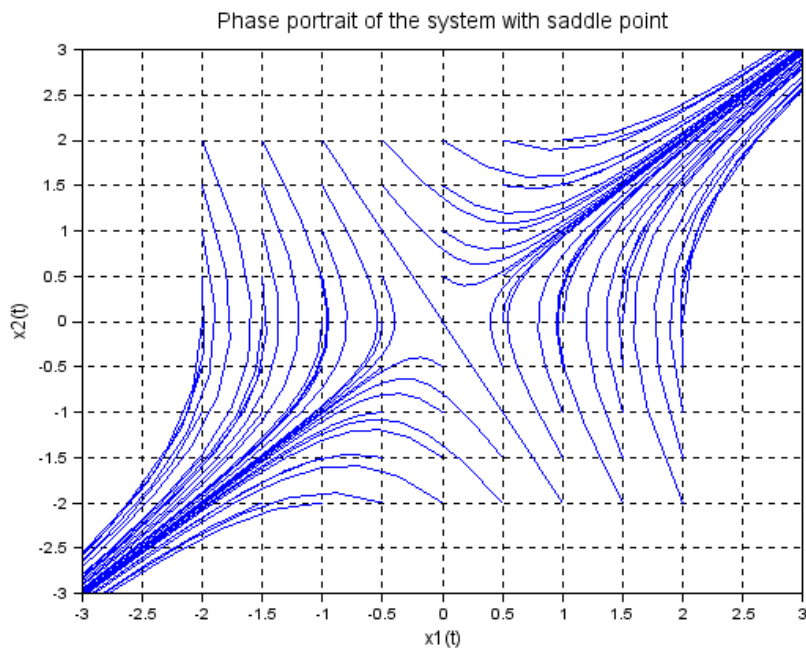


Figure 4.2: Lab04

## **Experiment: 5**

**To demonstrate limit cycles for vander pol's equation.**

This code can be downloaded from the website [www.scilab.in](http://www.scilab.in)

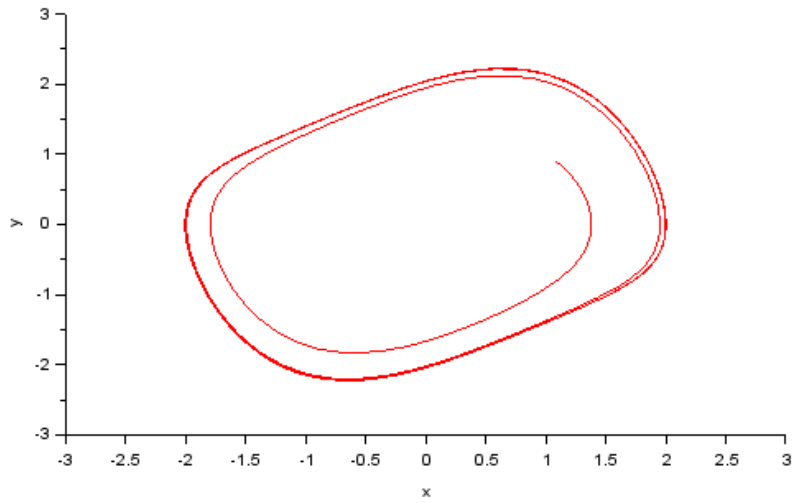


Figure 5.1: Lab05

## **Experiment: 6**

**To demonstrate the effect of  
the static nonlinearities.**

This code can be downloaded from the website [www.scilab.in](http://www.scilab.in)

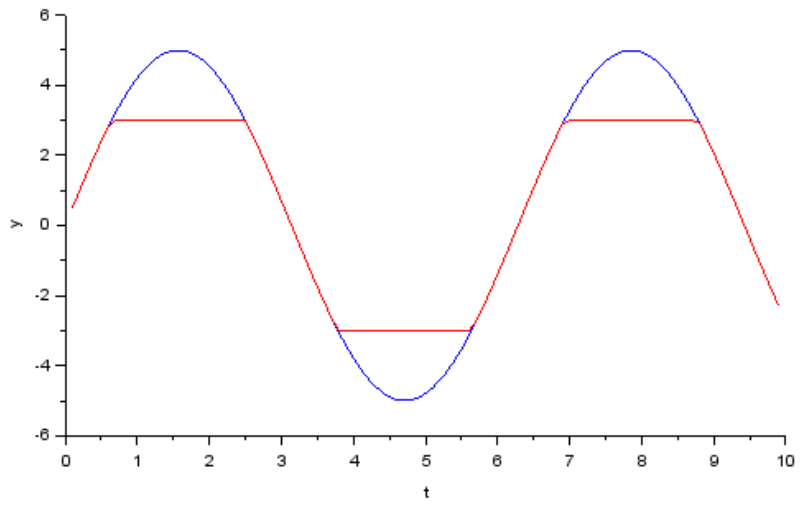


Figure 6.1: Lab06

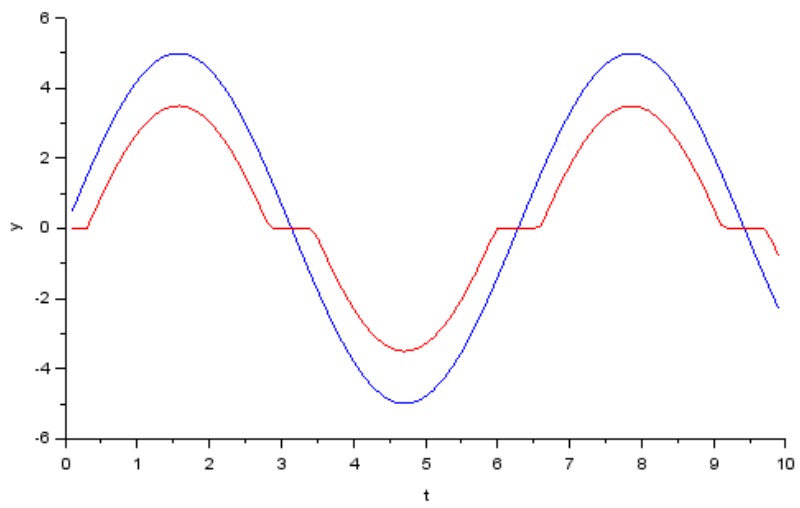


Figure 6.2: Lab06

## Experiment: 7

To demonstrate the stability of the system using Describing function.

Scilab code Solution 7.01 Lab07

```
1 //Lab. 08: To check the stability of the system
   using Describing Functions.
2
3 //scilab - 5.5.0
4 // Operating System : Windows 7, 32-bit
5
6 clc;
7 clear all;
8
9 // Frequency Bounds
10
11 wmin=1;
12 wmax=100;
13 fmin=wmin/2/%pi;
14 fmax=wmax/2/%pi;
15
16 //System Model
```

```

17
18 s=poly(0,'s');
19 g1=syslin('c',10/(0.5*s^3+1.5*s^2+s))
20
21 //Nyquist Plot
22
23 nyquist(g1,fmin,fmax)
24
25
26 // Plot of Describing Function of Relay with
    Deadzone Nonlinearity
27 x=1:0.5:10;
28 n=(2/%pi)*(asin(1 ./x)+(1 ./x) .* sqrt(1-(1 ./x)
    .^2));
29 n1=-1 ./n;
30 z=size(n1);
31 plot2d(n1,zeros(1,z(2)),2)
32
33 h=legend(['DF Contour';'System Contour'])

```

---

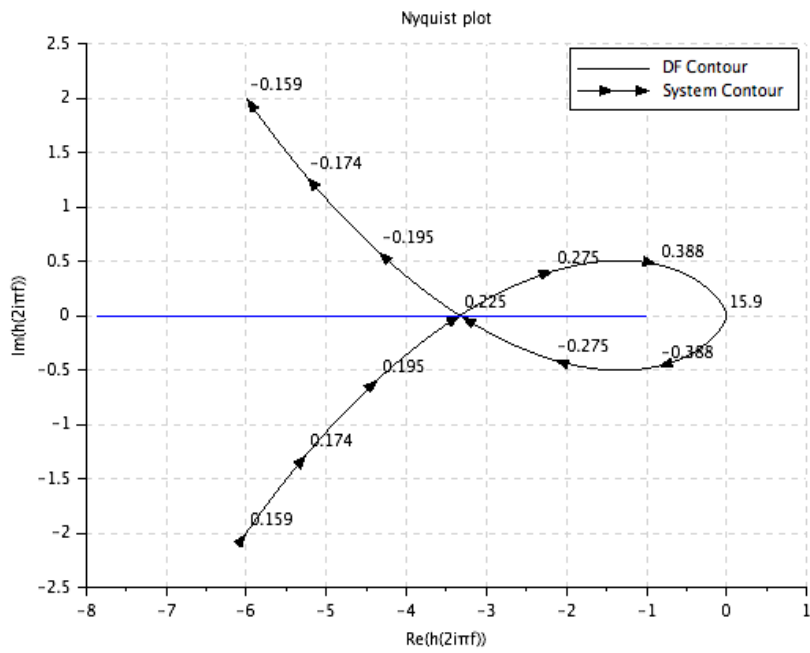


Figure 7.1: Lab07

## Experiment: 8

To demonstrate the stability of the system using Lyapunov equation.

Scilab code Solution 8.01 Lab08

```
1 //Lab. 08: To check the stability of the system
   using Lyapunov equation.
2
3 //scilab - 5.5.0
4 // Operating System : Windows 7, 32-bit
5
6 clc;
7 clear all;
8
9 //System model
10
11 a=[0 1 0;0 0 1;-2 -3 -2];
12 q=-eye(3,3);
13 p=lyap(a,q,'c');
14
15 // For a stable system matrix p should be positive
   definite for
```

```
16 //which all the principle minors or all eigen values
    of the matrix p should be positive
17 eig_val=spec(p);
18 m=length(eig_val);
19 stable=0;
20 for i=1:m;
21     if real(eig_val(i))>0 then
22         stable=stable+1;
23     end
24 end
25 if stable==m then
26     disp('The system is asymptotically stable')
27 else
28     disp('The system is unstable or critically
        stable')
29 end
```

---

## **Experiment: 9**

# **Stabilization of double integrator system using variable structure control.**

This code can be downloaded from the website [www.scilab.in](http://www.scilab.in)

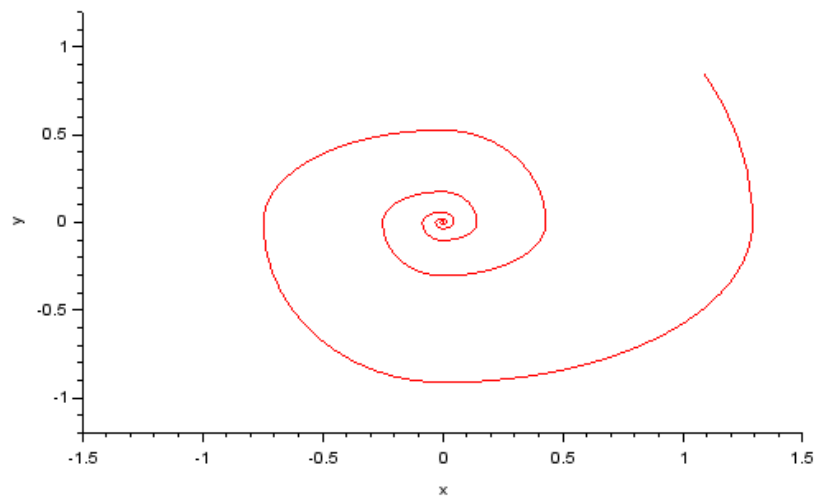


Figure 9.1: Lab09

## **Experiment: 10**

**Design the exact feedback linearizing controller for the non linear system.**

This code can be downloaded from the website [www.scilab.in](http://www.scilab.in)

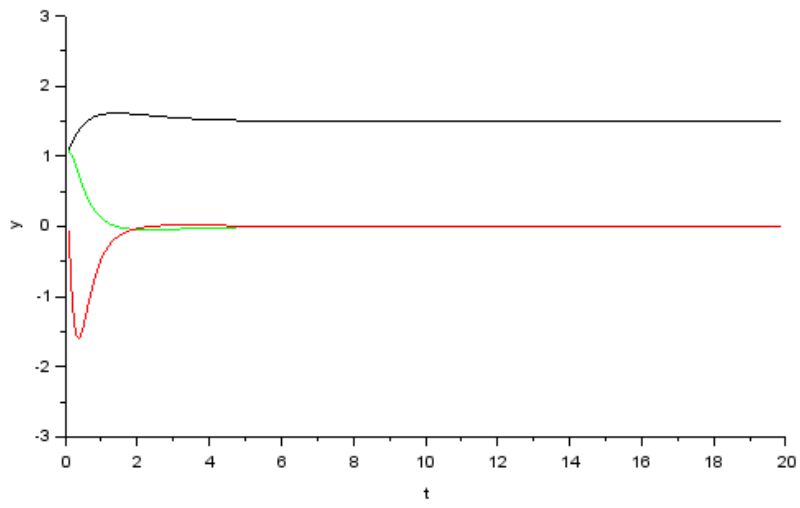


Figure 10.1: Lab10

## **Experiment: 11**

**To design sliding mode  
controller for a linear system.**

This code can be downloaded from the website [www.scilab.in](http://www.scilab.in)

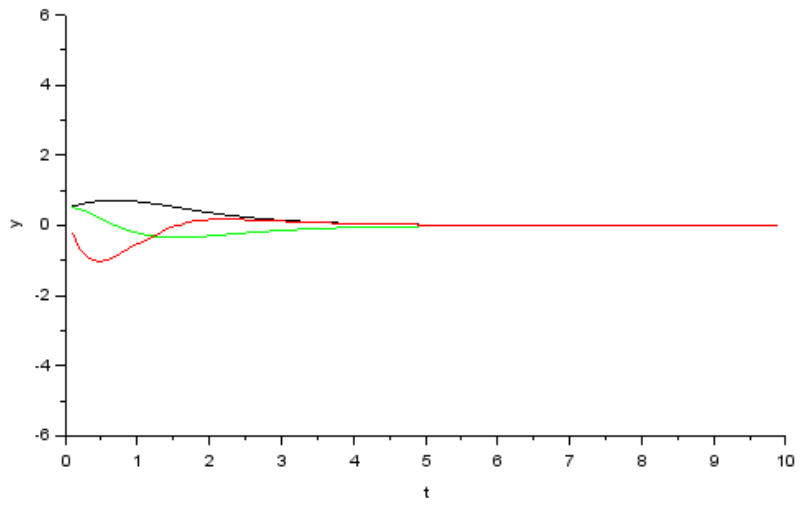


Figure 11.1: Lab11

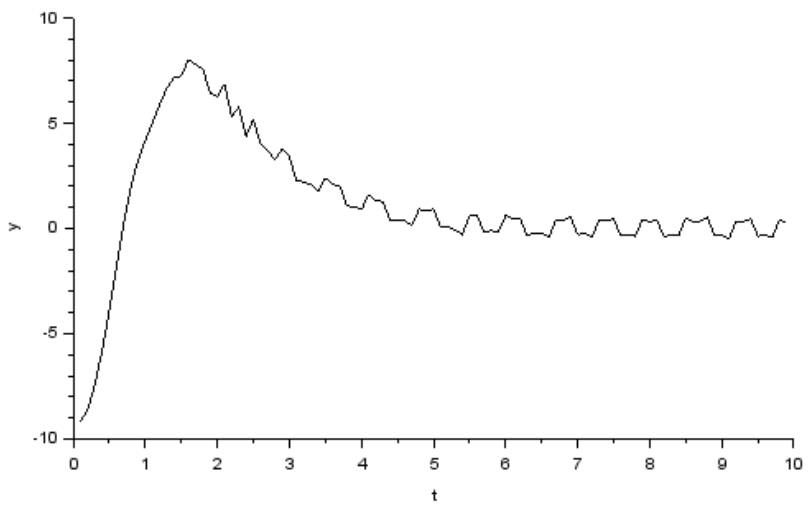


Figure 11.2: Lab11

## **Experiment: 12**

**To demonstrate model  
reference adaptive control  
system.**

This code can be downloaded from the website [www.scilab.in](http://www.scilab.in)

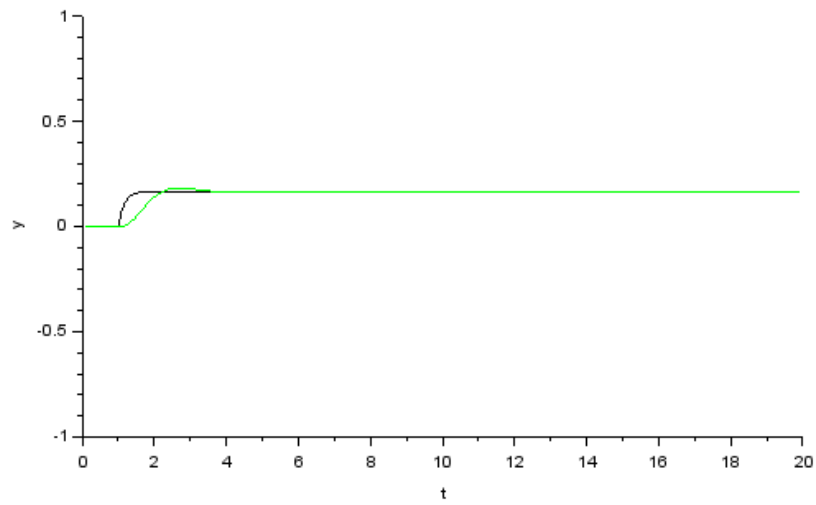


Figure 12.1: Lab12