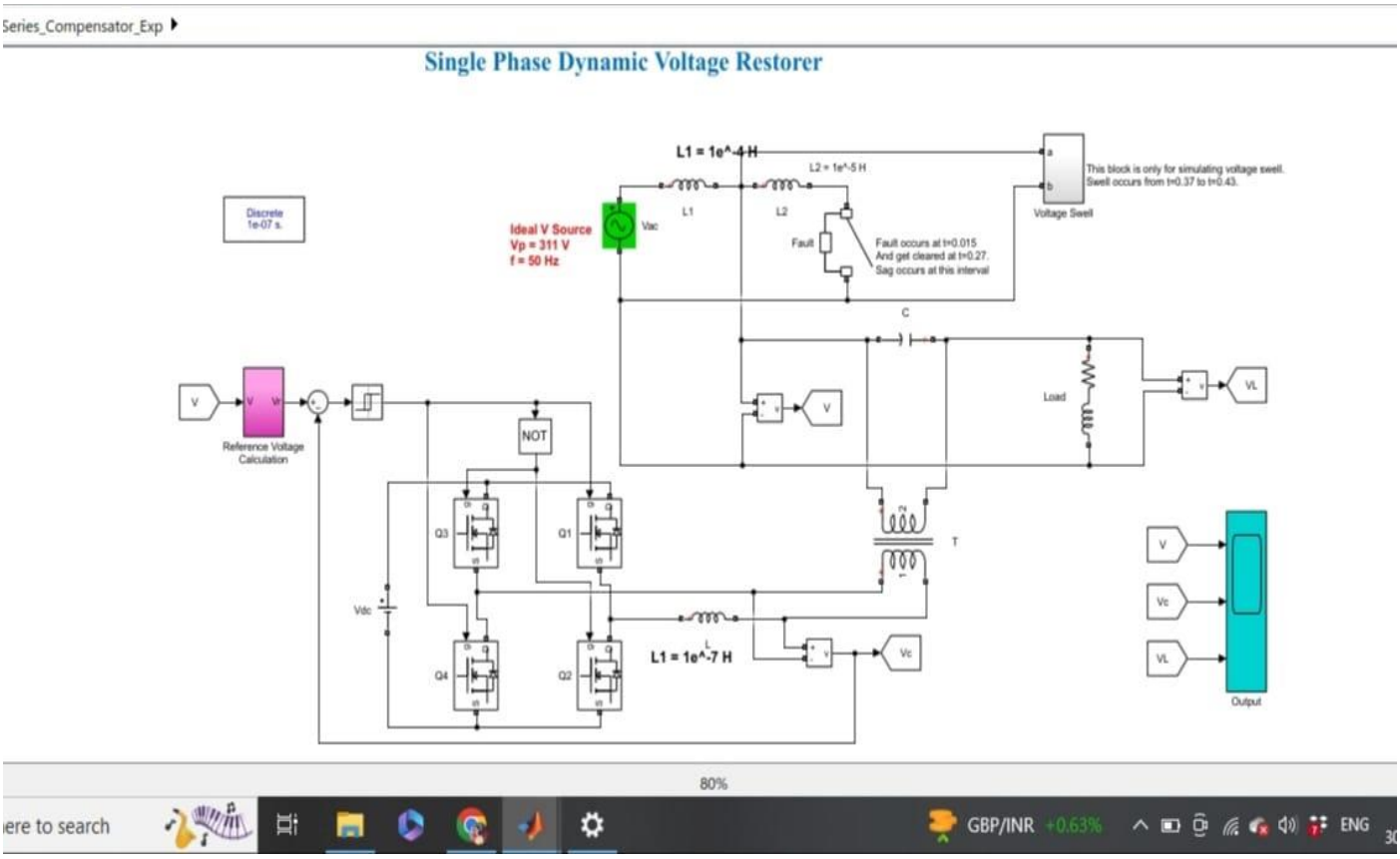
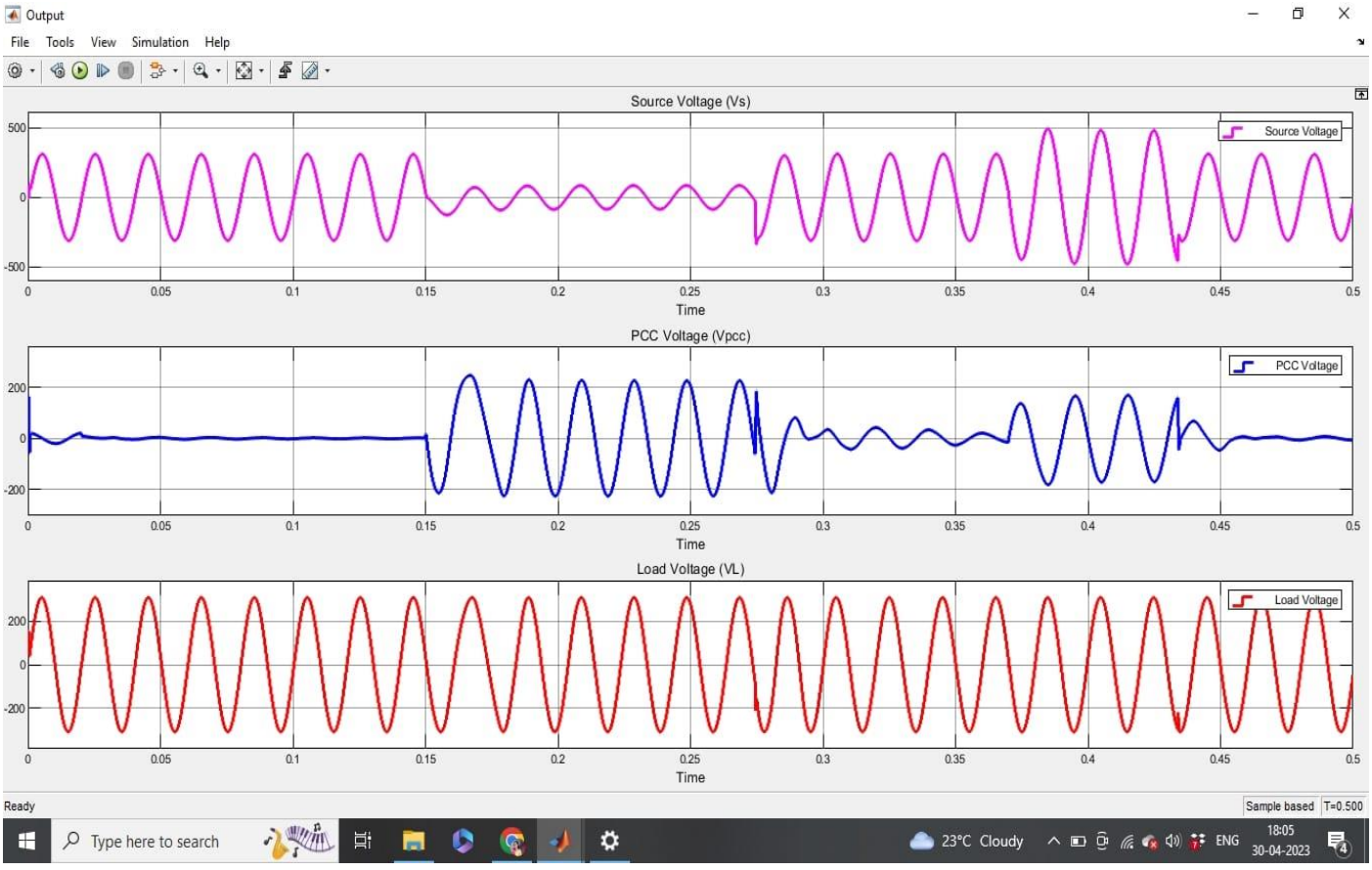
SIMULINK MODEL OF SINGLE PHASE DYNAMIC VOLTAGE RESTORER:



RESULT:

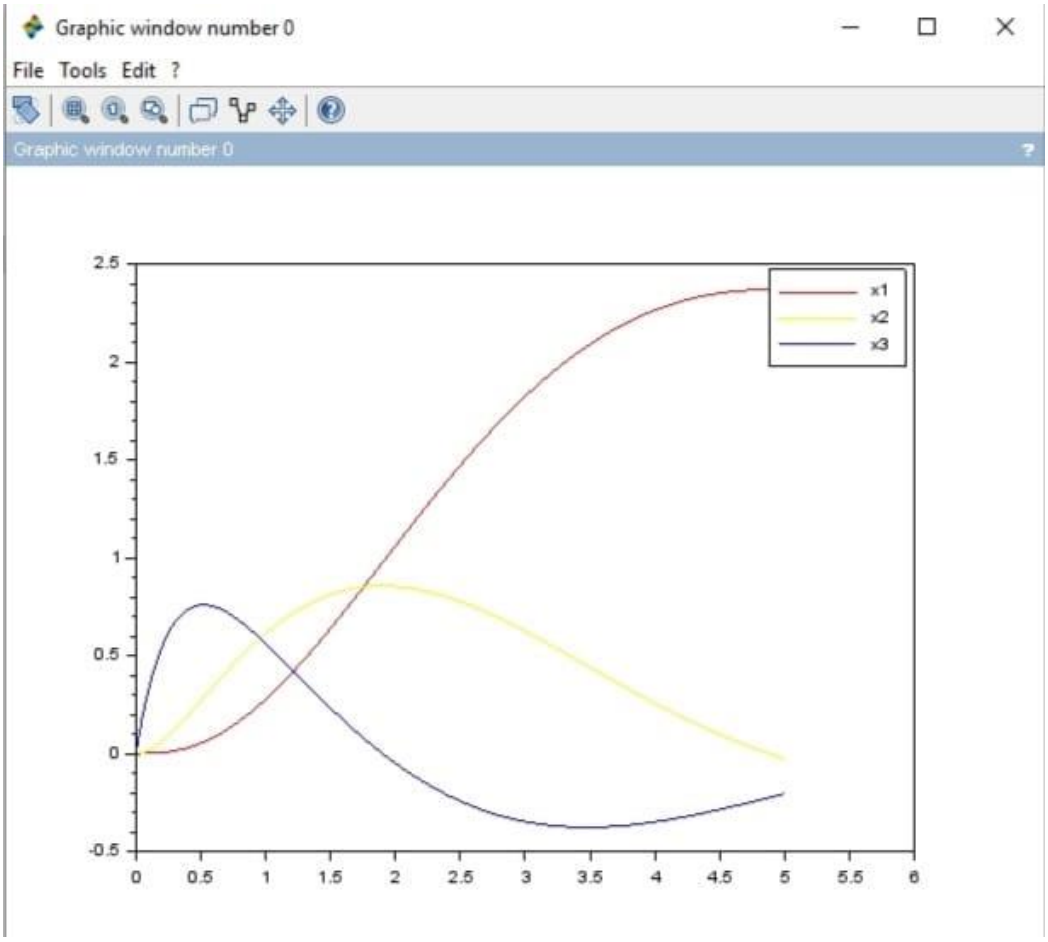


EXPERIMENT 1

CODE 1:

```
clc
clear
t_int=0;
t_fi=5;
h=0.001;
x1(1)=0;
x2(1)=0;
x3(1)=0;
t(1)=0;
A=[0,1,0;0,0,1;-2,-3,-4];
disp(spec(A))
n=(t_fi - t_int)/h
for i=1:n
    x1(i+1)= x1(i)+h*x2(i);
    x2(i+1)=x2(i)+h*x3(i);
    x3(i+1)= x3(i)+h*(-2*x1(i)-3*x2(i)-4*x3(i)+4);
    t(i+1)=t(i)+h;
end
plot(t, x1, 'r')
plot(t, x2, 'y')
plot(t, x3, 'b')
legend('x1','x2','x3')
```

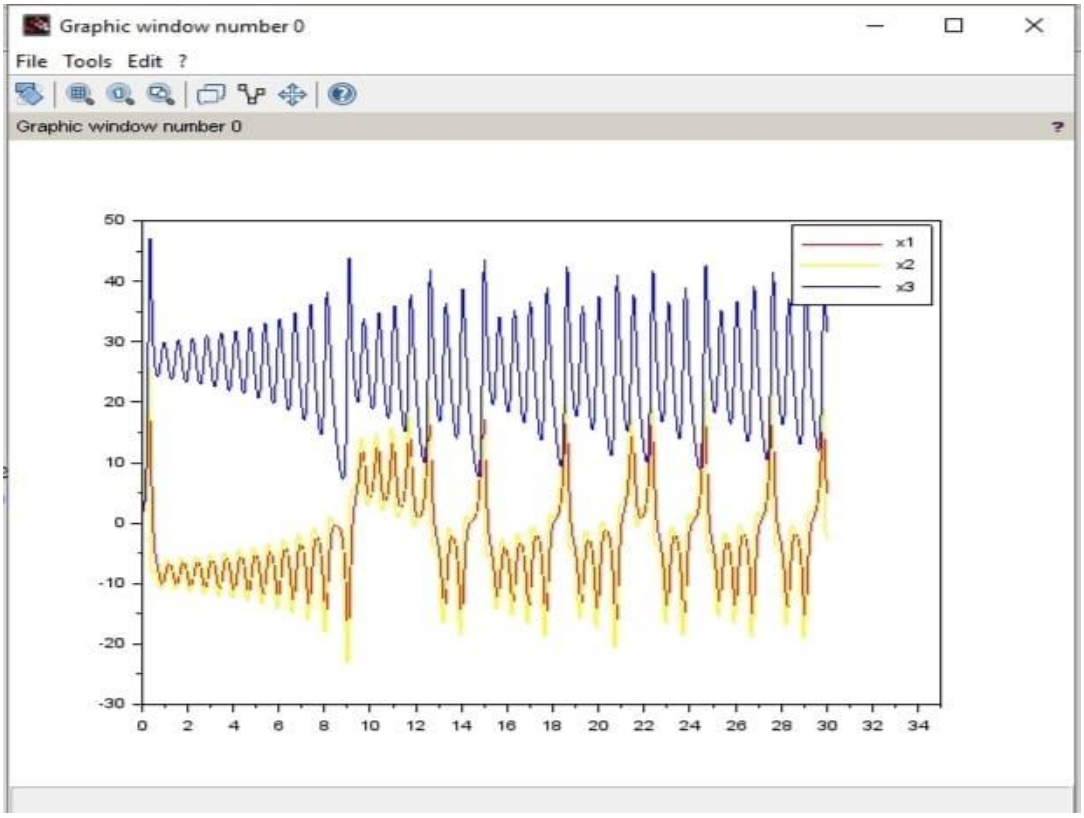
RESULT:



CODE 2:

```
clc
clear

ti=0;
tf=30;
h=0.001;
U = 1;
n=(tf-ti)/h;

x1(1)=1;
x2(1)=2;
x3(1)=3;
t(1)=0

for i=1:n
    x1(i+1)=x1(i)+h*(-10*x1(i)+10*x2(i));
    x2(i+1)=x2(i)+h*(28*x1(i)-x1(i)*x3(i) -x2(i));
    x3(i+1)=x3(i)+h*(x1(i)*x2(i)-(8/3)*x3(i));
    t(i+1)= t(i) + h
end
plot(t, x1, 'r')
plot(t, x2, 'y')
plot(t, x3, 'b')
legend('x1','x2','x3')

a = [0 1 0; 0 0 1; -2 -3 -4];
disp(spec(a));
```

RESULT:



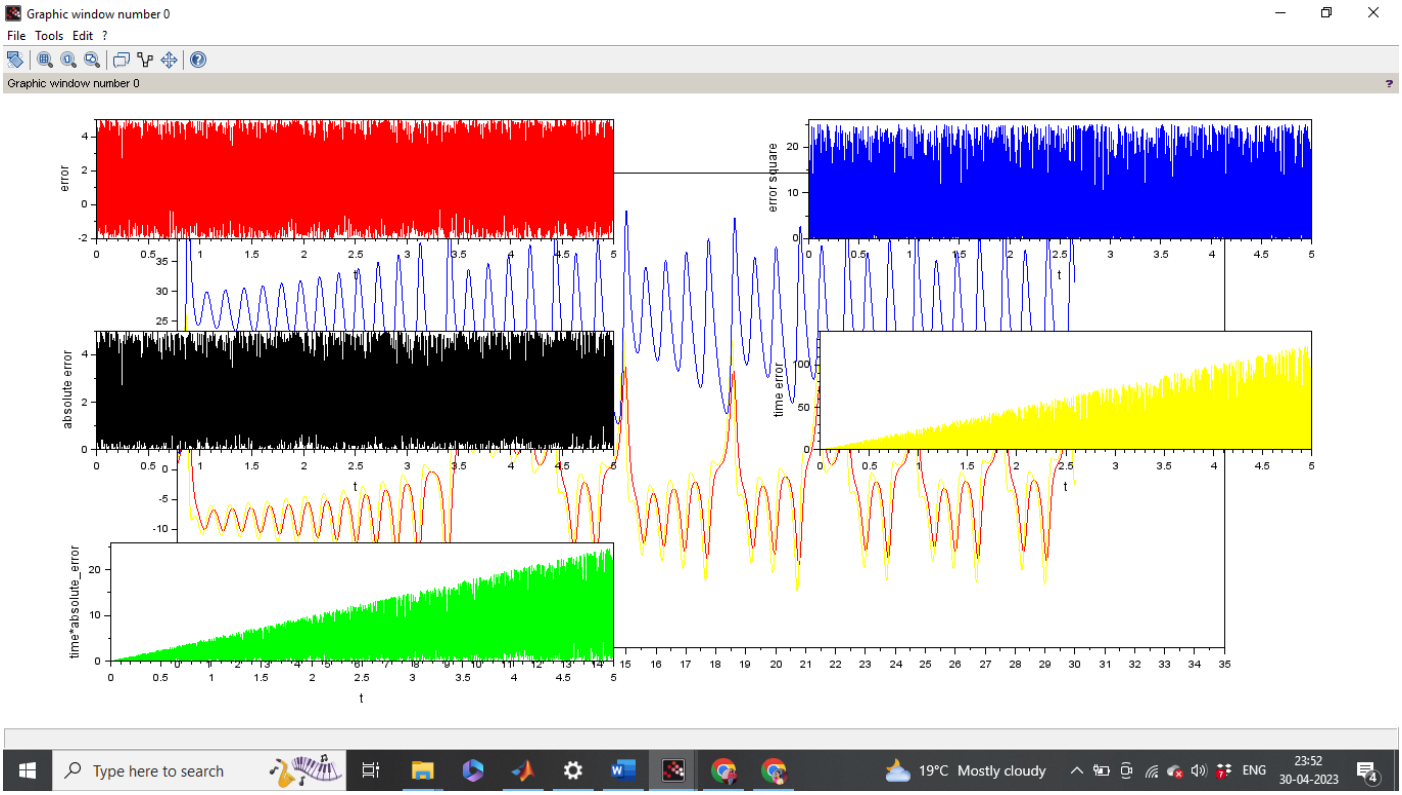
MACHINE LEARNING AND ROBOTICS LABORATORY (EE18218)

EXPERIMENT 2

CODE:

```
e_lower=-2;
e_upper=5;
t_lower=0;
t_upper=5;
h=0.001;
n=(t_upper-t_lower)/h;
e=e_lower+(e_upper-e_lower)*rand(1, 5000);
disp(size(e));
t=0.001:0.001:5
disp(size(t));

subplot(3,2,1)
plot(t,e,"r");
xlabel("t");
ylabel("error");

subplot(3,2,2)
plot(t,e.^2,"b");
xlabel("t");
ylabel("error square");

subplot(3,2,3)
plot(t,abs(e),"k");
xlabel("t");
ylabel("absolute error");

subplot(3,2,4)
plot(t,t.*e^2,"y");
xlabel("t");
ylabel("time error");

subplot(3,2,5)
plot(t,t.*abs(e),"g");
xlabel("t");
ylabel("time*absolute_error");

//error
//ISE
ISE=inttrap(t,e^2);
mprintf("integral square error is:%f\n",ISE);

//IAE
IAE=inttrap(t,abs(e));
mprintf("integral square error is:%f\n",IAE);

//ITSE
ITSE=inttrap(t,t.*e^2);
mprintf("integral square error is:%f\n",ITSE);

//ITAE
ITAE=inttrap(t,t.*abs(e));
mprintf("integral square error is:%f\n",ITAE);
```

RESULT:


Graphic window number 0
File Tools Edit ?
Graphic window number 0
error
error square
absolute error
time error
time*absolute_error
t
Type here to search
19°C Mostly cloudy
ENG
23:52
30-04-2023

MACHINE LEARNING AND ROBOTICS LABORATORY (EE18218)

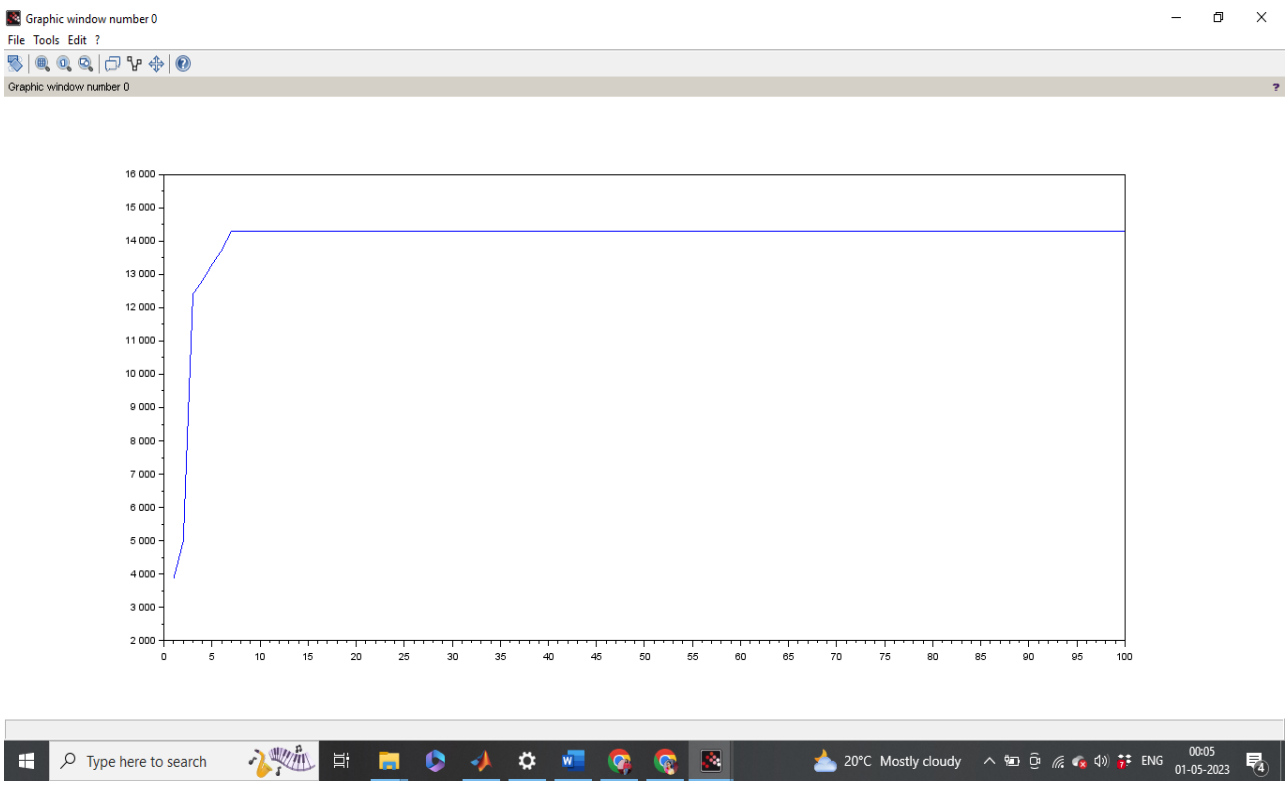
EXPERIMENT 3

CODE: (PSO 2 Variables)

```
clc
clear
dimensions = 2;
e_lower = -5;
e_upper = 5;

N = 15;   /*We will take 15 particles*/
w = 0.5;
c1 = 0.7;
c2 = 0.8;

no_of_iter = 100;

P = e_lower + (e_upper - e_lower)*rand(N, dimensions);

V = zeros(N, dimensions);  // Velocity

obj = zeros(N, 1);
pb_val = zeros(N, 1);
pb_particle = zeros(N, 2);
gb_val_max = -%inf;
gb_val_min = %inf;
// finding objective
for i = 1:N
    summ = 4*P(i, 1)^2 - 2.1*P(i, 1)^5 + (1/3)*P(i, 1)^6 + P(i, 1)*P(i, 2) -
4*P(i, 2)^2+ 4*P(i, 2)^4;

    obj(i, 1) = summ;
    pb_val(i, 1) = obj(i, 1);
    pb_particle(i, :) = P(i, :);

    if obj(i,1) > gb_val_max
        gb_val_max = obj(i, 1);
        gb_particle_max(1, :) = P(i, :)
    end


end

for k = 1:no_of_iter
    for i = 1:N
        for j = 1:dimensions
            V(i, j) = w*V(i, j) + c1*rand(1, 1)*(pb_particle(i, j) - P(i, j))+
c2*rand(1, 1)*(gb_particle_max(1, j) - P(i, j));
            P(i, j) = P(i, j) + V(i, j)

            if P(i,j) >5
                P(i,j) = 5
            end

            if P(i,j) < -5
                P(i,j) = -5
            end
```

```
        end
    end

    // Finding objective function to  know personal best and global best

    for i = 1:N
        summ = 4*P(i, 1)^2 - 2.1*P(i, 1)^5 + (1/3)*P(i, 1)^6 + P(i, 1)*P(i, 2)
- 4*P(i, 2)^2+ 4*P(i, 2)^4;
        obj(i, 1) = summ;

        // Finding the personal best and global best

        if obj(i,1) > pb_val(i, 1)
            pb_val(i, 1) = obj(i, 1);
            pb_particle(i, :) = P(i, :);
        end



        if obj(i,1) > gb_val_max
            gb_val_max = obj(i, 1);
            gb_particle_max(1, :) = P(i, :)
        end

    end

    gb_con(k, 1) = gb_val_max

end

plot(1:1:no_of_iter, gb_con)
```

RESULT:

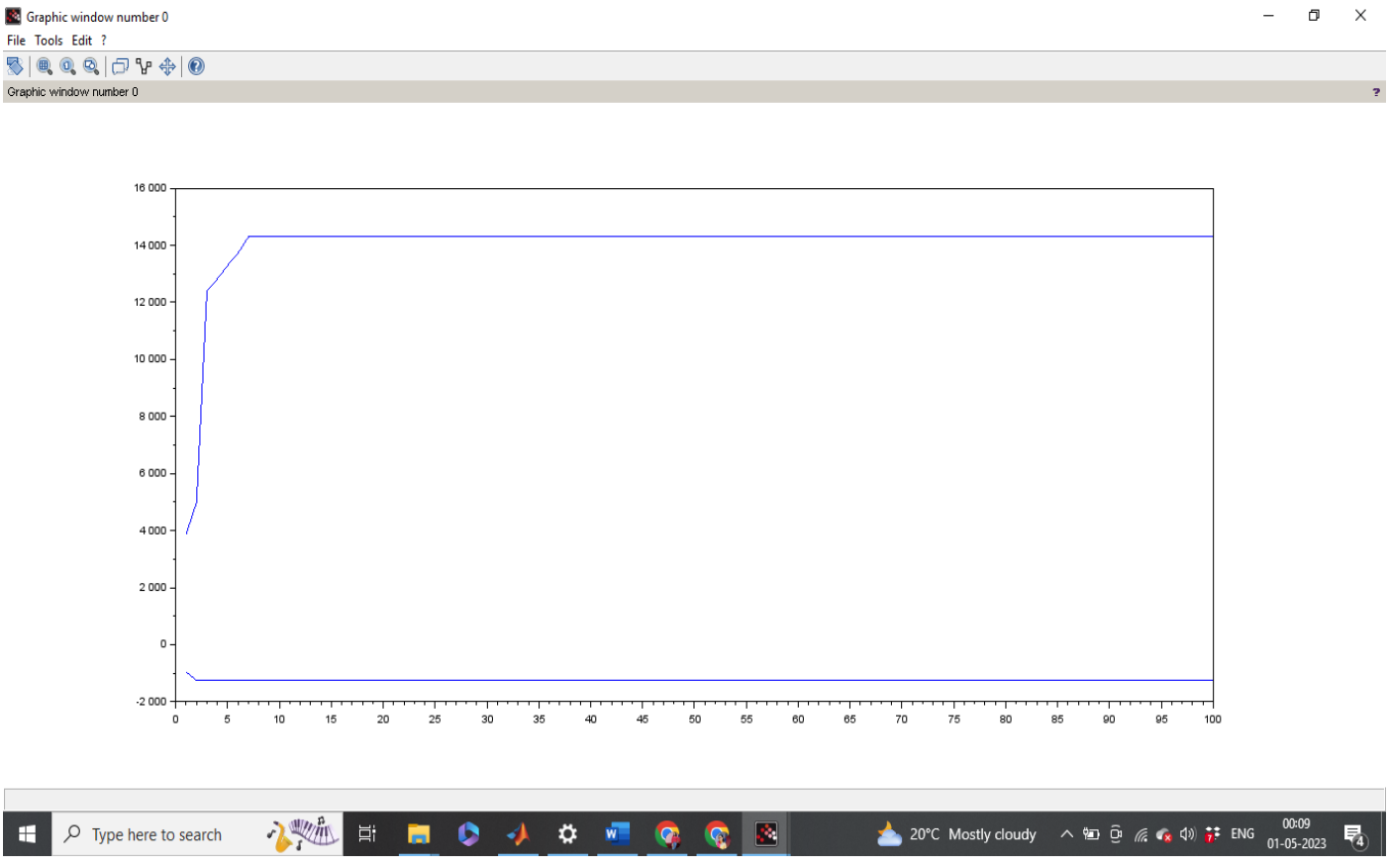


CODE: (PSO 2 Variables Minimization)

```
clc
clear
dimensions = 2;
e_lower = -5;
e_upper = 5;

N = 15;   /*We will take 15 particles*/
w = 0.5;
c1 = 0.7;
c2 = 0.8;

no_of_iter = 100;

P = e_lower + (e_upper - e_lower)*rand(N, dimensions);

V = zeros(N, dimensions);  // Velocity

obj = zeros(N, 1);
pb_val = zeros(N, 1);
pb_particle = zeros(N, 2);
gb_val_min = %inf;
// finding objective
for i = 1:N
    summ = 4*P(i, 1)^2 - 2.1*P(i, 1)^5 + (1/3)*P(i, 1)^6 + P(i, 1)*P(i, 2) -
4*P(i, 2)^2+ 4*P(i, 2)^4;

    obj(i, 1) = summ;
    pb_val(i, 1) = obj(i, 1);
    pb_particle(i, :) = P(i, :);

    if obj(i,1) < gb_val_min
        gb_val_min = obj(i, 1);
        gb_particle_min(1, :) = P(i, :)
    end


end

for k = 1:no_of_iter
    for i = 1:N
        for j = 1:dimensions
            V(i, j) = w*V(i, j) + c1*rand(1, 1)*(pb_particle(i, j) - P(i, j))+
c2*rand(1, 1)*(gb_particle_min(1, j) - P(i, j));
            P(i, j) = P(i, j) + V(i, j)

            if P(i,j) >5
                P(i,j) = 5
            end

            if P(i,j) < -5
                P(i,j) = -5
            end


        end
    end

    // Finding objective function to  know personal best and global best
```

```
    for i = 1:N
        summ = 4*P(i, 1)^2 - 2.1*P(i, 1)^5 + (1/3)*P(i, 1)^6 + P(i, 1)*P(i, 2)
- 4*P(i, 2)^2+ 4*P(i, 2)^4;
        obj(i, 1) = summ;

        // Finding the personal best and global best

        if obj(i,1) < pb_val(i, 1)
            pb_val(i, 1) = obj(i, 1);
            pb_particle(i, :) = P(i, :);
        end



        if obj(i,1) < gb_val_min
            gb_val_min = obj(i, 1);
            gb_particle_min(1, :) = P(i, :)
        end

    end

    gb_con(k, 1) = gb_val_min

end

plot(1:1:no_of_iter, gb_con)
```

RESULT:



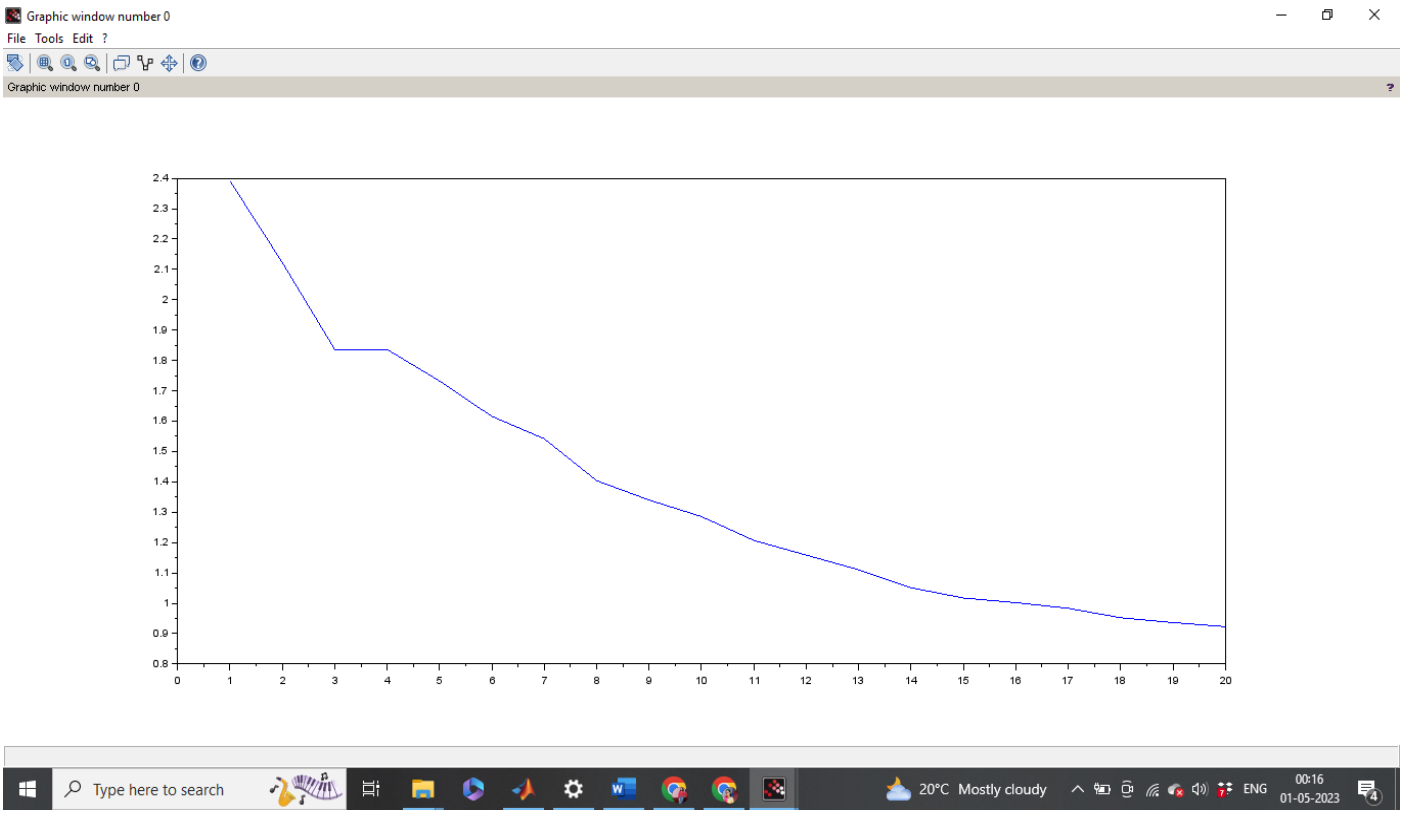
EXPERIMENT 4

CODE: (PSO state feedback controller)

```
clc
clear
A = [0 1 0; 0 0 1; 2 3 4];
B = [0 0 1]';
C = [1 0 0];
/* Objective function*/
function [ISE]=func(k1, k2, k3)
    ti=0;
    tf=5;
    h=0.001;
    U = 1;
    n=(tf-ti)/h;
    x1(1)=1;
    x2(1)=2;
    x3(1)=3;
    t(1)=0
  for i=1:n
        x1(i+1)=x1(i)+h*(x2(i));
        x2(i+1)=x2(i)+h*(x3(i));
        x3(i+1)=x3(i)+h*((2- k1)*x1(i) + (3- k2)*x2(i)+(4-k3)*x3(i));
        t(i+1)= t(i) + h
    end
    e = (1-x1)^2

    ISE = inttrap(t, e);
endfunction

dimensions = 3;

e_upper = 10;

N = 15;   /*We will take 15 particles*/
w = 0.5;
c1 = 0.7;
c2 = 0.8;

no_of_iter = 20;

k1_lower = 2.1;
k2_lower = 3.1;
k3_lower = 4.1;

k1 = k1_lower + (e_upper - k1_lower)*rand(N, 1);
k2 = k2_lower + (e_upper - k2_lower)*rand(N, 1);
k3 = k3_lower + (e_upper - k3_lower)*rand(N, 1);

P = cat(2, k1, k2, k3);
V = zeros(N, dimensions);  // Velocity

obj = zeros(N, 1);
pb_val = zeros(N, 1);
pb_particle = zeros(N, dimensions);
gb_val_min = %inf;
// finding objective
for i = 1:N
```

```
    summ = func(P(i, 1), P(i, 2), P(i, 3));

    obj(i, 1) = summ;
    pb_val(i, 1) = obj(i, 1);
    pb_particle(i, :) = P(i, :);

    if obj(i,1) < gb_val_min
        gb_val_min = obj(i, 1);
        gb_particle_min(1, :) = P(i, :)
    end

end

for k = 1:no_of_iter
    for i = 1:N
        for j = 1:dimensions
            V(i, j) = w*V(i, j) + c1*rand(1, 1)*(pb_particle(i, j) - P(i, j))+
c2*rand(1, 1)*(gb_particle_min(1, j) - P(i, j));
            updated_val = P(i, j) + V(i, j);

            if (j == 1) & (updated_val > 2)
                P(i,j) = updated_val
            end

            if (j == 2) & (updated_val > 3)
                P(i,j) = updated_val
            end

            if (j == 3) & (updated_val > 4)
                P(i,j) = updated_val
            end


        end
    end

    // Finding objective function to  know personal best and global best

    for i = 1:N
        summ = func(P(i, 1), P(i, 2), P(i, 3));
        obj(i, 1) = summ;

        // Finding the personal best and global best

        if obj(i,1) < pb_val(i, 1)
            pb_val(i, 1) = obj(i, 1);
            pb_particle(i, :) = P(i, :);
        end

        if obj(i,1) < gb_val_min
            gb_val_min = obj(i, 1);
            gb_particle_min(1, :) = P(i, :)
        end

    end

    gb_con(k, 1) = gb_val_min

end

plot(1:1:no_of_iter, gb_con)
```

RESULT:


Graphic window number 0
File Tools Edit ?
Graphic window number 0
2.4
2.3
2.2
2.1
2
1.9
1.8
1.7
1.6
1.5
1.4
1.3
1.2
1.1
1
0.9
0.8
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

Type here to search
20°C Mostly cloudy
ENG
00:16
01-05-2023