Scilab Manual for
Advanced Control Systems
by Prof Deepti Khimani
Instrumentation Engineering
VESIT[1]

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

April 19, 2025

[1]Funded by a grant from the National Mission on Education through ICT, http://spoken-tutorial.org/NMEICT-Intro. This Scilab Manual and Scilab codes written in it can be downloaded from the "Migrated Labs" section at the website http://scilab.in

Contents

List of Experiments

List of Figures

Experiment: 1

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

Scilab code Solution 1.01 Lab01

```
//Lab. 01: To plot the phase portrait of systems
   having stable and unstable nodes.

//scilab - 5.5.0
// Operating System : Windows 7, 32-bit

clc;
clear all;
clf;

//System transfer function
s=poly(0,'s');
g=1/(s^2+3*s+2);

//Draw pole zero map of the system
plzr(g);
title('Pole-zero map of the system with real stable
```

```
eigen values','fontsize',3)
//Convert the given transfer function into state space form
sys=tf2ss(g);

//Plot of system phase trajectory
sys.c=[1,0;0,1];
sys.d=[0 0]';

t=0:0.2:10;
a1=size(t);
u=zeros(a1(1),a1(2));
figure
for i=-2.0:0.5:2;
    for j=-2:0.5:2;
        y1=csim(u,t,sys,[i,j]');
        plot(y1(1,:),y1(2,:));
    end
end

set(gca(),"grid",[0.3 0.3])
title('Phase portrait of the system with stable node','fontsize',3)
xlabel('x1(t)','fontsize',2)
ylabel('x2(t)','fontsize',2)
f=get("current_figure") //Current figure handle
f.background=8
//System transfer function
s=poly(0,'s');
g=1/(s^2-3*s+2);

// Draw pole zero map of the system
figure;
plzr(g);
title('Pole-zero map of the system with real unstable eigen values','fontsize',3)
f=get("current_figure") //Current figure handle
f.background=8
```

```
//Convert the given transfer function into state space form
sys=tf2ss(g);

//Plot of system phase trajectory
sys.c=[1,0;0,1];
sys.d=[0 0]';

a1=size(t);
u=zeros(a1(1),a1(2));
figure
for i=-2.0:0.5:2;
    for j=-2:0.5:2;
        y1=csim(u,t,sys,[i,j]');
        plot(y1(1,:),y1(2,:));
    end
end
set(gca(),"grid",[0.3 0.3])
f=get("current_figure") //Current figure handle
f.background=8
zoom_rect([-5,-5,5,5])
title('Phase portrait of the system with unstable node','fontsize',3)
xlabel('x1(t)','fontsize',2)
ylabel('x2(t)','fontsize',2)
```

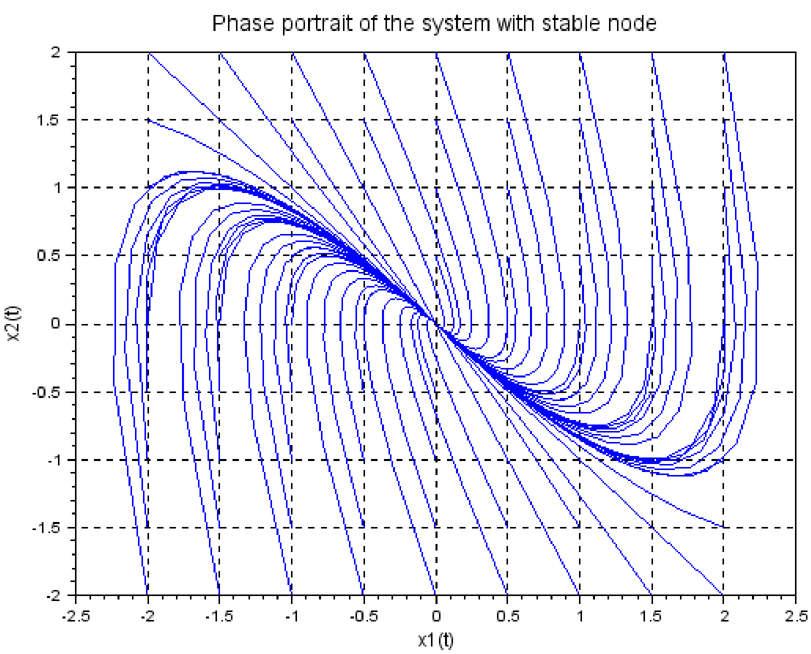

Figure 1.1: Lab01

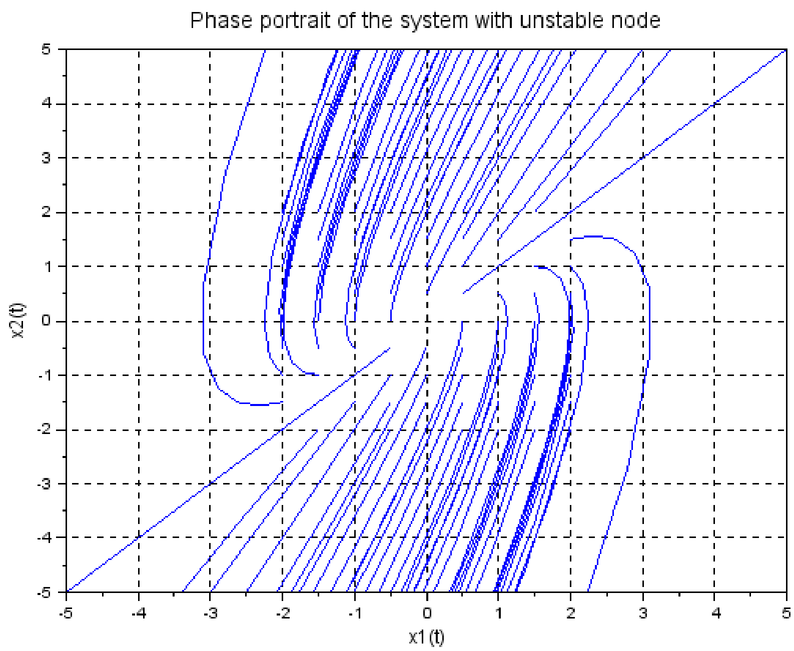

Figure 1.2: Lab01

Experiment: 2

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

Scilab code Solution 2.1 Lab02

```
//Lab. 02: To plot the phase portrait of systems
   having stable and unstable focus point.

//scilab - 5.5.0
// Operating System : Windows 7, 32-bit

clc;
clear all;
clf;

//System transfer function
s=poly(0,'s');
g=1/(s^2+s+1);

// Draw pole zero map of the system
plzr(g);
title('Pole-zero map of the system with stable
```

```
underdamped eigen values','fontsize',3)
//Convert the given transfer function into state space form
sys=tf2ss(g);

//Plot of system phase trajectory
sys.c=[1,0;0,1];
sys.d=[0 0]';

t=0:0.2:10;
a1=size(t);
u=zeros(a1(1),a1(2));
figure
for i=-2.0:0.5:2;
    for j=-2:0.5:2;
y1=csim(u,t,sys,[i,j]');
plot(y1(1,:),y1(2,:));
end
end
set(gca(),"grid",[0.3 0.3])
f=get("current_figure") //Current figure handle
f.background=8
title('Phase portrait of the system with stable focus','fontsize',3)
xlabel('x1(t)','fontsize',2)
ylabel('x2(t)','fontsize',2)

//System transfer function
s=poly(0,'s');
g=1/(s^2-s+1);

//Convert the given transfer function into state space form
sys=tf2ss(g);

// Draw pole zero map of the system
figure;
plzr(g);
```

```
f=get("current_figure") //Current figure handle
f.background=8
title('Pole-zero map of the system with negatively damped eigen values','fontsize',3)
//Plot of system phase trajectory
sys.c=[1,0;0,1];
sys.d=[0 0]';

a1=size(t);
u=zeros(a1(1),a1(2));
figure
for i=-2.0:0.5:2;
    for j=-2:0.5:2;
y1=csim(u,t,sys,[i,j]');
plot(y1(1,:),y1(2,:));
end
end
set(gca(),"grid",[0.3 0.3])
f=get("current_figure") //Current figure handle
f.background=8
zoom_rect([-5,-5,5,5])
title('Phase portrait of the system with unstable focus ','fontsize',3)
xlabel('x1(t)','fontsize',2)
ylabel('x2(t)','fontsize',2)
```

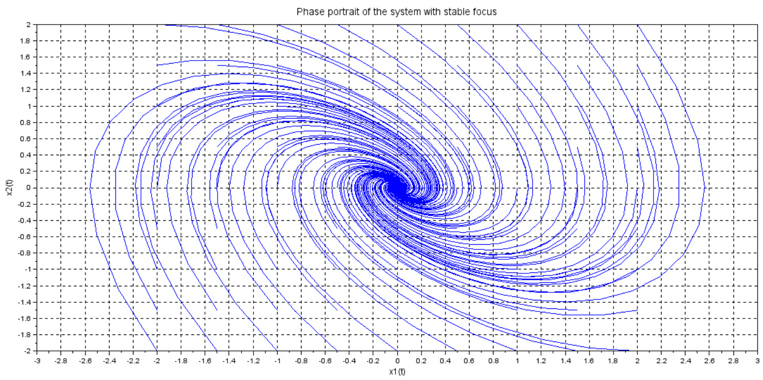

Figure 2.1: Lab02

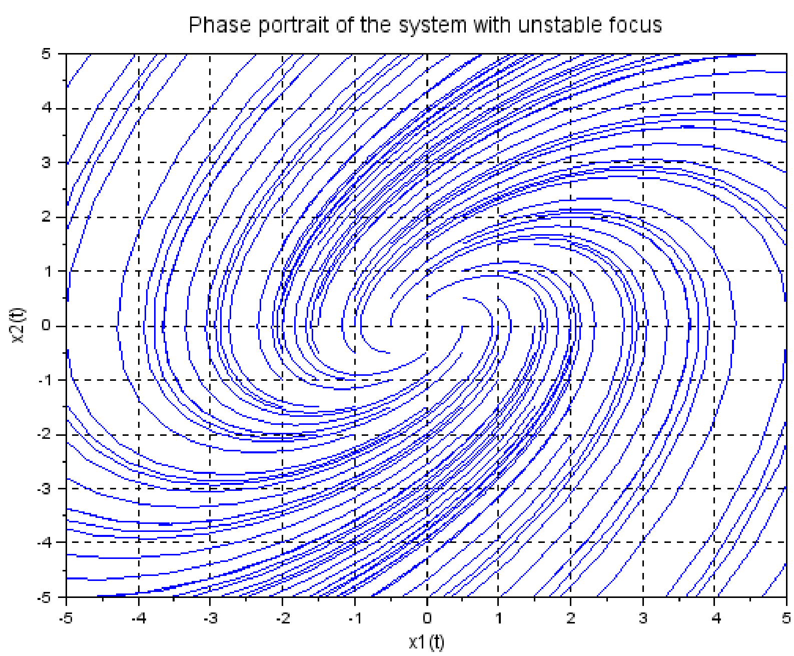

Figure 2.2: Lab02

Experiment: 3

To plot the phase portrait of systems having vortex point.

Scilab code Solution 3.01 Lab03

```
//Lab. 03: To plot the phase portrait of systems having vortex point.

//scilab-5.5.0
// Operating System : Windows 7, 32-bit

clc;
clear all;
clf;

//System transfer function
s=poly(0,'s');
g=1/(s^2+4);

// Draw pole zero map of the system
plzr(g);
title('Pole-zero map of the system with critically damped eigen values','fontsize',3)
//Convert the given transfer function into state
```

```
    space form
sys=tf2ss(g);

//Plot of system phase trajectory
sys.c=[1,0;0,1];
sys.d=[0 0]';

t=0:0.2:10;
a1=size(t);
u=zeros(a1(1),a1(2));
figure
for i=-2.0:0.5:2;
    for j=-2:0.5:2;
y1=csim(u,t,sys,[i,j]');
plot(y1(1,:),y1(2,:));
end
end
set(gca(),"grid",[0.3 0.3])
f=get("current_figure") //Current figure handle
f.background=8
title('Phase portrait of the system with vortex
    point','fontsize',3)
xlabel('x1(t)','fontsize',2)
ylabel('x2(t)','fontsize',2)
```

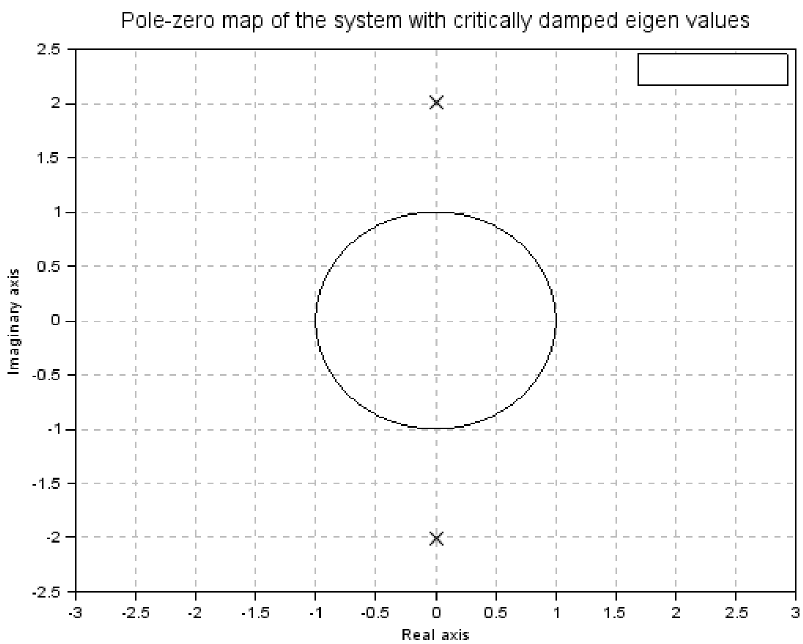

Figure 3.1: Lab03

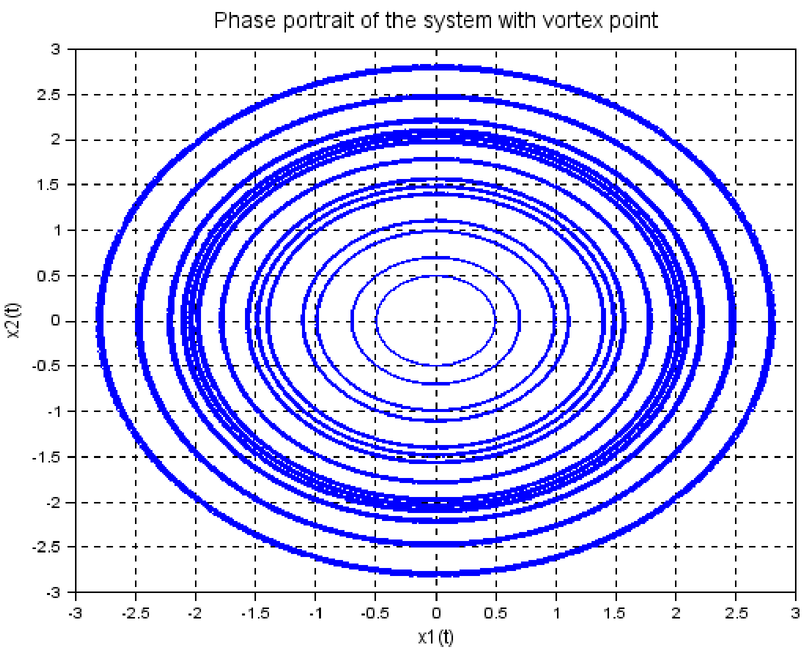

Figure 3.2: Lab03

Experiment: 4

To plot the phase portrait of systems having saddle point.

Scilab code Solution 4.01 Lab04

```
//Lab. 04: To plot the Phase portrait of systems
   having saddle point.

//scilab - 5.5.0
// Operating System : Windows 7, 32-bit

clc;
clear all;
clf;

//System transfer function
s=poly(0,'s');
g=1/(s^2+1*s-2);

//Convert the given transfer function into state
   space form
sys=tf2ss(g);

// Draw pole zero map of the system
```

```
plzr(sys);
title('Pole-zero map of the system with real stable
    and unstable eigen values','fontsize',3)
//Plot of system phase trajectory
sys.c=[1,0;0,1];
sys.d=[0 0]';

t=0:0.2:5;
a1=size(t);
u=zeros(a1(1),a1(2));
figure;
for i=-2.0:0.5:2;
    for j=-2:0.5:2;
y1=csim(u,t,sys,[i,j]');
plot(y1(1,:),y1(2,:));
end
end
set(gca(),"grid",[0.3 0.3])
f=get("current_figure") //Current figure handle
f.background=8
zoom_rect([-3,-3,3,3])
title('Phase portrait of the system with saddle
    point','fontsize',3)
xlabel('x1(t)','fontsize',2)
ylabel('x2(t)','fontsize',2)
```

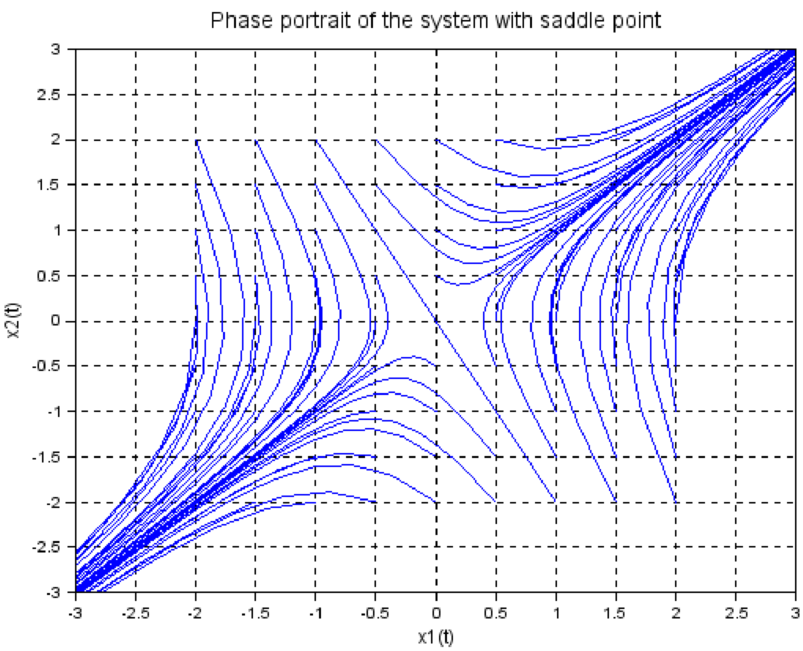

Figure 4.1: Lab04

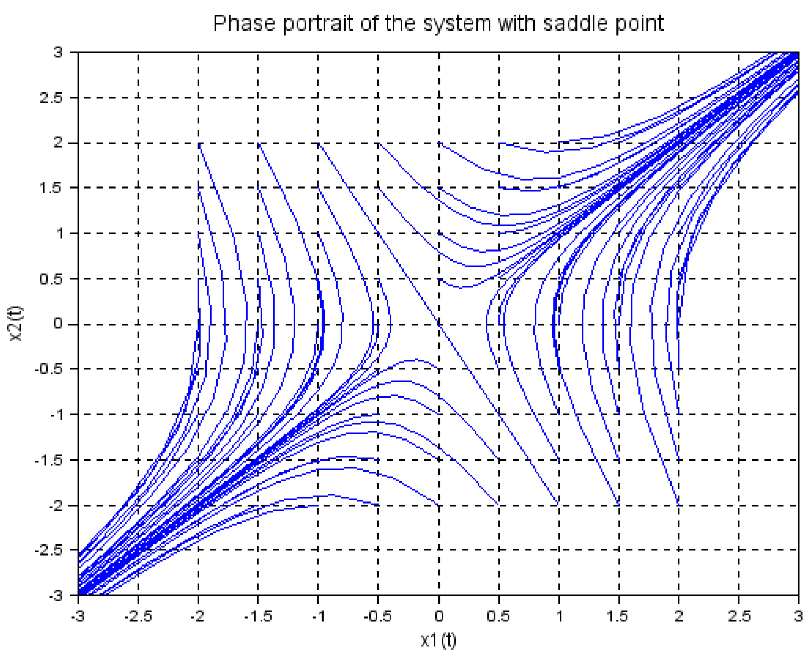

Figure 4.2: Lab04

Experiment: 5

To demonstrate limit cycles for vander pol's equation.

This code can be downloaded from the website wwww.scilab.in

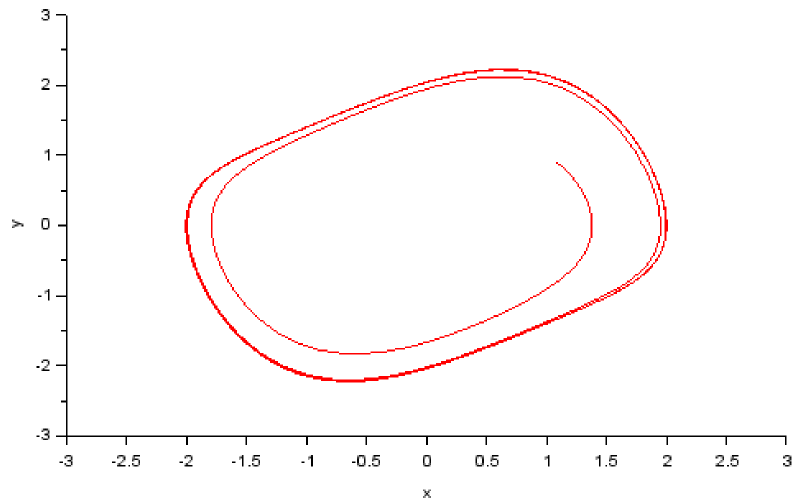

Figure 5.1: Lab05

Experiment: 6

To demonstrate the effect of the static nonlinearities.

This code can be downloaded from the website wwww.scilab.in

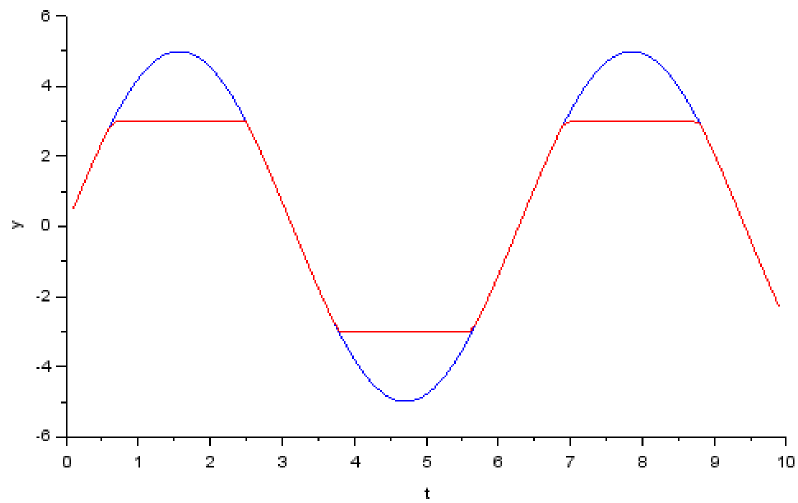

Figure 6.1: Lab06

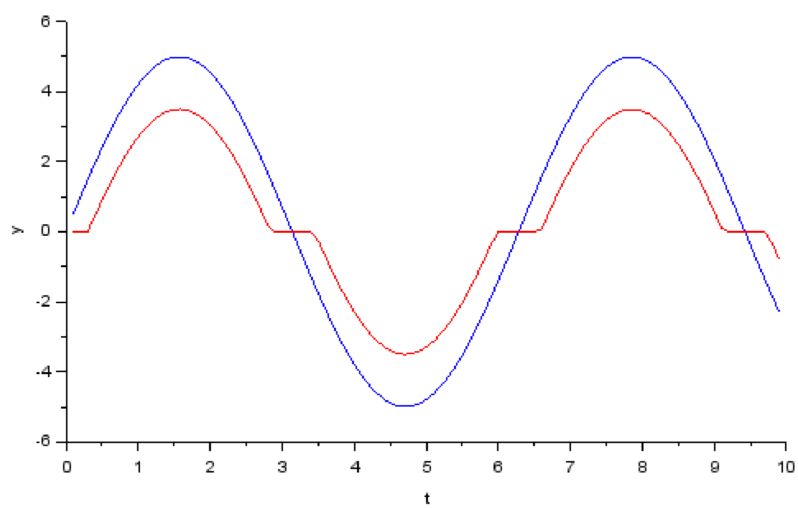

Figure 6.2: Lab06

Experiment: 7

To demonstrate the stability of the system using Describing function.

Scilab code Solution 7.01 Lab07

```
//Lab. 08: To check the stability of the system
   using Describing Functions.

//scilab - 5.5.0
// Operating System : Windows 7, 32-bit

clc;
clear all;

// Frequency Bounds

wmin=1;
wmax=100;
fmin=wmin/2/%pi;
fmax=wmax/2/%pi;

//System Model
```

```
s=poly(0,'s');
g1=syslin('c',10/(0.5*s^3+1.5*s^2+s))

//Nyquist Plot

nyquist(g1,fmin,fmax)


// Plot of Describing Function of Relay with
   Deadzone Nonlinearity
x=1:0.5:10;
n=(2/%pi)*(asin(1 ./x)+(1 ./x) .* sqrt(1-(1 ./x)
   .^2));
n1=-1 ./n;
z=size(n1);
plot2d(n1,zeros(1,z(2)),2)

h=legend (['DF Contour';'System Contour'])
```

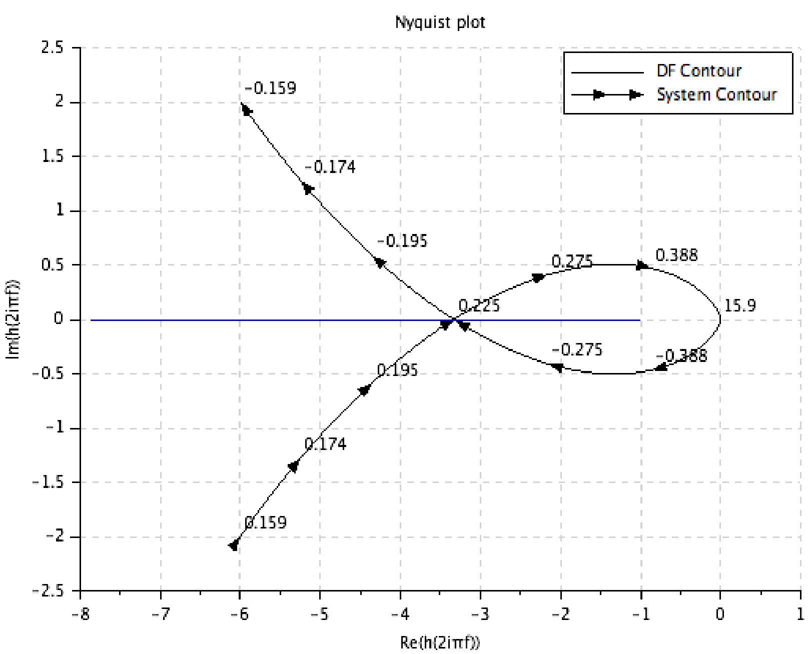

Figure 7.1: Lab07

Experiment: 8

To demonstrate the stability of the system using Lyapunov equation.

Scilab code Solution 8.01 Lab08

```
//Lab. 08: To check the stability of the system
   using Lyapunov equation.

//scilab - 5.5.0
// Operating System : Windows 7, 32-bit

clc;
clear all;

//System model

a=[0 1 0;0 0 1;-2 -3 -2];
q=-eye(3,3);
p=lyap(a,q,'c');

// For a stable system matrix p should be positive
   definite for
```

```
//which all the principle minors or all eigen values
    of the matrix p should be positive
eig_val=spec(p);
m=length(eig_val);
stable=0;
for i=1:m;
    if real(eig_val(i))>0 then
        stable=stable+1;
        end
end
if stable==m then
    disp('The system is asymptotically stable')
else
    disp('The system is unstable or critically
       stable')
end
```

Experiment: 9

Stabilization of double integrator system using variable structure control.

This code can be downloaded from the website wwww.scilab.in

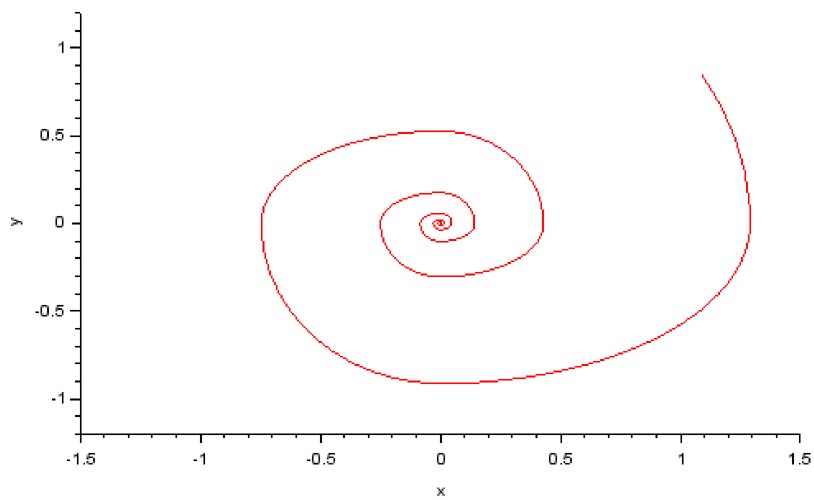

Figure 9.1: Lab09

Experiment: 10

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

This code can be downloaded from the website wwww.scilab.in

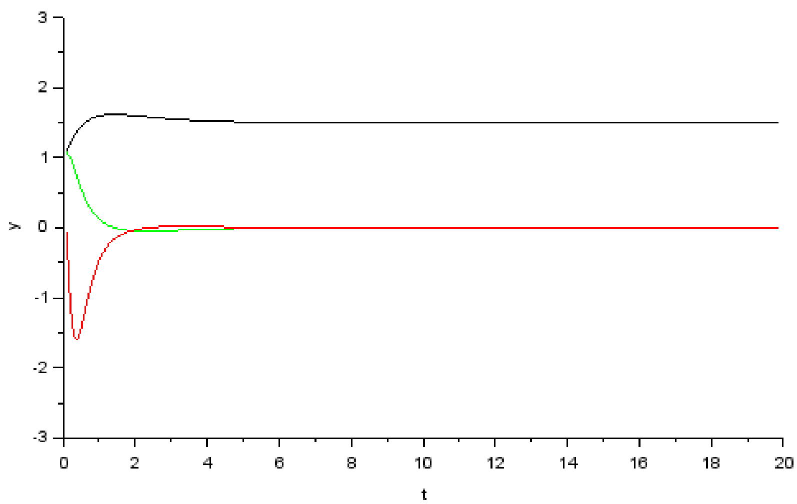

Figure 10.1: Lab10

Experiment: 11

To design sliding mode controller for a linear system.

This code can be downloaded from the website wwww.scilab.in

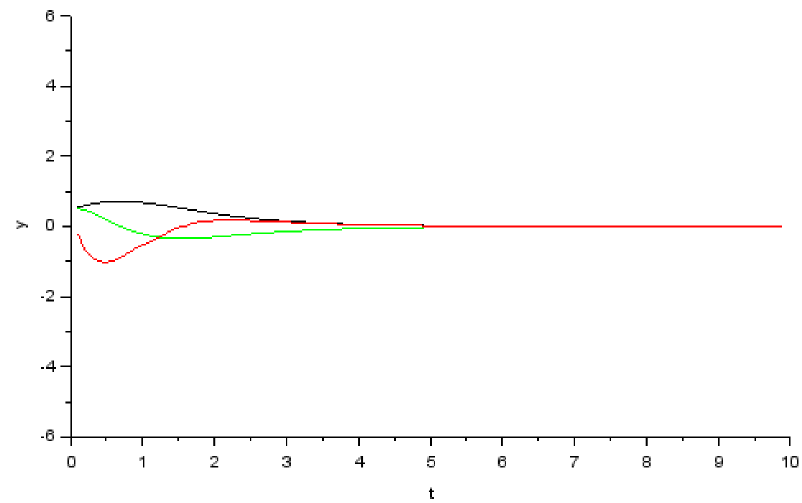

Figure 11.1: Lab11

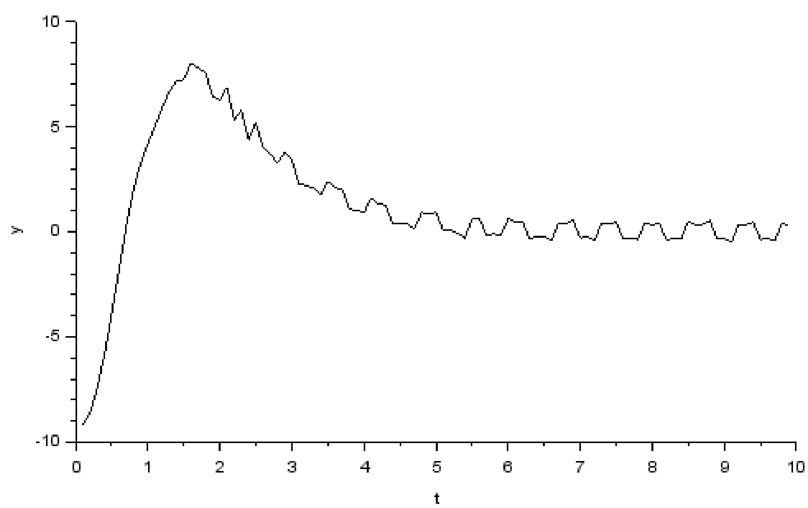

Figure 11.2: Lab11

Experiment: 12

To demonstrate model reference adaptive control system.

This code can be downloaded from the website wwww.scilab.in

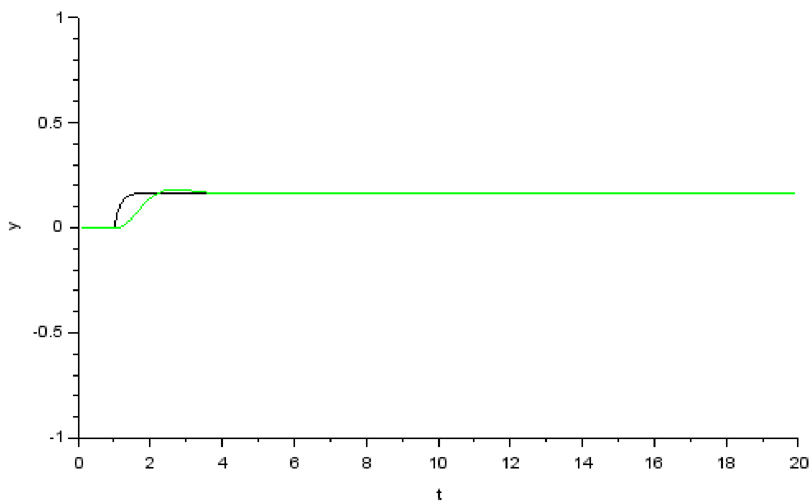

Figure 12.1: Lab12