Assignment-5

1.

simulations for different values of damping constant and investigate its effect on the oscillations.

Code:

```
clear;close all;
global cnst;
global beta;
g=9.8;
l=1;
cnst=g/l;
mass=2;
b=sqrt(4*mass*mass*cnst);
beta=b/(2*mass);
beta=beta/5;
timescale=2*pi*sqrt(l/g);
dt=timescale/100;
tstart=0;
tfinal=10*timescale;
u0=zeros(2,1);
u0(1)=.5;
u0(2)=0;
[t,u]=ode45(@rhs34,[tstart:dt:tfinal],u0);
x1=57.5*u(:,1);
v1=u(:,2);



beta=b/(2*mass);
beta=beta;
timescale=2*pi*sqrt(l/g);
dt=timescale/100;
tstart=0;
tfinal=10*timescale;
u0=zeros(2,1);
u0(1)=.5;
```

```
u0(2)=0;
[t,u]=ode45(@rhs34,[tstart:dt:tfinal],u0);
x2=57.5*u(:,1);
v2=u(:,2);






beta=b/(2*mass);
beta=beta*10;
timescale=2*pi*sqrt(l/g);
dt=timescale/100;
tstart=0;
tfinal=10*timescale;
u0=zeros(2,1);
u0(1)=.5;
u0(2)=0;
[t,u]=ode45(@rhs34,[tstart:dt:tfinal],u0);
x3=57.5*u(:,1);
v3=u(:,2);



plot(t,x1,'r')
hold on
plot(t,x2,'b')
hold on
plot(t,x3,'g')

title('position vs time : red for underdamped, blue for critically damped,
green for overdamped')

figure
plot(x1,v1,'r')
hold on
plot(x2,v2,'b')
hold on
```

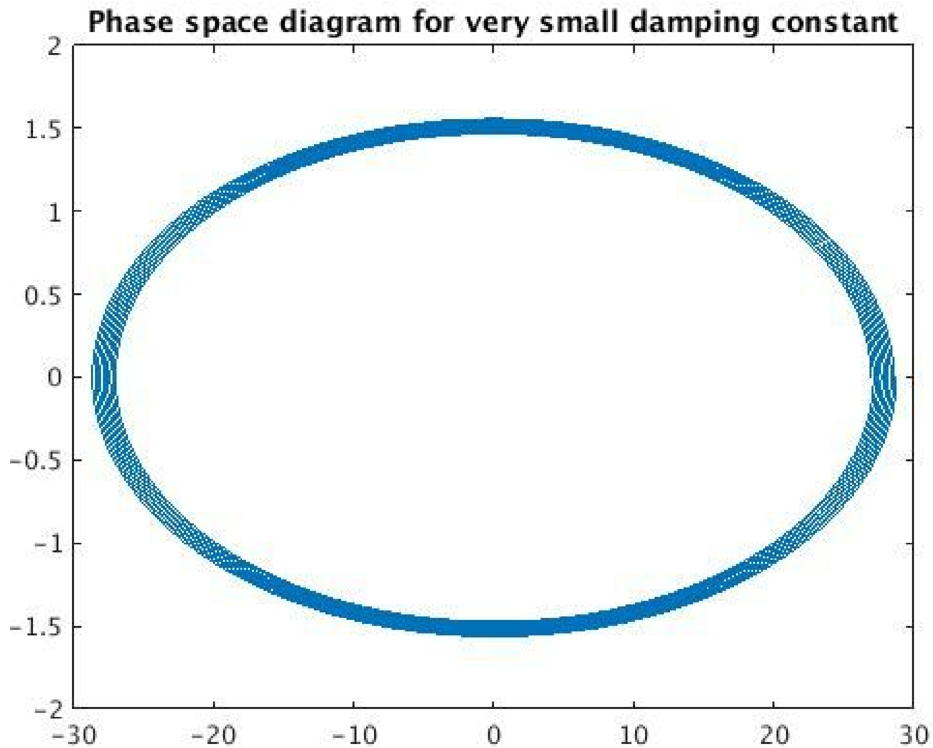
```
plot(x3,v3,'g')
title('Phase space diagram  : red for underdamped, blue for critically
damped, green for overdamped')
```

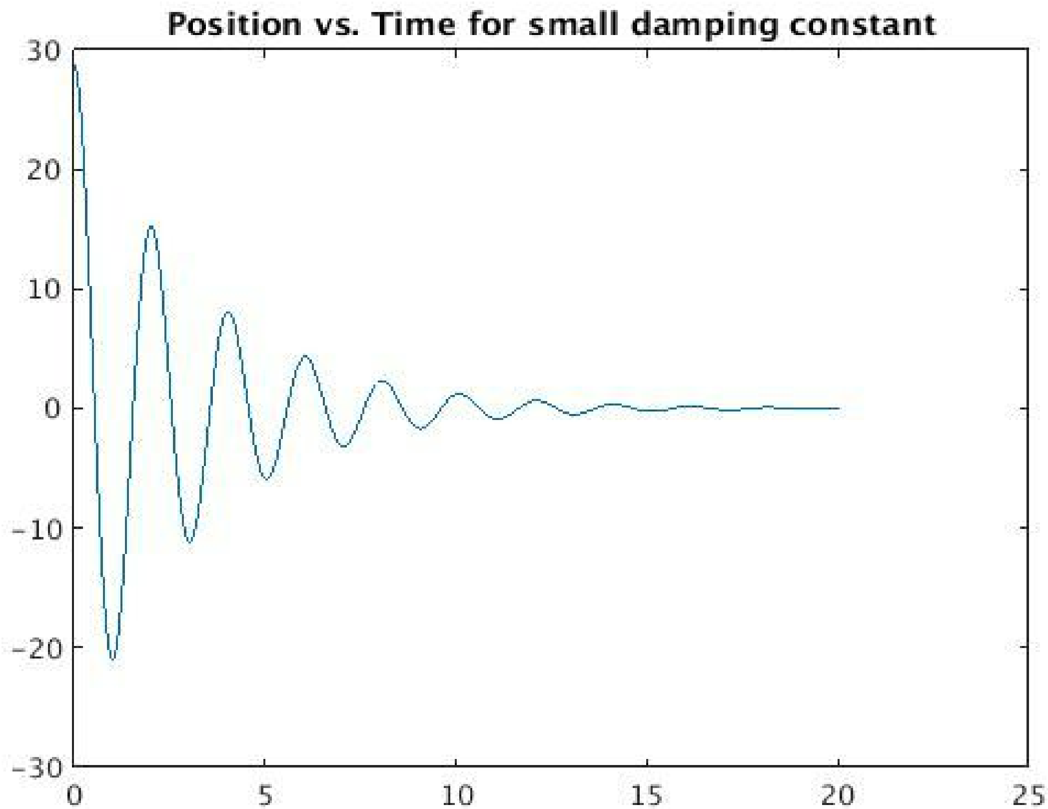
RHS:

```
function F=rhs34(t,u)

global cnst;

global beta;
F=zeros(length(u),1);
F(1)=u(2);
F(2)=-1*(2*beta*u(2)+cnst*u(1));
```

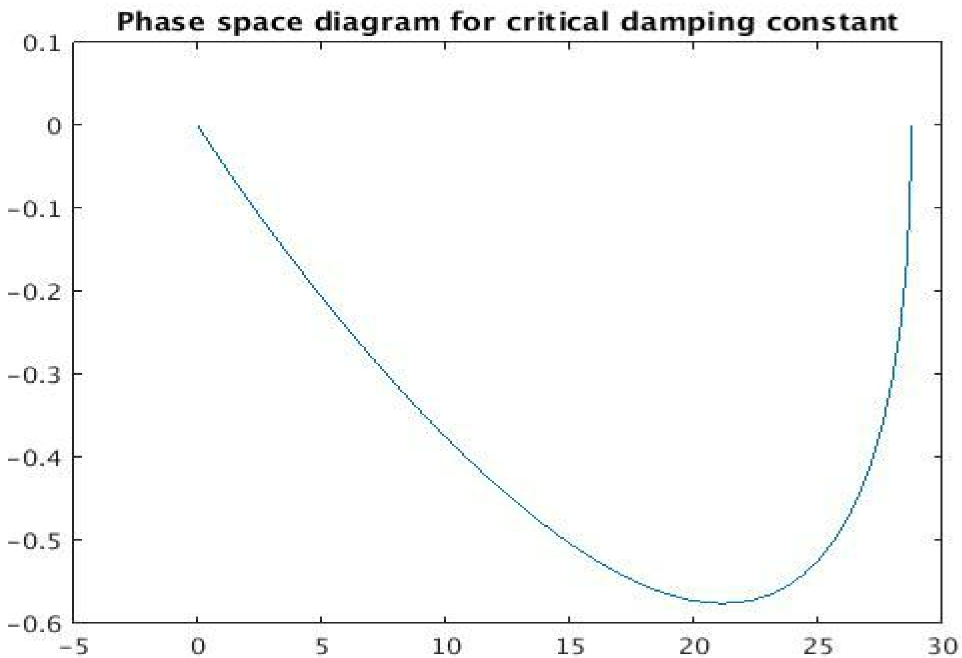
Diagram for small damping constant i.e. underdamped oscillations.

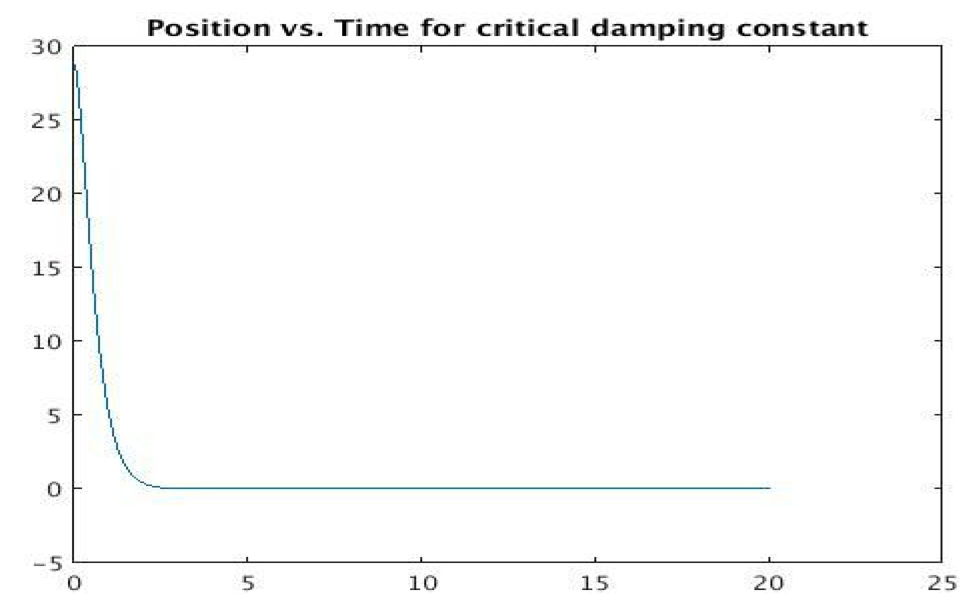
The phase plot clearly shows how the momentum and displacement diminish with time resulting in a dramatic spiral trajectory. Without the damping force, the trajectory would be a closed ovoid similar to that for the simple pendulum with small oscillations


Phase space diagram for very small damping constant
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-30
-20
-10
0
10
20
30



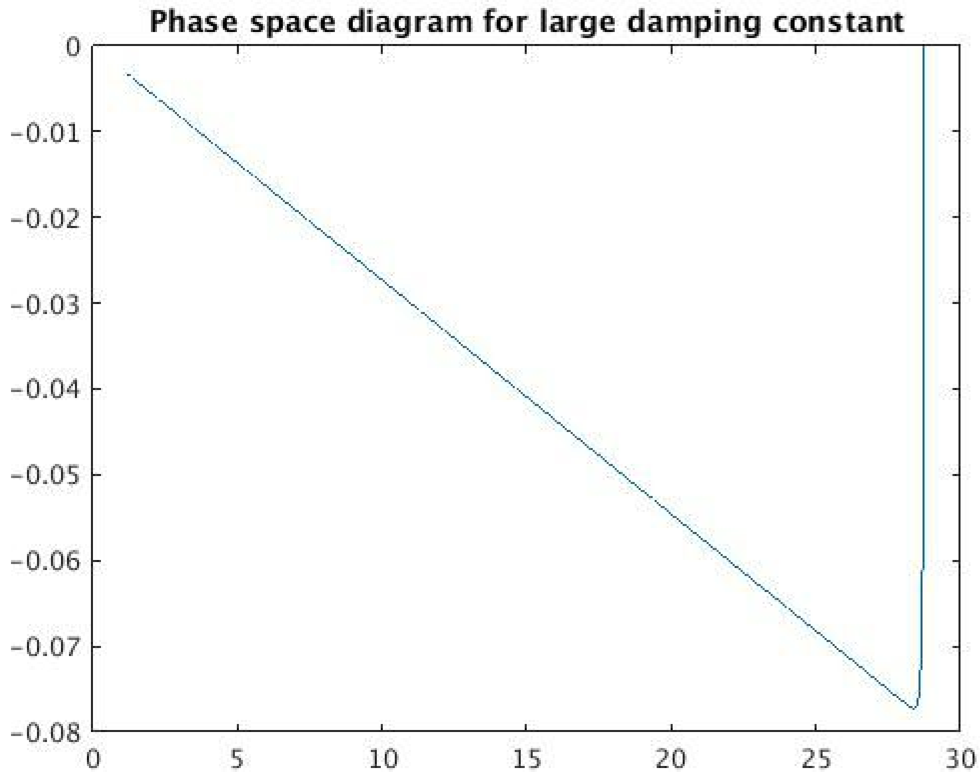
Below is the diagram for critical damping.
Here it is clearly seen that we get graph only for x<0 and in that case v>0. This is because once oscillator reaches equilibrium position(x=0), it stops oscillating and at that point v=x=0; i.e, there is no motion.

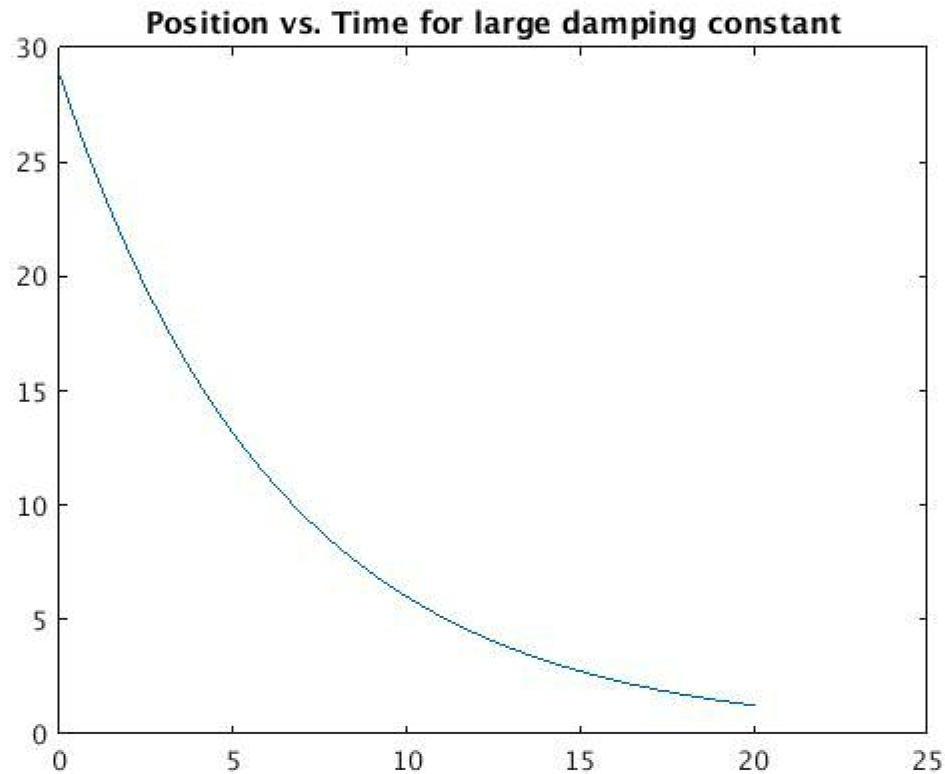




Below is diagram for over damping:

The phase space of over damped oscillations show that at longer times, it becomes asymptotic to curve v=(w2-b)x, where b=damping constant and w2=sqrt(b*b-w0*w0).



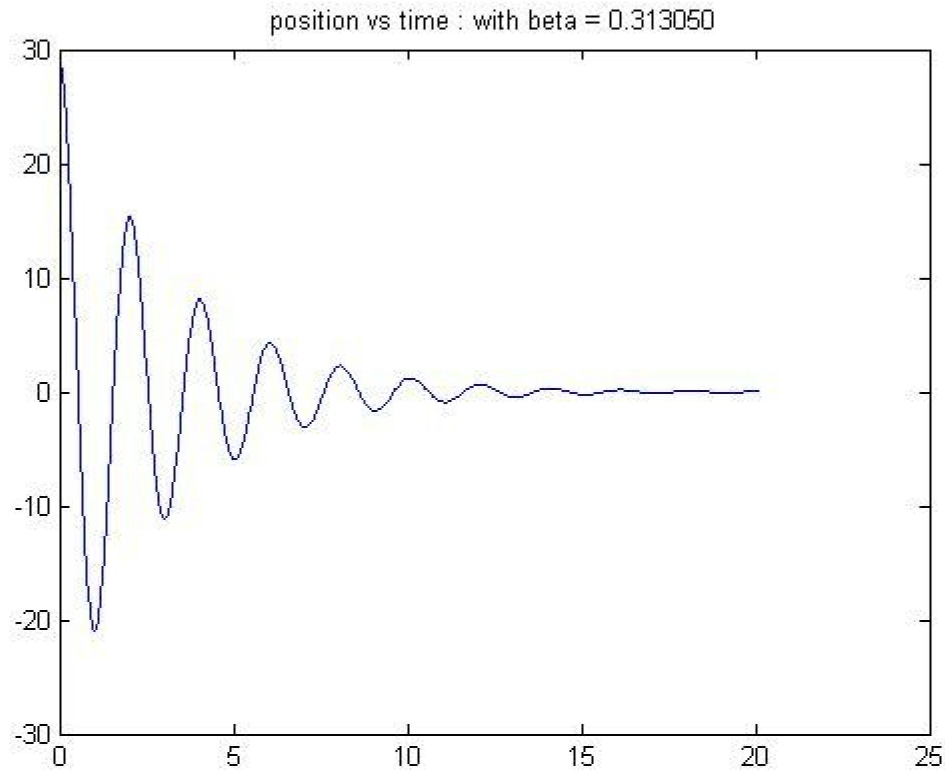


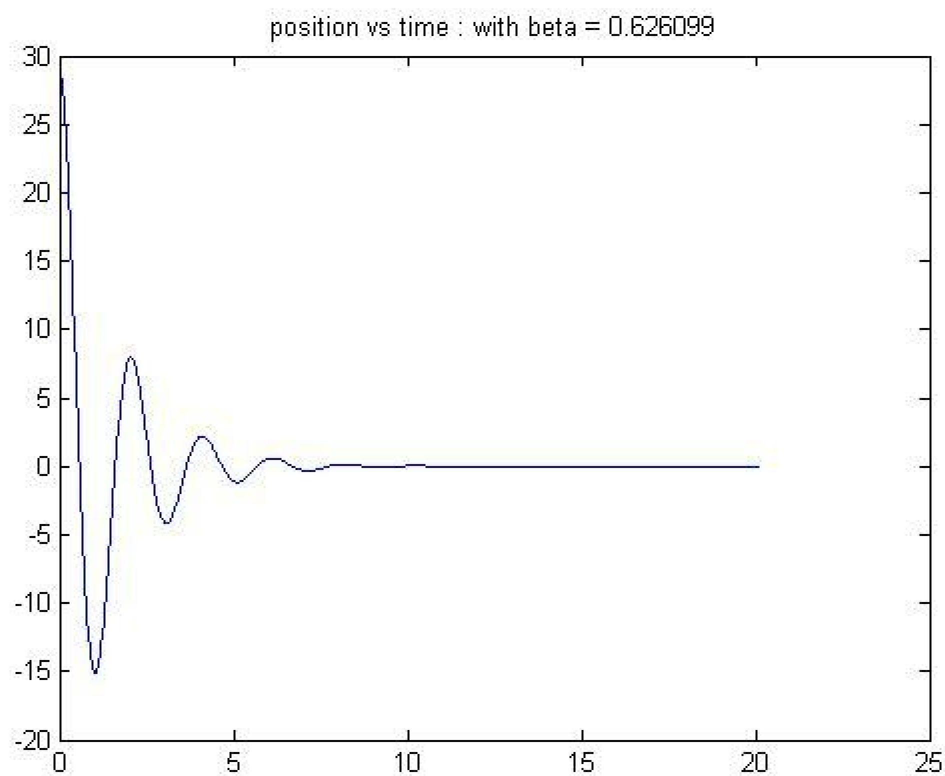
What is the effect of different initial condition?

There wont be any change if we change initial theta. Because the frequency and the damping does not depend on initial theta at all.

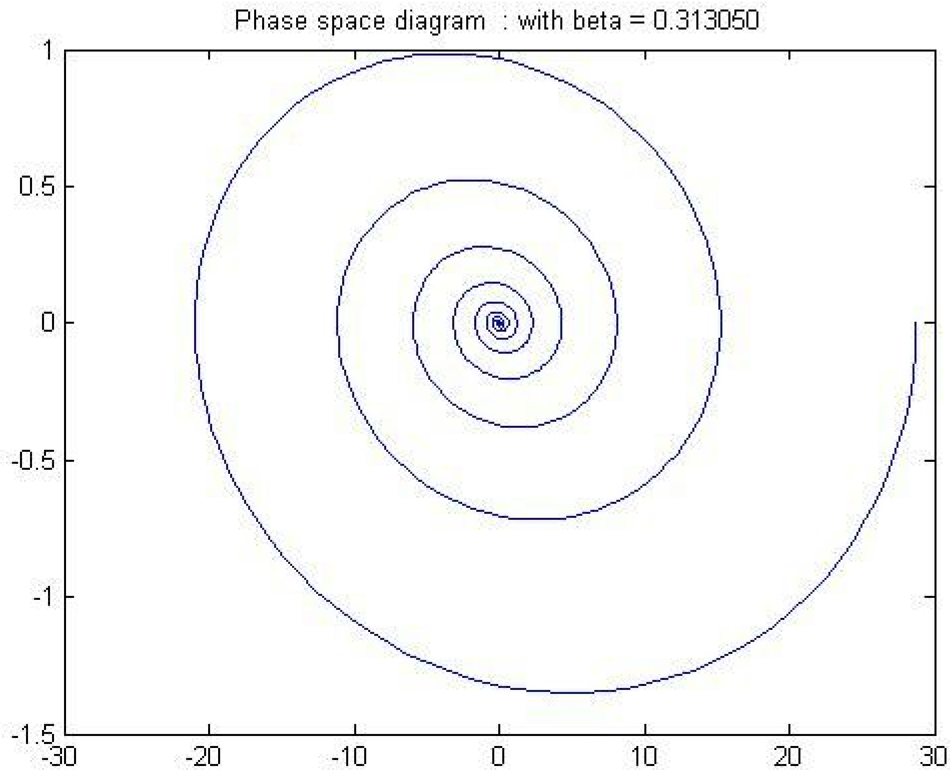
And if we change initial beta , yes there would be change in time for which the system attains the equilibrium position, but (in case of under damping) it wont change the damping frequency.

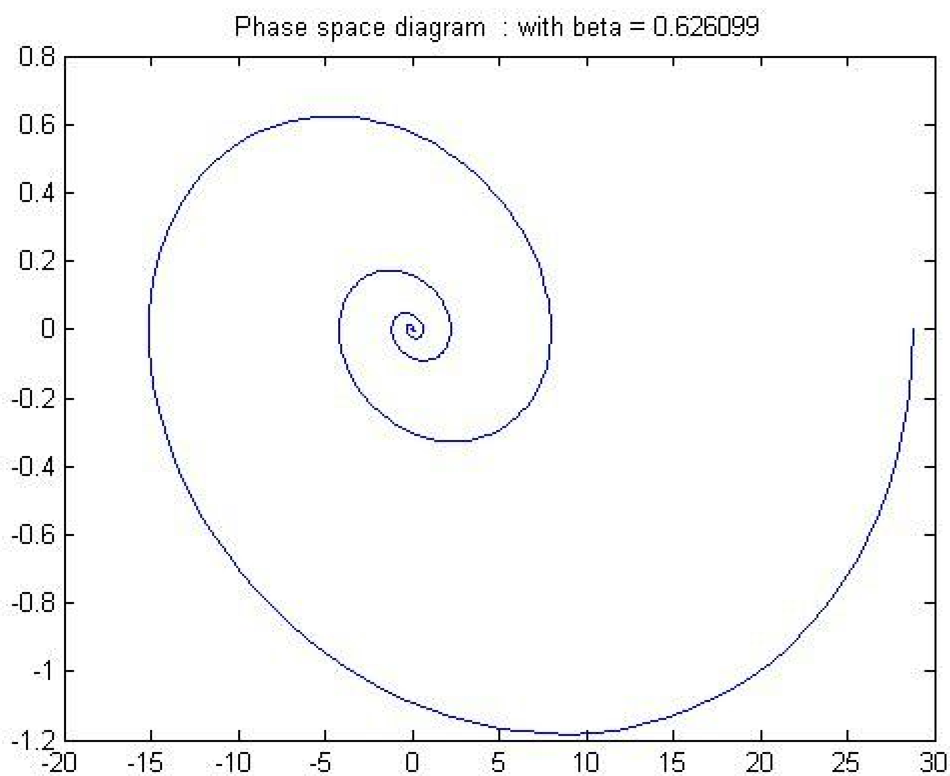
Hence, overall there won't be any huge effect by changing the initial conditions.
Graph to complement above explanation :


position vs time : with beta = 0.313050
30
20
10
0
-10
-20
-30
0
5
10
15
20
25

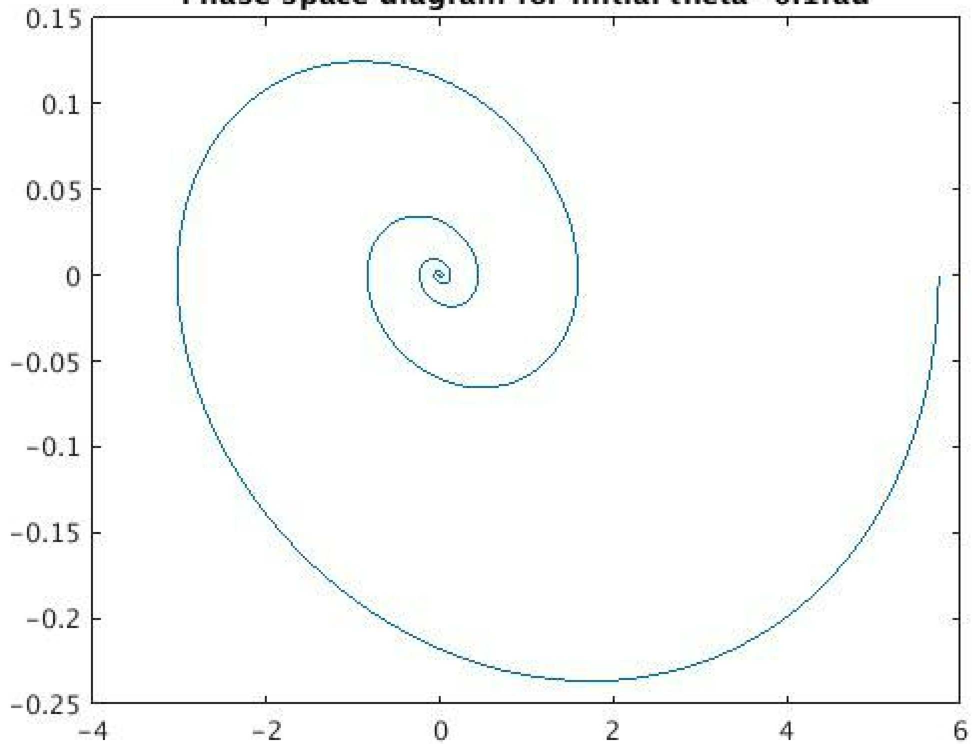

position vs time : with beta = 0.626099
30
25
20
15
10
5
0
-5
-10
-15
-20
0
5
10
15
20
25

Here we took different beta as our initial condition but we can see that it doesn't change with beta.

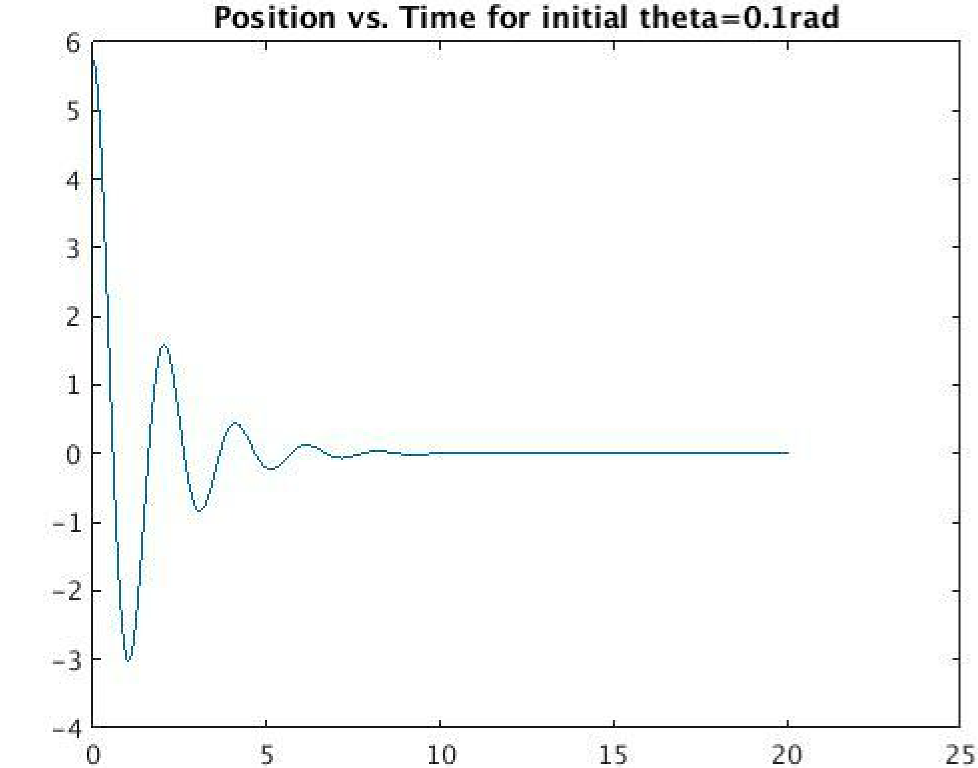

Phase space diagram : with beta = 0.313050
1
0.5
0
-0.5
-1
-1.5
-30
-20
-10
0
10
20
30

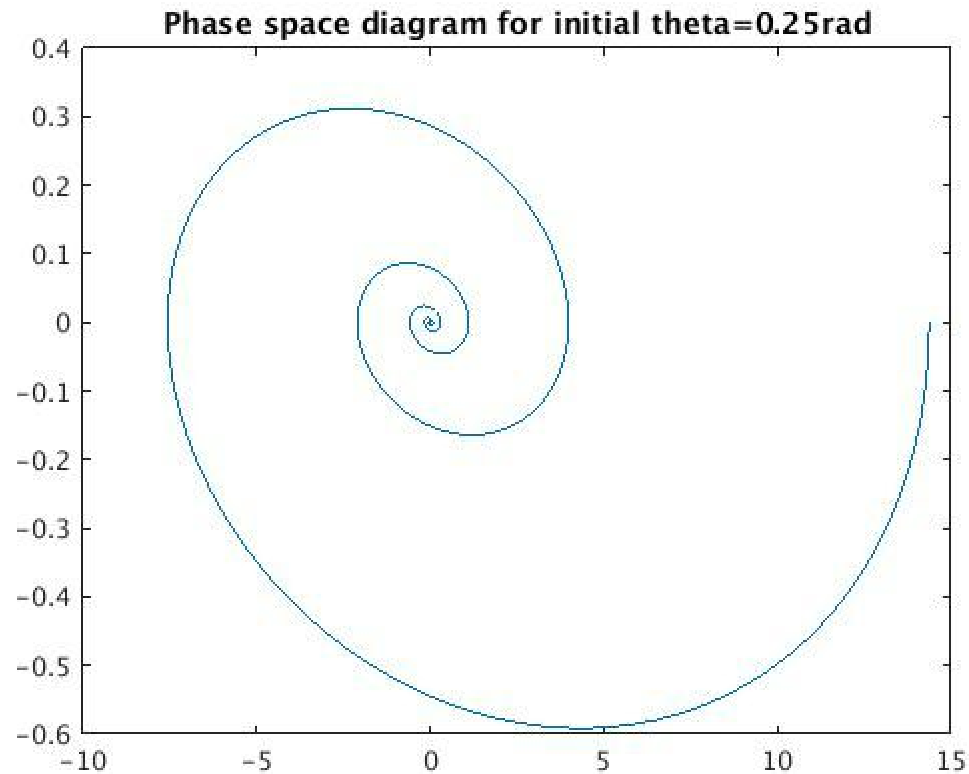

Phase space diagram : with beta = 0.626099
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1.2
-20
-15
-10
-5
0
5
10
15
20
25
30

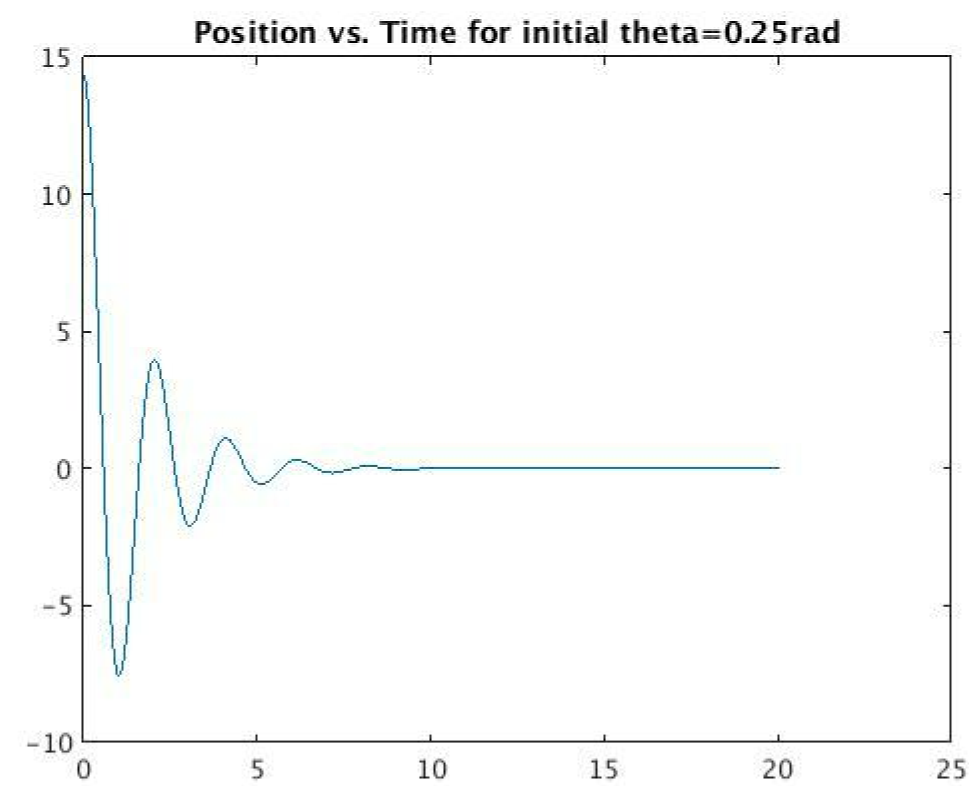
Phase space diagram for initial theta=0.1rad

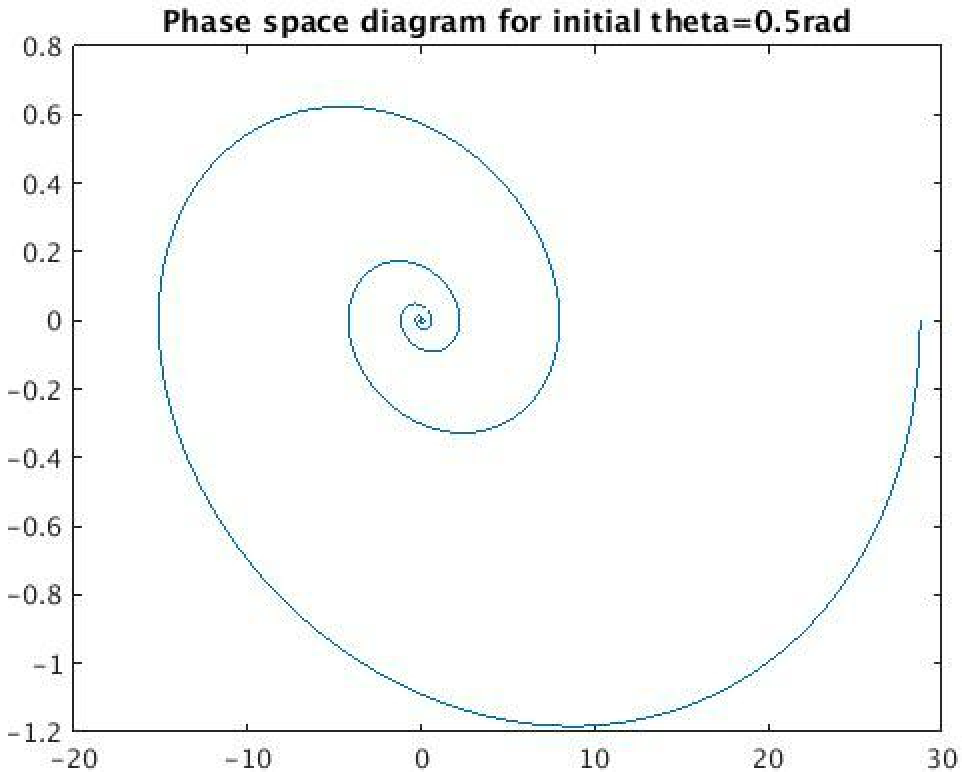


Position vs. Time for initial theta=0.1rad

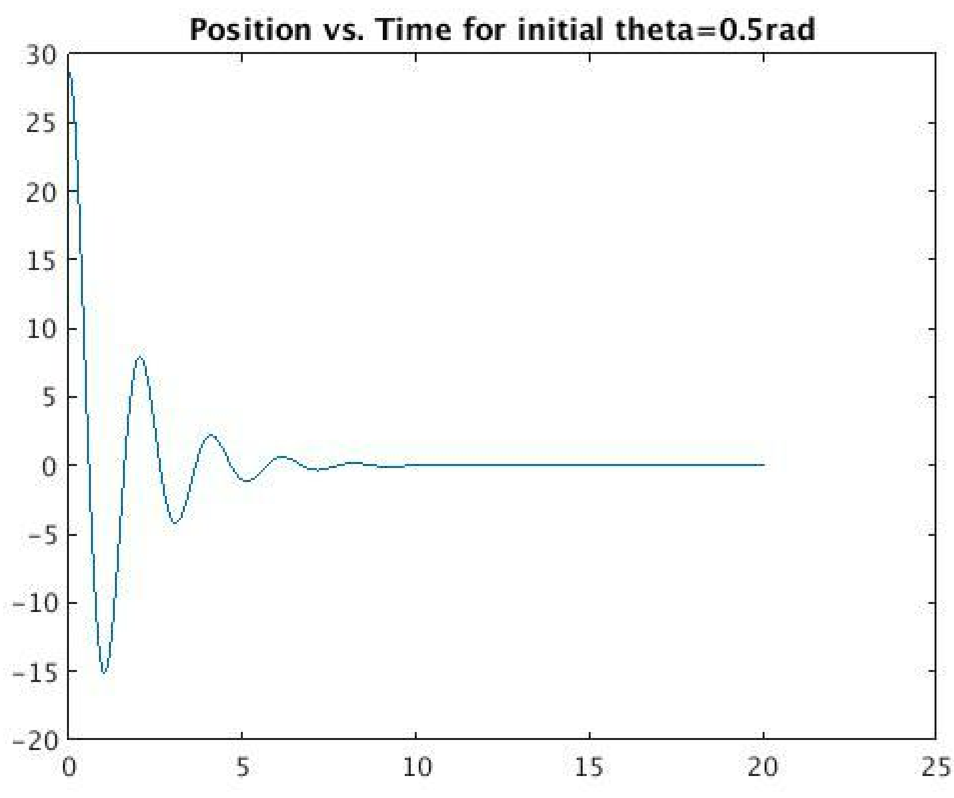



Phase space diagram for initial theta=0.25rad
0.4
0.3
0.2
0.1
0
−0.1
−0.2
−0.3
−0.4
−0.5
−0.6
−10
−5
0
5
10
15


Position vs. Time for initial theta=0.25rad
15
10
5
0
−5
−10
0
5
10
15
20
25

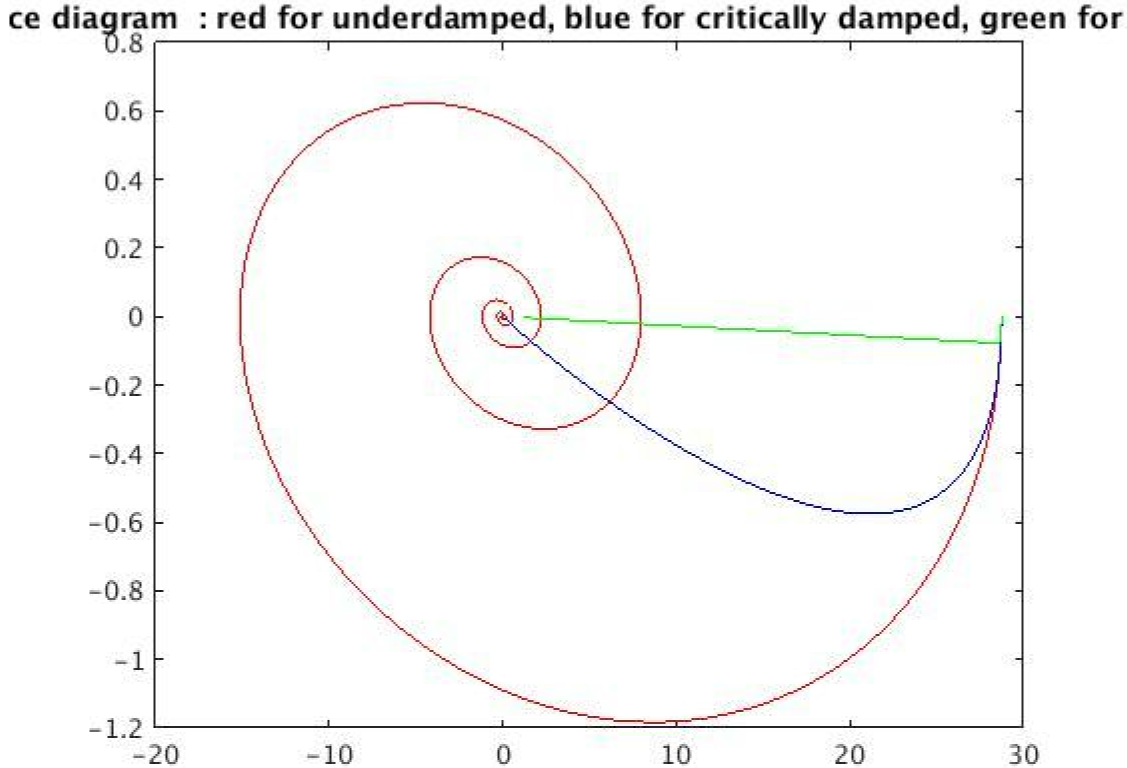


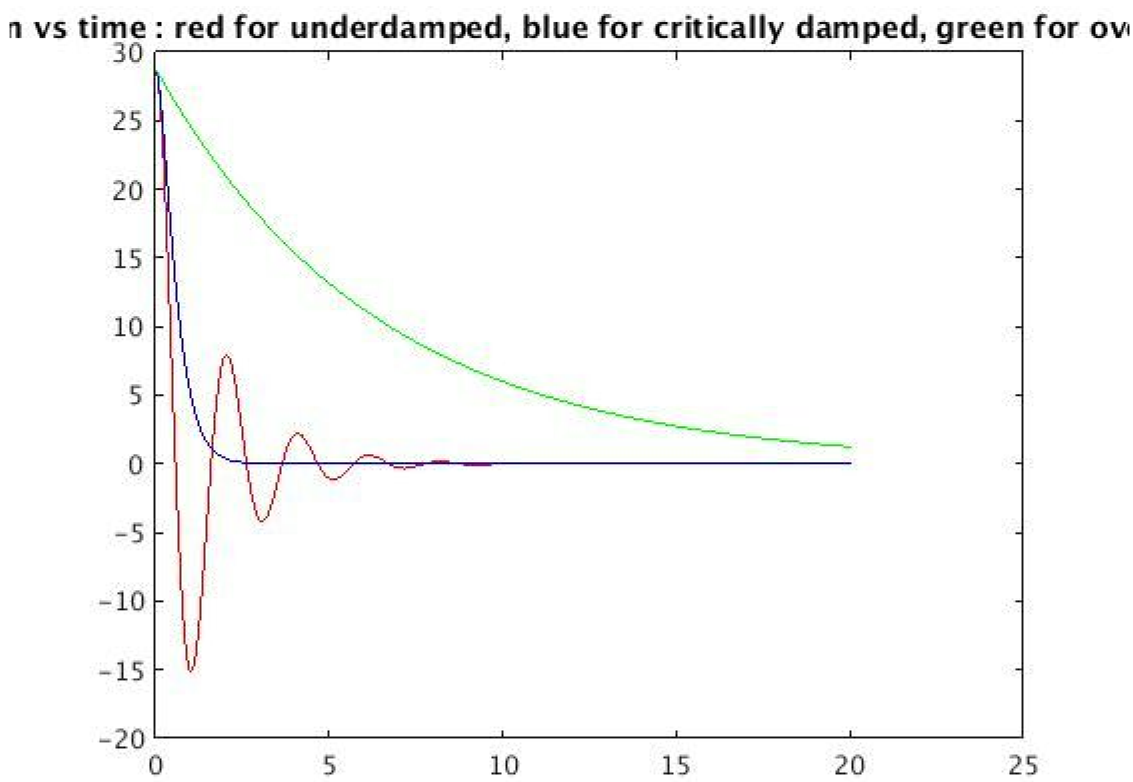
Phase space diagram for initial theta=0.5rad
0.8
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−0.8
−1
−1.2
−20
−10
0
10
20
30



Position vs. Time for initial theta=0.5rad
30
25
20
15
10
5
0
−5
−10
−15
−20
0
5
10
15
20
25

Phase space plot for all the three cases and explain the nature of the curve:


ce diagram : red for underdamped, blue for critically damped, green for
0.8
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−0.8
−1
−1.2
−20
−10
0
10
20
30


n vs time : red for underdamped, blue for critically damped, green for ov
30
25
20
15
10
5
0
−5
−10
−15
−20
0
5
10
15
20
25

In the above graph, red line represents underdamped oscillations, blue represents critically damped oscillations and green represents over damped oscillations. It is clear from the graph that critically damped oscillations takes minimum time to reach equilibrium position and maintains its equilibrium over time. Here from figure we can even say that underdamped oscillation reaches the equilibrium position very quickly, but it does not remain steady over there, it oscillates around the equilibrium point and damps out over time according to the value of damping constant.

2:

In the case of driven damped oscillation motion the required factor to stop the motion from termination is external force. In this example we applied a force F=f0*cos(wd*t). Instead continuously oscillate with smaller and smaller amplitude and getting stopped the spring mass system continues to oscillate. We can say that now the only frequency that is responsible for motion is the frequency of the driven force, so now it will oscillate with frequency of driven force.

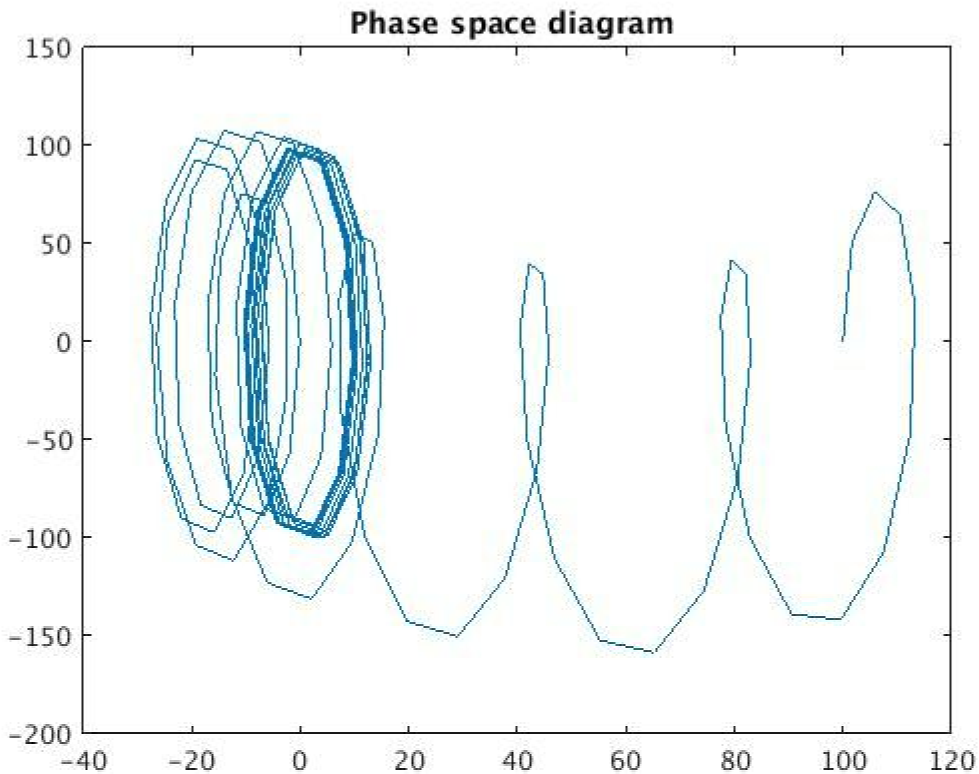
Code:

```
clear;close all;
global cnst;
global beta;
global mass;
mass=500;
k=500;
cnst=mass/k;
global time;
beta=.5;
global tstart;
global dt;
timescale=2*pi*sqrt(mass/k);
dt=timescale/100;
```

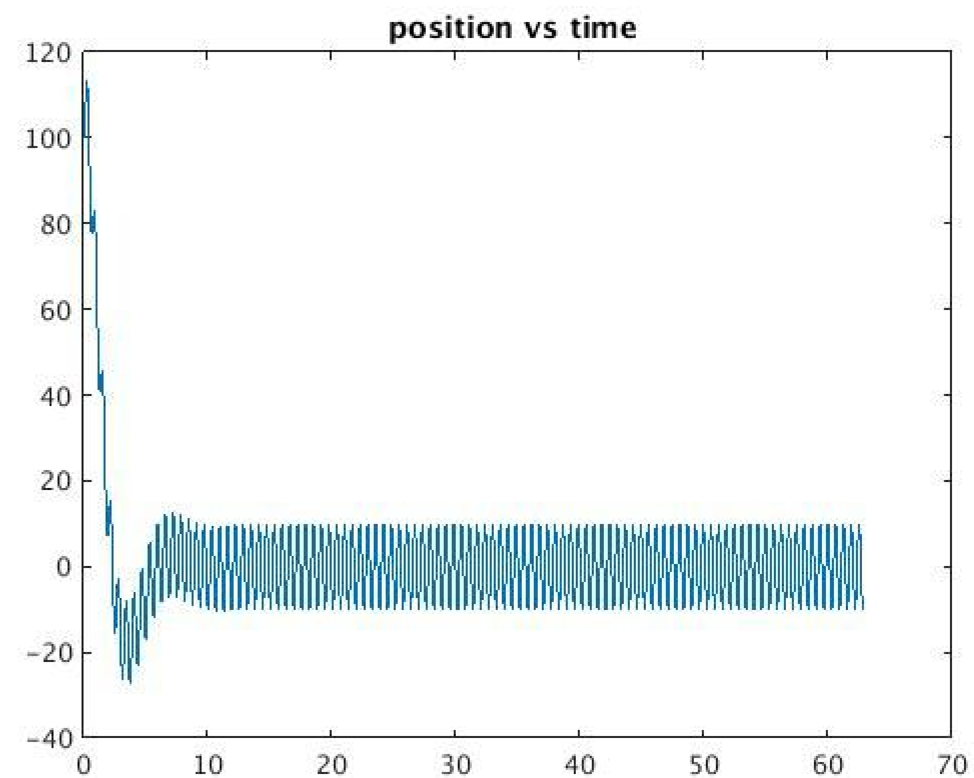
```
tstart=0;
tfinal=10*timescale;
u0=zeros(2,1);
u0(1)=100;
u0(2)=0;


[t,u]=ode45(@rhs52,[tstart:dt:tfinal],u0);
x1=u(:,1);
v1=u(:,2);


plot(t,x1)
title('position vs time ')
figure
plot(x1,v1)

title('Phase space diagram ')
```

RHS:

```
function F=rhs52(t,u)

global cnst;
global mass;
global beta;

F=zeros(length(u),1);
wd=10;
exF=1000*cos(wd*t);
F(1)=u(2);
F(2)=-1*(2*beta*u(2)+cnst*u(1))+exF;
```



Phase space diagram
150
100
50
0
−50
−100
−150
−200
−40
−20
0
20
40
60
80
100
120



position vs time
120
100
80
60
40
20
0
−20
−40
0
10
20
30
40
50
60
70

Here, initially, in transient state, we see some unusual activity. The amplitude in steady state remains constant, whereas in transient state it varies largely. Initial transient behavior results from different combination of the natural frequency of the system and the driving frequency and minimal damping can cause the initial transient behavior to last for a long time before steady state is attained. The apparent struggle between the driving frequency and the natural frequency of the system can produce some bizarre phase plot trajectories that can resemble chaotic motion. But in steady state frequency remains constant, it would be equal to the frequency of driving force.

2.b:

Code:

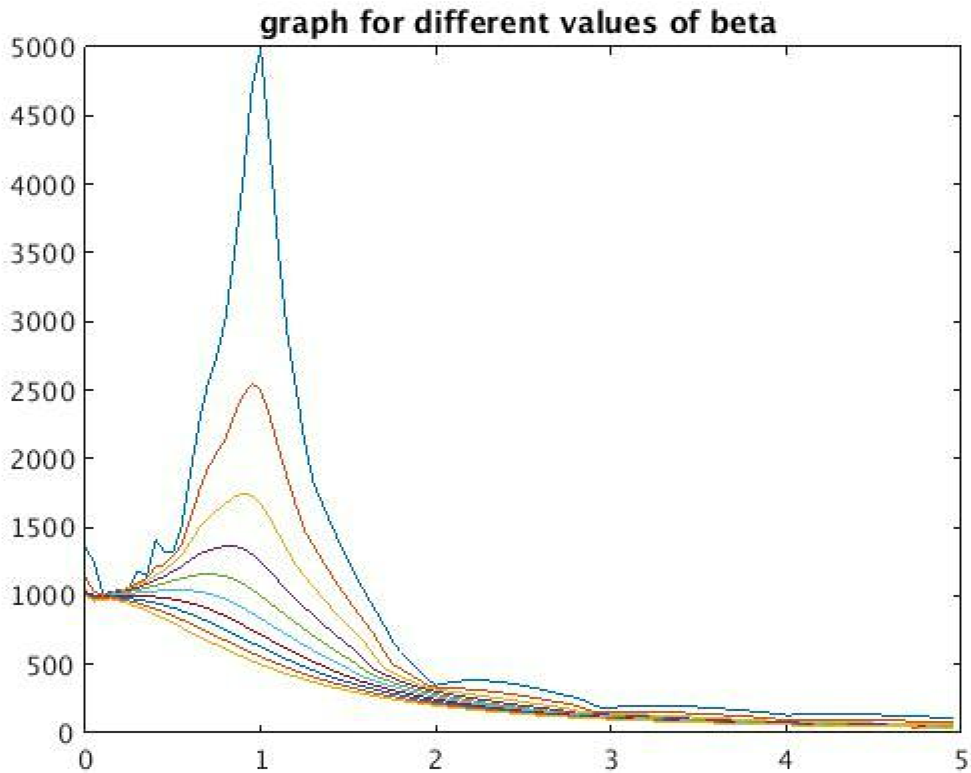
```
clear;close all;
global cnst;
global beta;
global mass;
mass=500;
k=500;
cnst=mass/k;
global time;
global tstart;
global dt;
timescale=2*pi*sqrt(mass/k);
dt=timescale/1000;
tstart=0;
tfinal=10*timescale;
u0=zeros(2,1);
u0(1)=100;
u0(2)=0;
global wd;
ni=100;
w=zeros(ni,1);
amp=zeros(ni,1);
ws=0;
wfinal=5*sqrt(k/mass);
dw=(wfinal-ws)/ni;
wd=ws;
```

```
n=10;
beta = 0;
for i = 1:n
        beta = beta + 0.1;wd=ws;
        for step = 1: ni
        w(step)=wd;
        [t,u]=ode45(@rhs53,[tstart:dt:tfinal],u0);
        x1=u(:,1);
        v1=u(:,2);
        maxa=-1;

        for st1= 900:tfinal/dt
        if x1(st1)>maxa
                maxa=x1(st1);
        end
        end
        amp(step)=maxa;
        wd=wd+dw;
        end
   plot(w,amp)
        %str = sprintf('Plot with beta = %f',beta);
        title('graph for different values of beta')
        hold on
        %title('graph for beta = ' beta)
        w=zeros(ni,1);
        amp=zeros(ni,1);
end
```

RHS:

```
function F=rhs53(t,u)

global cnst;
global mass;
global beta;
global wd;
F=zeros(length(u),1);

exF=1000*cos(wd*t);
```

```
F(1)=u(2);
F(2)=-1*(2*beta*u(2)+cnst*u(1))+exF;
```



Here is a plot for different values of beta. Here we vary beta from 0.1 to 1. We get maximum amplitude for beta=0.1 and amplitude decreases as beta increases. Here in each curve we get maximum amplitude of that particular curve at w=1, as at that point it would resonate.

Phase difference:

Code:

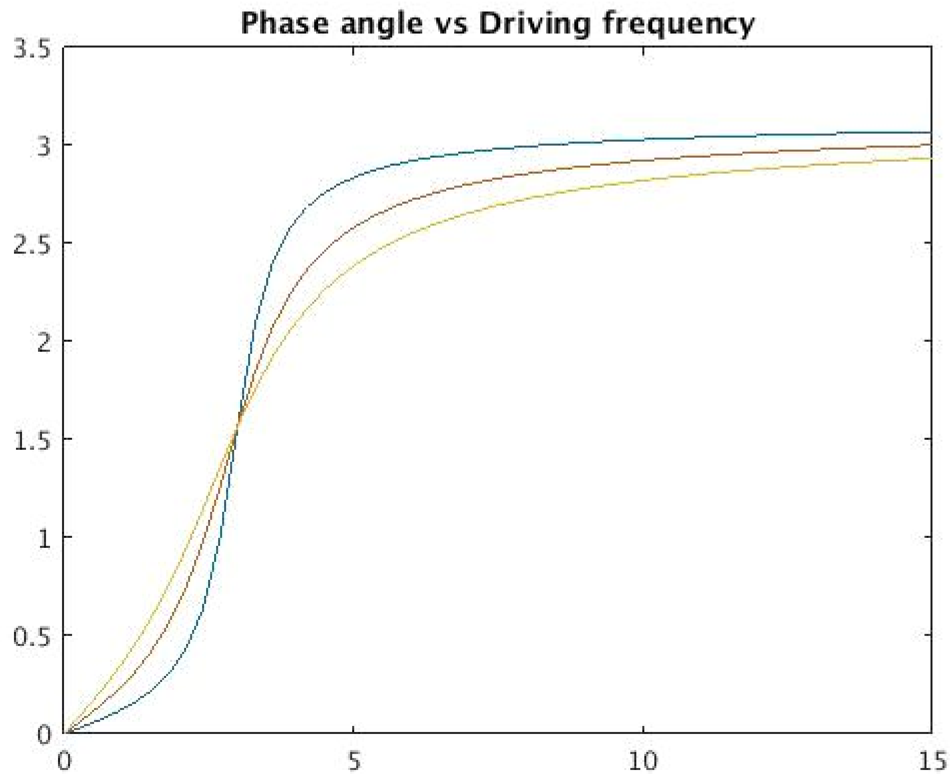
```
global b wd;
omega=zeros(51,1);
 w0 =1;
 dt=1/100;
```

```
 b=0.1
 tstart=0;
 tfinal=500;
 u0=zeros(2,1);
u0(1)=0.5;
 u0(2)=0;
 delta = zeros(51,1);
 i=1;
for step=0:.1:5*w0
        wd=step;
        omega(i)=wd;
        [t,u]=ode45(@rhs3,[tstart:dt:tfinal],u0);
        x1=57.5*u(:,1);
        % radian-->degree   v1=u(:,2);
        delta(i)=atan((wd*2*b)/((w0*w0)-(wd*wd)));
        if(delta(i)<0)
                delta(i)=delta(i) + pi;
        end
        i=i+1;
end
 plot(omega,delta);
title('omega vs delta');
```

rhs3.m:

```
function F=rhs3(t,u);
 F=zeros(length(u),1);
global w0;
global b;
global wd;
 F(1)=u(2);
 F(2)=-2*b*u(2)-w0*w0*u(1)+1000*cos(wd*t);
```

graphs: frequency vs phase angle for different forces:


Phase angle vs Driving frequency
3.5
3
2.5
2
1.5
1
0.5
0
0
5
10
15