

Mécanique Numérique DM4

Etude d'un oscillateur linéaire amorti à un degré de liberté

EX1.1 Résolution avec un schéma d'Euler explicite

Programmation en Matlab est comme ca:

```
clc
clear all
close all

%% Condition initiale
w0=2*pi;
x0=0.01;
Dx0=0;
T0=1;
eps=0.02;

%% Euler explicite
Uexp(1,1)=x0;
Uexp(2,1)=Dx0;
% changer le pas de temps
% dt=0.01;
% a) dt > 2*eps/w0 (0.0064) x diverge
% dt=2*eps/w0; % b) x oscille
% dt=0.8*2*eps/w0;
% c) x converge
for dt=[0.01 2*eps/w0 0.8*2*eps/w0 0.2*2*eps/w0 0.05*2*eps/w0]
n=floor(10*T0/dt);
A=[1 dt;-w0^2*dt 1-2*dt*eps*w0];
% oscillateur linéaire amorti à 1ddl en Euler explicite
for i=2:n+1
    Uexp(:,i)=A*Uexp(:,i-1);
end
syms t
x=dsolve('D2x+2*0.02*2*pi*Dx+(2*pi)^2*x=0','x(0)=0.01','Dx(0)=0'); t=0:dt:dt*n;
x=subs(x,t);
```

```
%% Figure de x-exp et x exacte
```

```
figure  
plot(t,Uexp(1,:))  
hold on  
plot(t,x)  
grid on  
legend('x-exp','x-exacte');  
title('x-exp et x-exacte');  
xlabel('t');  
end
```

Conclusion:

1.1.a) Quand $dt > 2 \cdot \text{eps}/w_0$ (0.0064), on choisit $dt=0.01$, x diverge.

1.1.b) Quand $dt=2 \cdot \text{eps}/w_0$, x oscille.

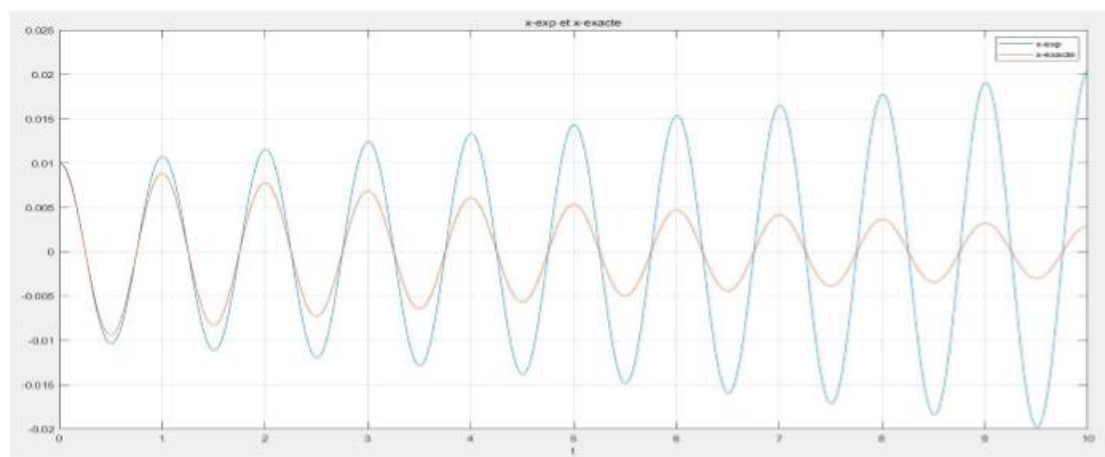
1.1.c) Quand $dt=0.8 \cdot 2 \cdot \text{eps}/w_0$, x converge.

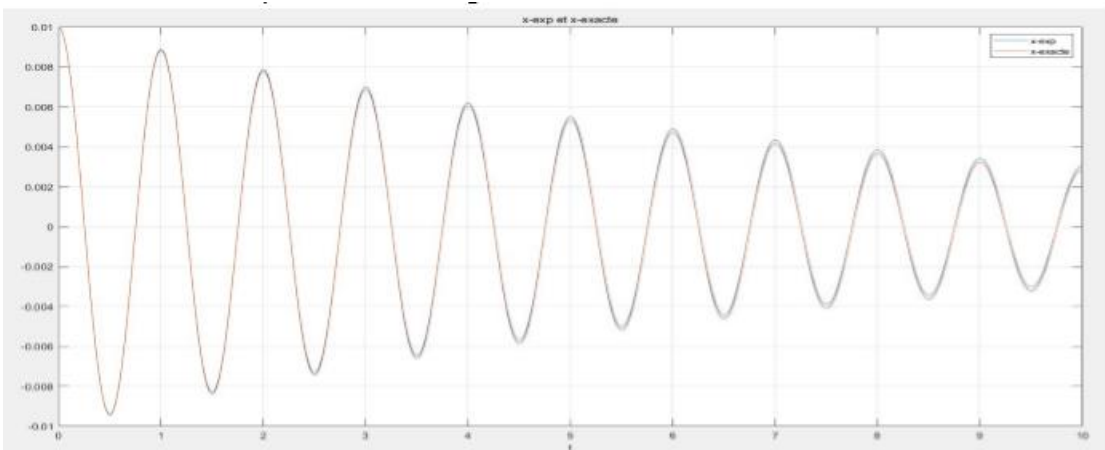
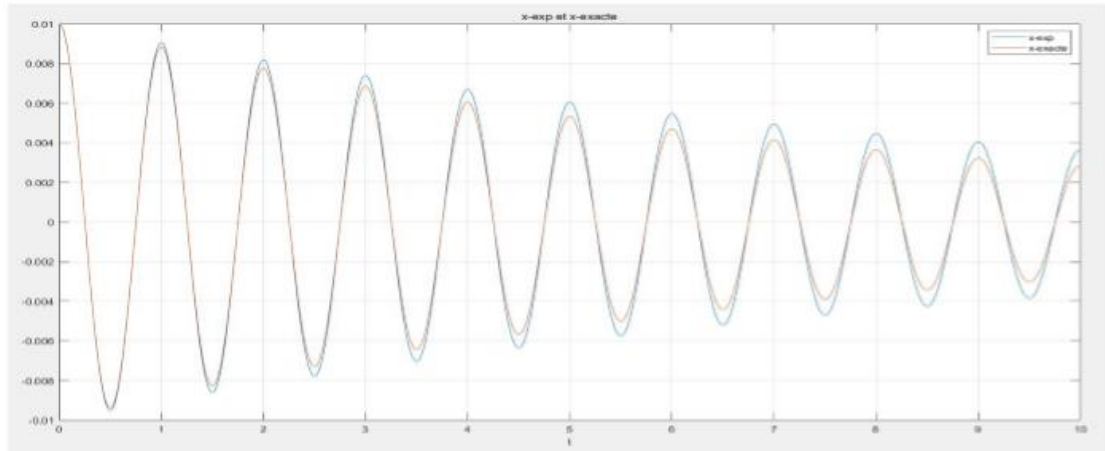
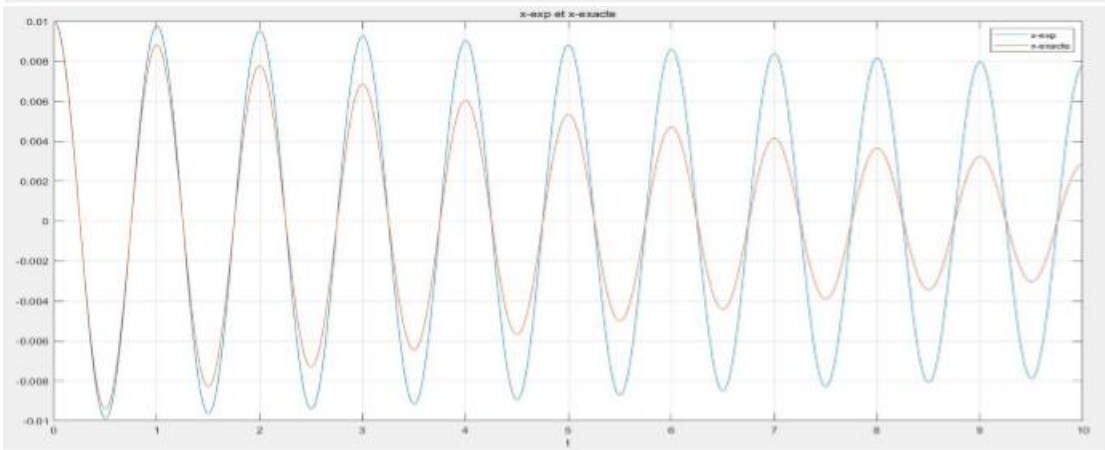
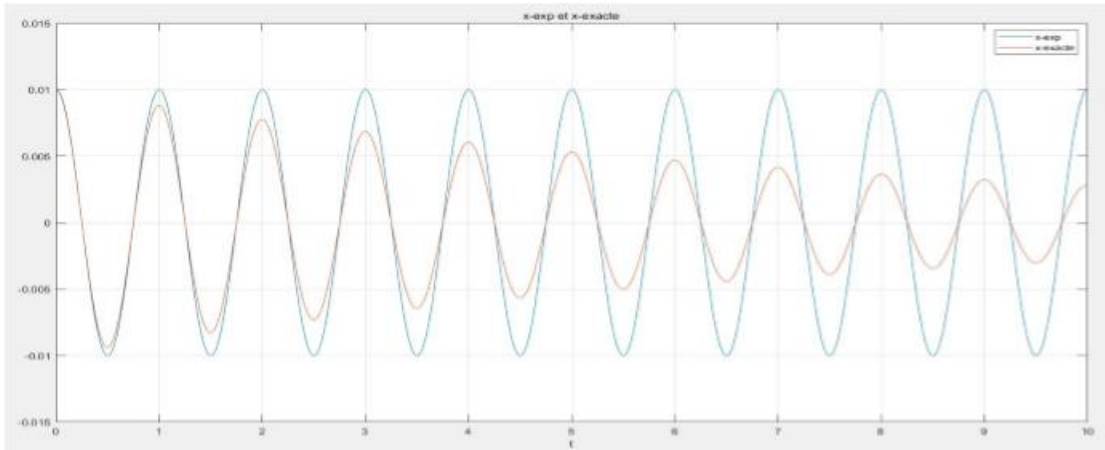
1.1.d) Etude de la précision Quand $dt=0.2 \cdot 2 \cdot \text{eps}/w_0$, x converge.

Quand $dt=0.05 \cdot 2 \cdot \text{eps}/w_0$, x converge.

Critère de la précision : x doit converger comme x -exacte, donc $dt/(2 \cdot \text{eps}/w_0) \leq 0.05$, mais le schéma Euler explicite diverge, ainsi la différence entre x -exp et x -exacte devient de plus en plus grande.

On obtient les dessins dessous:





EX1.2 Résolution avec un schéma d'EULER implicite

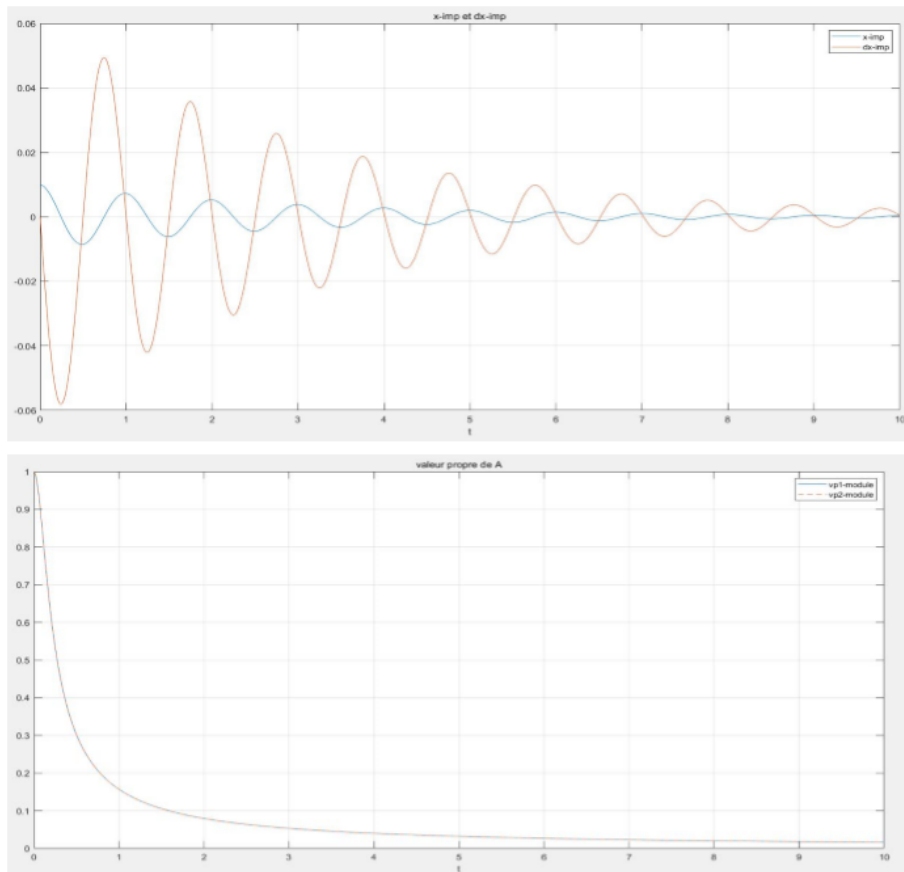
Programme en matlab est comme ca:

```
clc
clear all
close all
%% Condition initiale
w0=2*pi;
x0=0.01;
Dx0=0;
T0=1;
eps=0.02;
%% Euler implicite
Uimp(1,1)=x0;
Uimp(2,1)=Dx0;
dt=0.01;
% calculer le pas de temps critique
n=floor(10*T0/dt);
B=[1+(w0*dt)^2 , 2*eps*w0*dt^2 ; w0^2*dt , 1+2*eps*w0*dt];
C=[1 , dt ; 0 , 1];
A=C/B;
% oscillateur linéaire amorti à 1ddl en Euler implicite
for i=2:n+1
    Uimp(:,i)=A*Uimp(:,i-1);
end
%% Figure imp
figure
t=0:dt:dt*n;
plot(t,Uimp(1,:))
hold on
plot(t,Uimp(2,:))
grid on
legend('x-imp','dx-imp');
title('x-imp et dx-imp');
xlabel('t');
```

```

%% Matrice A
dt=0:0.01:10;
for i=1:length(dt)
Bnew{1,i}=[1+(w0*dt(i))^2 , 2*eps*w0*dt(i)^2 ;
w0^2*dt(i) , 1+2*eps*w0*dt(i)];
Cnew{1,i}=[1 , dt(i) ; 0 , 1];
Anew{1,i}=Cnew{1,i}/Bnew{1,i};
E{1,i}=eig(Anew{1,i});
vp1_module(1,i)=norm(E{1,i}{1,1});
vp2_module(1,i)=norm(E{1,i}{2,1});
end
figure
plot(dt,vp1_module)
hold on
plot(dt,vp2_module,'--')
grid on
legend('vp1-module','vp2-module');
title('valeur propre de A');
xlabel('t');

```



Conclusion:

x et dx convergent, donc le schéma d'EULER implicite est précis.

Les modules des valeurs propres de A sont inférieur à 1 et ils convergent, donc il n'y a pas le pas de temps critique, car les solutions convergent toujours.

EX1.3 Résolution avec un schéma de RUNGEKUTTA

Programmation en MATLAB est comme ca:

main.m

clc

clear all

close all

%% Condition initiale

w0=2*pi;

x0=0.01;

Dx0=0;

T0=1;

eps=0.02;

a=0;

```

b=100*T0;
% Changer le coefficient h
for h=[0.04 0.96 1.04]
dt=h*2*sqrt(2)/w0;
%% Figure de RK_45
figure
[t,x]=RK_45(@fun_OLA_1ddl,dt,[0.01 0],a,b);
plot(t,x(1,:),t,x(2,:));
grid on;
legend('x-RK45','dx-RK45');
title('x-RK45 et dx-RK45');
xlabel('t');
end

```

fun_OLA_1ddl.m

```

function dy=fun_OLA_1ddl(y,t)
w0=2*pi;
eps=0.02;
dy=zeros(2,1);
dy(1)=y(2);
dy(2)=-2*eps*w0*y(2)-w0^2*y(1);
end

```

RK_45.m

```

function [t,y]=RK_45(fun,dt,y0,a,b)
%(f,dt,C1,t0,tf)
y(:,1)=y0;
t(1,1)=0;
n=(b-a)/dt;
for i=1:n
    t(1,i+1)=t(1,i)+dt;
    k1=fun(y(:,i),t(1,i));
    k2=fun(y(:,i)+k1*dt/2,t(1,i)+dt/2);
    k3=fun(y(:,i)+k2*dt/2,t(1,i)+dt/2);
    k4=fun(y(:,i)+k3*dt,t(1,i)+dt);

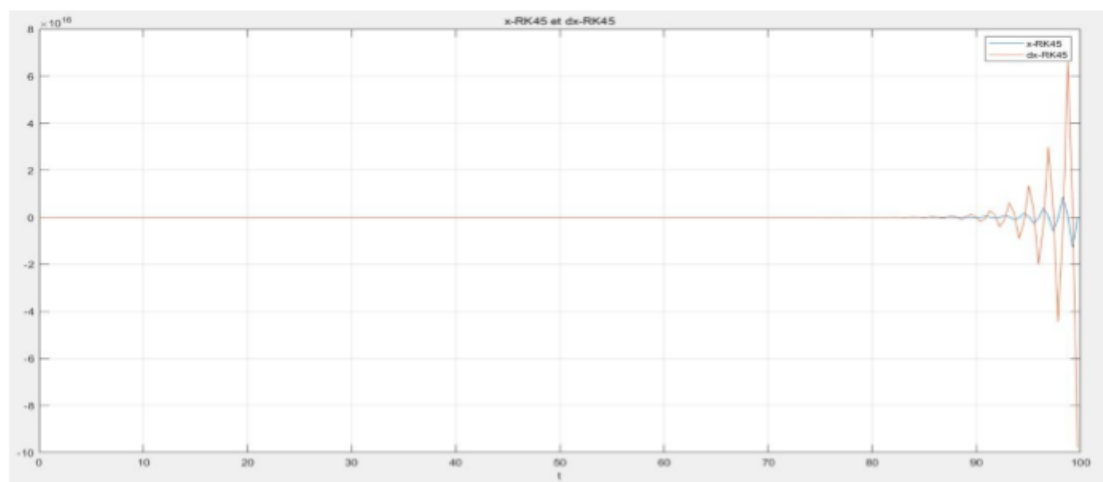
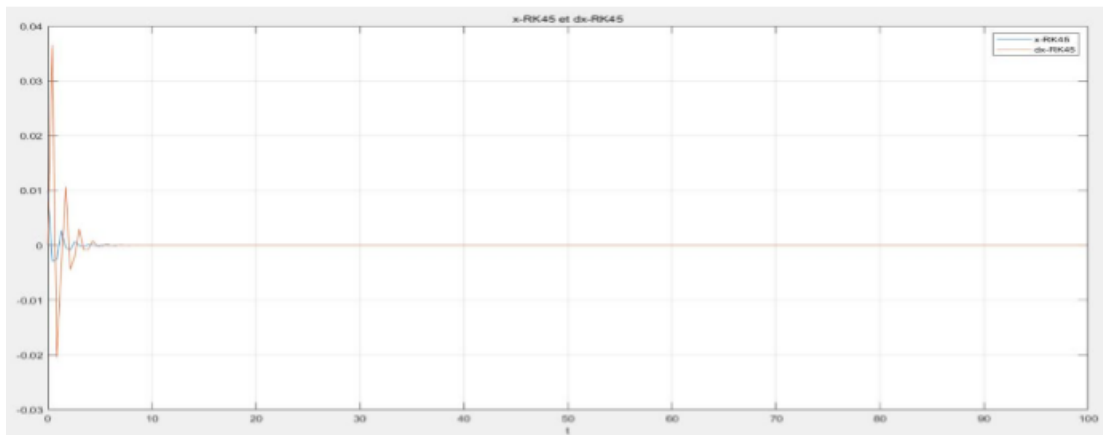
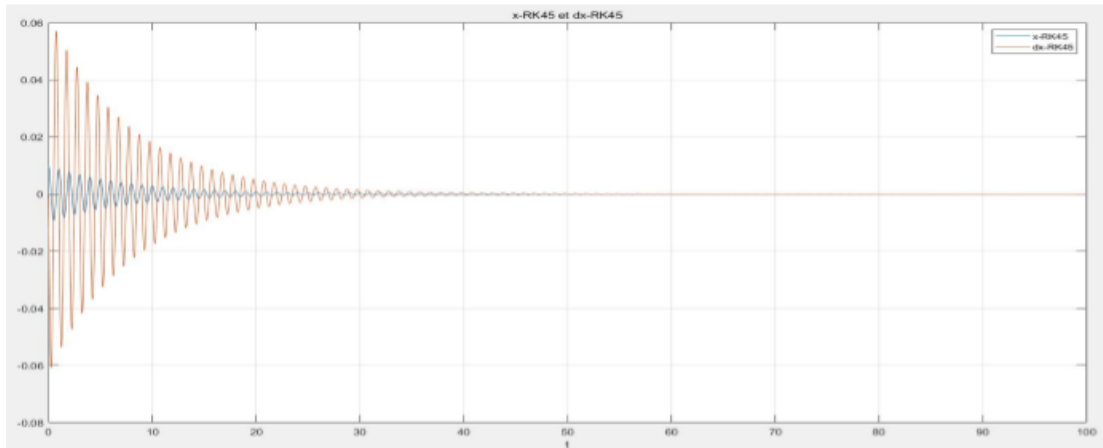
```

```
K=(k1+2*k2+2*k3+k4)/6; y(:,i+1)=y(:,i)+K*dt;
```

```
end
```

```
end
```

Conclusion: Quand $h=0.04, h=0.96, h=1.04$, les dessins sont comme ca. On voit bien que quand $h < h_c$, la solution converge, plus h est petit, plus la convergence est lente. Quand $h > h_c$, la solution diverge, plus h est grand, plus la divergence est vite. Par dichotomie, on obtient approximative du pas de temps critique: $h_c=1.013$, avec $h_{max}=1.014$, $h_{min}=1.013$



Etude d'un double pendule avec l'hypothèse des petits mouvements EX1 Résolution avec un schéma de NEWMARK explicite

1.1

La programmation en MATLAB est comme ca :

```
%% Calcul de q en Newmark explicite
U_Newexp(1,1)=theta10;
U_Newexp(2,1)=theta20;
U_Newexp(3,1)=dtheta10;
U_Newexp(4,1)=dtheta20;
n=floor(T0/dt);
for t=0:dt:dt*n
    B=m*a^2*[2 1;1 1];
    C=m*g*a*[2 0;0 1];
    D=F0*sin(w*t).*[a;a/sqrt(2)];
    U_Newexp(5:6,1)=inv(B)*(D-C*[theta10;theta20]);
    % équation du mouvement: Bq''+Cq=D;
    E=[dt/2*inv(B)*D ;
    dt*inv(B)*D-dt^2/4*inv(B)*C*inv(B)*D];
    A=[eye(2)-dt^2/2*inv(B)*C , eye(2)*dt ; -dt*inv(B)*C+dt^3/4*(inv(B)*C)^2 , eye(2)-
    dt^2/2*inv(B)*C];
    for i=2:n+1
        U_Newexp(1:4,i)=A*U_Newexp(1:4,i-1)+E;
        U_Newexp(5:6,i)=inv(B)*(D-C*[U_Newexp(1,i);
        U_Newexp(2, i)]);
        %U(j+1)=A*U(j)+E , Bq''+Cq=D;
    end
end
```

Conclusion: Ici on note que l'équation du mouvement: $Bq''+Cq=D$, et que $q=[\theta_1; \theta_2]$, $q(j+1)=A*q(j)+E$

Donc, on peut obtenir la matrice d'amplification :

```
A=[eye(2)-dt^2/2*inv(B)*C, eye(2)*dt; -dt*inv(B)*C+dt^3/4*(inv(B)*C)^2, eye(2)-
dt^2/2*inv(B)*C];
```

avec $E=[dt/2*inv(B)*D; dt*inv(B)*D-dt^2/4*inv(B)*C*inv(B)*D]$;

1.2

La programmation en MATLAB est comme ca :

```
%% Matrice A
```

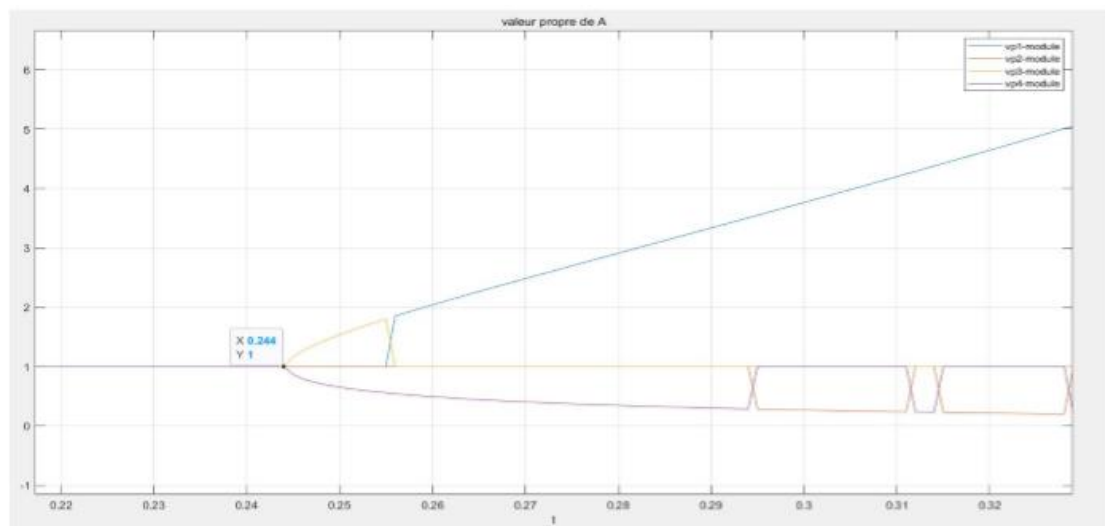
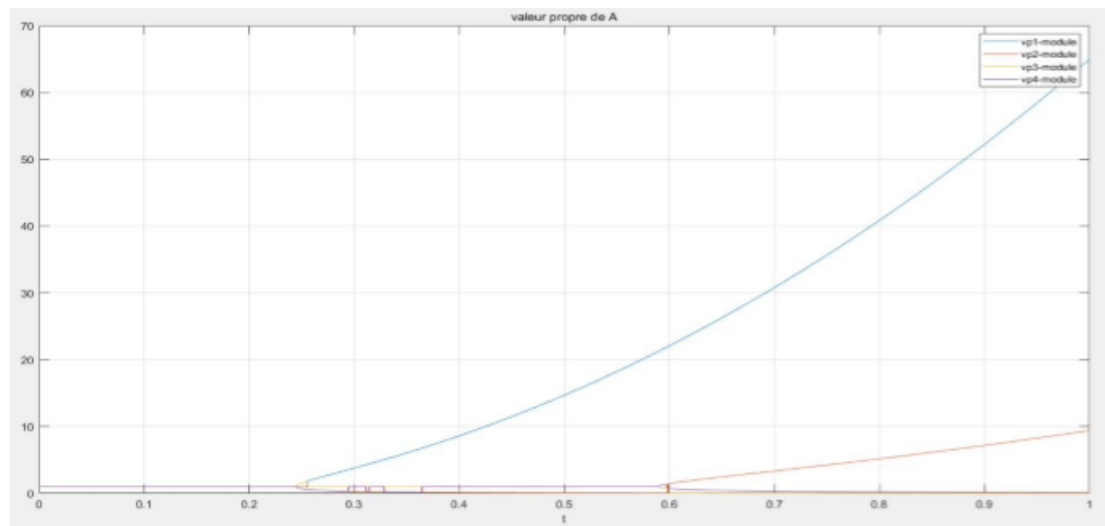
```
dt=0:0.001:1;
```

```

for i=1:length(dt)
B=m*a^2*[2 1;1 1];
C=m*g*a*[2 0;0 1];
Anew{1,i}=[eye(2)-dt(i)^2/2*inv(B)*C , eye(2)*dt(i) ; -
dt(i)*inv(B)*C+dt(i)^3/4*(inv(B)*C)^2 ,
eye(2)-dt(i)^2/2*inv(B)*C];
Enew{1,i}=eig(Anew{1,i});
vp1_module(1,i)=norm(Enew{1,i}(1,1)); vp2_module(1,i)=norm(Enew{1,i}(2,1));
vp3_module(1,i)=norm(Enew{1,i}(3,1)); vp4_module(1,i)=norm(Enew{1,i}(4,1));
end

```

On peut voir que on peut obtenir le pas de temps critique est $dt=0.244$.



1.3

$$m*a^2*[21;11]*q''(0)+m*g*a*[20;01]*q(0)=F0*\sin(wt)*[a;a/\sqrt{2}];$$

1.4

$$q(j+1)=q(j)+dt*q'(j)+dt^2/2*q''(j);$$

$$m \cdot a^2 \cdot [21; 11] \cdot q''(j) + m \cdot g \cdot a \cdot [20; 01] \cdot q(j) = F_0 \cdot \sin(\omega t) \cdot [a; a/\sqrt{2}];$$

$$q'(j+1) = q'(j) + dt/2 \cdot q''(j) + dt/2 \cdot q''(j+1);$$

1.5

clc

clear all

close all

%% Condition initiale

syms t

m=2;

a=0.5;

g=9.81;

F0=20;

w=2*pi;

theta10=0;

theta20=0;

dtheta10=-1.31519275;

dtheta20=-1.85996342;

T0=8;

dt=0.02;

gamma=0.5;

beta=0;

%% Calcul de q en Newmark explicite

U_Newexp(1,1)=theta10;

U_Newexp(2,1)=theta20;

U_Newexp(3,1)=dtheta10;

U_Newexp(4,1)=dtheta20;

n=floor(T0/dt);

for t=0:dt:dt*n

B=m*a^2*[2 1;1 1];

C=m*g*a*[2 0;0 1];

D=F0*sin(w*t).*[a;a/sqrt(2)];

U_Newexp(5:6,1)=inv(B)*(D-C*[theta10;theta20]);

% équation du mouvement: Bq''+Cq=D;

E=[dt/2*inv(B)*D ;

dt*inv(B)*D-dt^2/4*inv(B)*C*inv(B)*D];

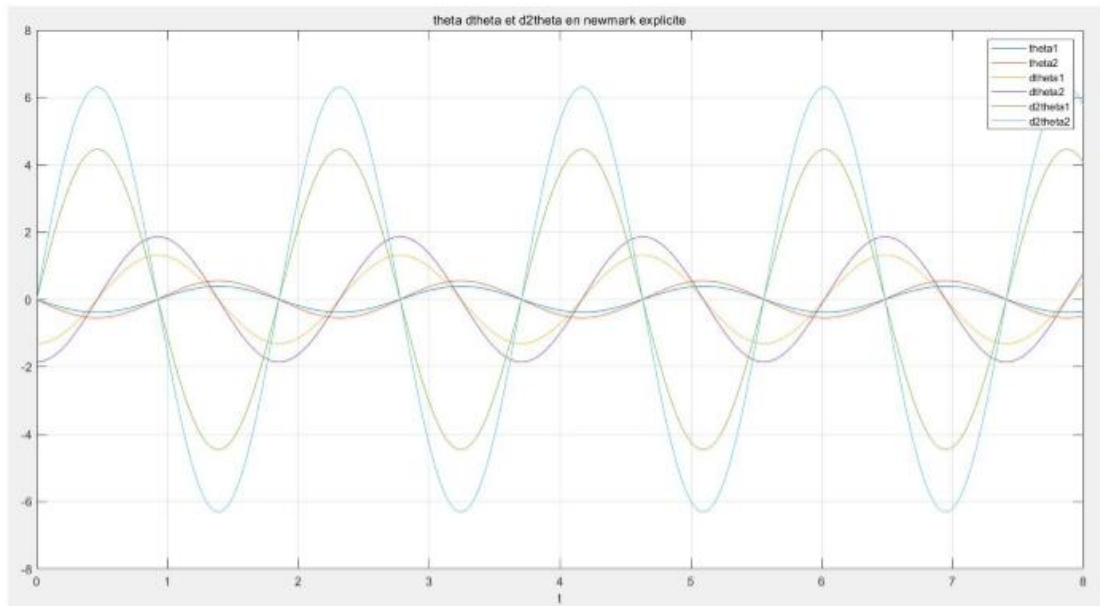
A=[eye(2)-dt^2/2*inv(B)*C , eye(2)*dt ;

-dt*inv(B)*C+dt^3/4*(inv(B)*C)^2 ,

```

eye(2)-dt^2/2*inv(B)*C];
for i=2:n+1
U_Newexp(1:4,i)=A*U_Newexp(1:4,i-1)+E;
U_Newexp(5:6,i)=inv(B)*(D-C*[U_Newexp(1,i);
U_Newexp(2, i)]);
%U(j+1)=A*U(j)+E , Bq''+Cq=D;
end
end

```



1.6

% en 0s, dt, 2*dt, 0.5s

```
U=U_Newexp(:,[1 2 3 26])
```

U=

```
[0, -0.0263, -0.0525, -0.3852; 0, -0.0372, -0.0742, -0.5447; -1.3152, -1.3122, -1.3031,
0.1635; -1.8600, -1.8557, -1.8429, 0.2312; -0.0000, 0.3023, 0.6032, 4.4267; -0.0000,
0.4275, 0.8531, 6.2603]
```

EX2 Résolution avec un schéma de NEWMARK implicite

2.1

%% Calcul de q en Newmark implicite

```
U_Newimp(1,1)=theta10;
```

```
U_Newimp(2,1)=theta20;
```

```
U_Newimp(3,1)=dtheta10;
```

```
U_Newimp(4,1)=dtheta20;
```

```
n=floor(T0/dt);
```

```

for t=0:dt:dt*n
B=m*a^2*[2 1;1 1];
C=m*g*a*[2 0;0 1];
D=F0*sin(w*t).*[a;a/sqrt(2)];
U_Newimp(5:6,1)=inv(B)*(D-C*[theta10;theta20]);
% U_Newimp(5:6,1)=[0;0];
% équation du mouvement: Bq''+Cq=D;
E=[eye(2)+1/4*dt^2*inv(B)*C];
F=[E zeros(2) ; zeros(2) eye(2)];
G=[eye(2)-1/4*dt^2*inv(B)*C , eye(2)*dt ; -1/2*dt*inv(B)*C*(eye(2)+inv(E))-
1/4*dt^2*inv(E)*inv(B)*C , eye(2)-1/2*dt^2*inv(B)*C*inv(E) ];
H=[1/2*dt^2*inv(B)*D ;
dt*inv(B)*D-1/4*dt^3*inv(B)*C*inv(E)*inv(B)*D];
A=inv(F)*G;
%F*U(j+1)=G*U(j)+H
for i=2:n+1
U_Newimp(1:4,i)=A*U_Newimp(1:4,i-1)+inv(F)*H;
U_Newimp(5:6,i)=inv(B)*(D-C*[U_Newimp(1,i);U_Newimp(2, i)]);
%U(j+1)=A*U(j)+E , Bq''+Cq=D;
end
end

```

Conclusion: On note ici l'équation du mouvement: $Bq''+Cq=D$. Et on note que $q=[\theta_1;\theta_2]$, $F*q(j+1)= G*q(j)+H$

Donc, on peut obtenir la matrice d'amplification $A=inv(F)*G$; avec $H=[1/2*dt^2*inv(B)*D;dt*inv(B)*D-1/4*dt^3*inv(B)*C*inv(E)*inv(B)*D];$

2.2

%% Matrice A

```

dt=0:0.001:1;
for i=1:length(dt)
B=m*a^2*[2 1;1 1];
C=m*g*a*[2 0;0 1];
Enew{1,i}=[eye(2)+1/4*dt(i)^2*inv(B)*C];
Fnew{1,i}=[Enew{1,i} zeros(2) ; zeros(2) eye(2)];
Gnew{1,i}=[eye(2)-1/4*dt(i)^2*inv(B)*C , eye(2)*dt(i) ; -
1/2*dt(i)*inv(B)*C*(eye(2)+inv(Enew{1,i}))-1/4*dt(i)^2*inv(Enew{1,i})*inv(B)*C ,
eye(2)-1/2*dt(i)^2*inv(B)*C*inv(Enew{1,i}) ];
Anew{1,i}=inv(Fnew{1,i})*Gnew{1,i};

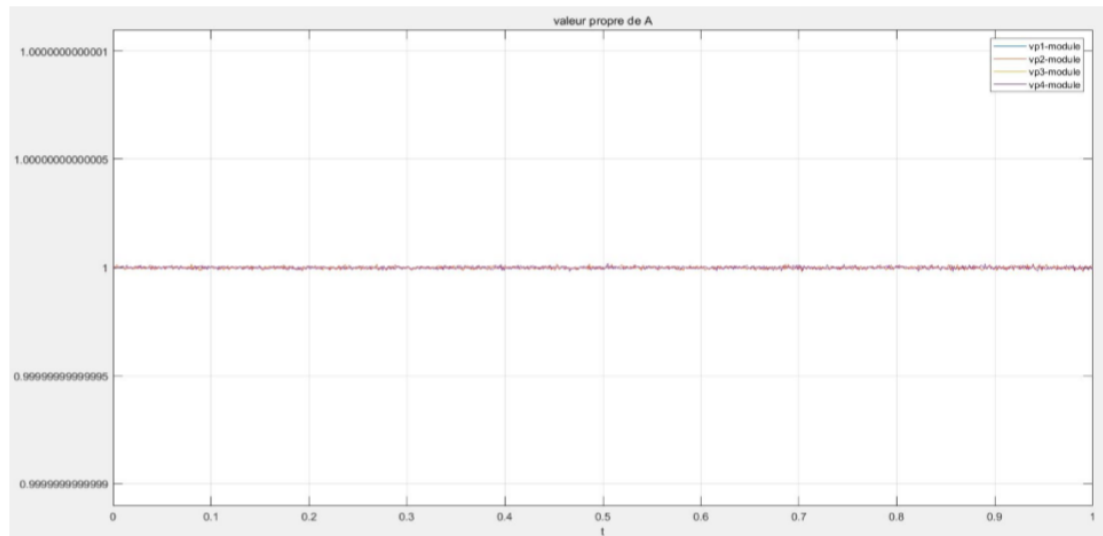
```

```

Enew{1,i}=eig(Anew{1,i});
vp1_module(1,i)=norm(Enew{1,i}(1,1));
vp2_module(1,i)=norm(Enew{1,i}(2,1));
vp3_module(1,i)=norm(Enew{1,i}(3,1));
vp4_module(1,i)=norm(Enew{1,i}(4,1));
end

```

Conclusion : On voit que il n'y a pas le pas de temps critique, parce que les modules des valeurs propres de A sont environ 1.



2.3

$$m \cdot a^2 \cdot [21; 11] \cdot q''(0) + m \cdot g \cdot a \cdot [20; 01] \cdot q(0) = F_0 \cdot \sin(\omega t) \cdot [a; a/\sqrt{2}];$$

2.4

$$q(j+1) = q(j) + dt \cdot q'(j) + dt^2/4 \cdot q''(j) + dt^2/4 \cdot q''(j+1);$$

$$m \cdot a^2 \cdot [21; 11] \cdot q''(j) + m \cdot g \cdot a \cdot [20; 01] \cdot q(j) = F_0 \cdot \sin(\omega t) \cdot [a; a/\sqrt{2}];$$

$$q'(j+1) = q'(j) + dt/2 \cdot q''(j) + dt/2 \cdot q''(j+1);$$

2.5

```
clc
```

```
clear all
```

```
close all
```

```
%% Condition initiale
```

```
syms t
```

```
m=2;
```

```
a=0.5;
```

```
g=9.81;
```

```
F0=20;
```

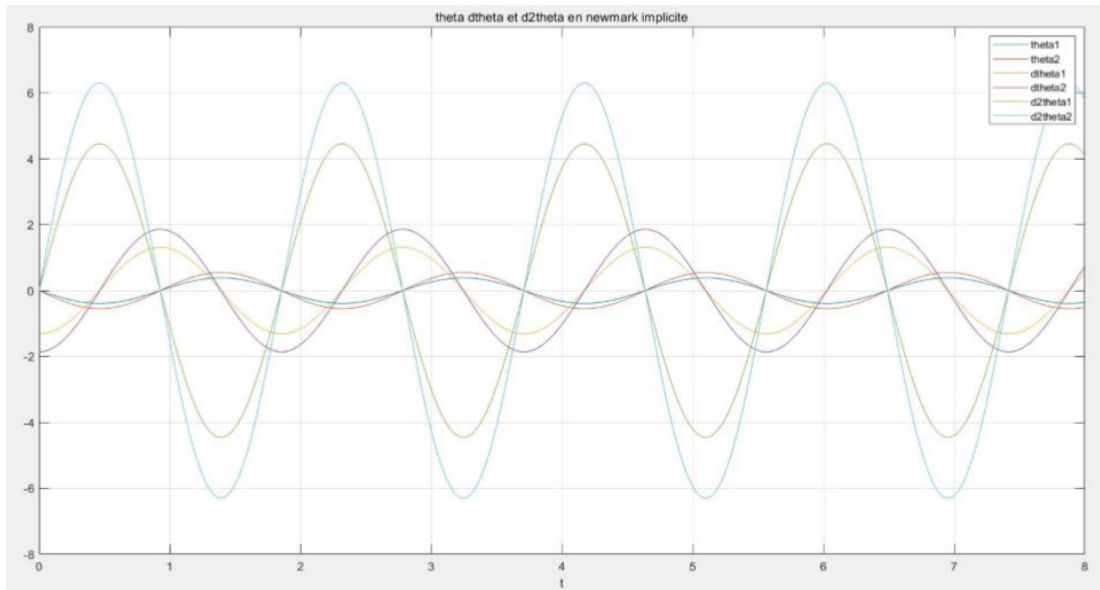
```
w=2*pi;
```

```
theta10=0;
```

```

theta20=0;
dtheta10=-1.31519275;
dtheta20=-1.85996342;
T0=8;
dt=0.02;
gamma=0.5;
beta=0.25;
%% Calcul de q en Newmark implicite
U_Newimp(1,1)=theta10;
U_Newimp(2,1)=theta20;
U_Newimp(3,1)=dtheta10;
U_Newimp(4,1)=dtheta20;
n=floor(T0/dt);
for t=0:dt:dt*n
B=m*a^2*[2 1;1 1];
C=m*g*a*[2 0;0 1];
D=F0*sin(w*t).*[a;a/sqrt(2)];
U_Newimp(5:6,1)=inv(B)*(D-C*[theta10;theta20]);
% U_Newimp(5:6,1)=[0;0];
% équation du mouvement: Bq''+Cq=D;
E=[eye(2)+1/4*dt^2*inv(B)*C];
F=[E zeros(2) ; zeros(2) eye(2)];
G=[eye(2)-1/4*dt^2*inv(B)*C , eye(2)*dt ; -1/2*dt*inv(B)*C*(eye(2)+inv(E))-
1/4*dt^2*inv(E)*inv(B) *C) , eye(2)-1/2*dt^2*inv(B)*C*inv(E) ];
H=[1/2*dt^2*inv(B)*D ; dt*inv(B)*D-1/4*dt^3*inv(B)*C*inv(E)*inv(B)*D];
A=inv(F)*G;
%F*U(j+1)=G*U(j)+H
for i=2:n+1
U_Newimp(1:4,i)=A*U_Newimp(1:4,i-1)+inv(F)*H;
U_Newimp(5:6,i)=inv(B)*(D-C*[U_Newimp(1,i);U_Newimp(2, i)]);
%U(j+1)=A*U(j)+E , Bq''+Cq=D;
end
end

```



2.6

% en 0s, dt, 2*dt, 0.5s

U=U_Newexp(:,[1 2 3 26])

U=

[0, -0.0263, -0.0524, -0.3850;

0, -0.0372, -0.0741, -0.5444;

-1.3152, -1.3122, -1.3031, 0.1622;

-1.8600, -1.8557, -1.8429, 0.2294;

-0.0000, 0.3020, 0.6025, 4.4247;

-0.0000, 0.4270, 0.8521, 6.2574]

Oscillateur non linéaire à un degré de liberté

EX1 Résolution avec un schéma de Newmark explicite

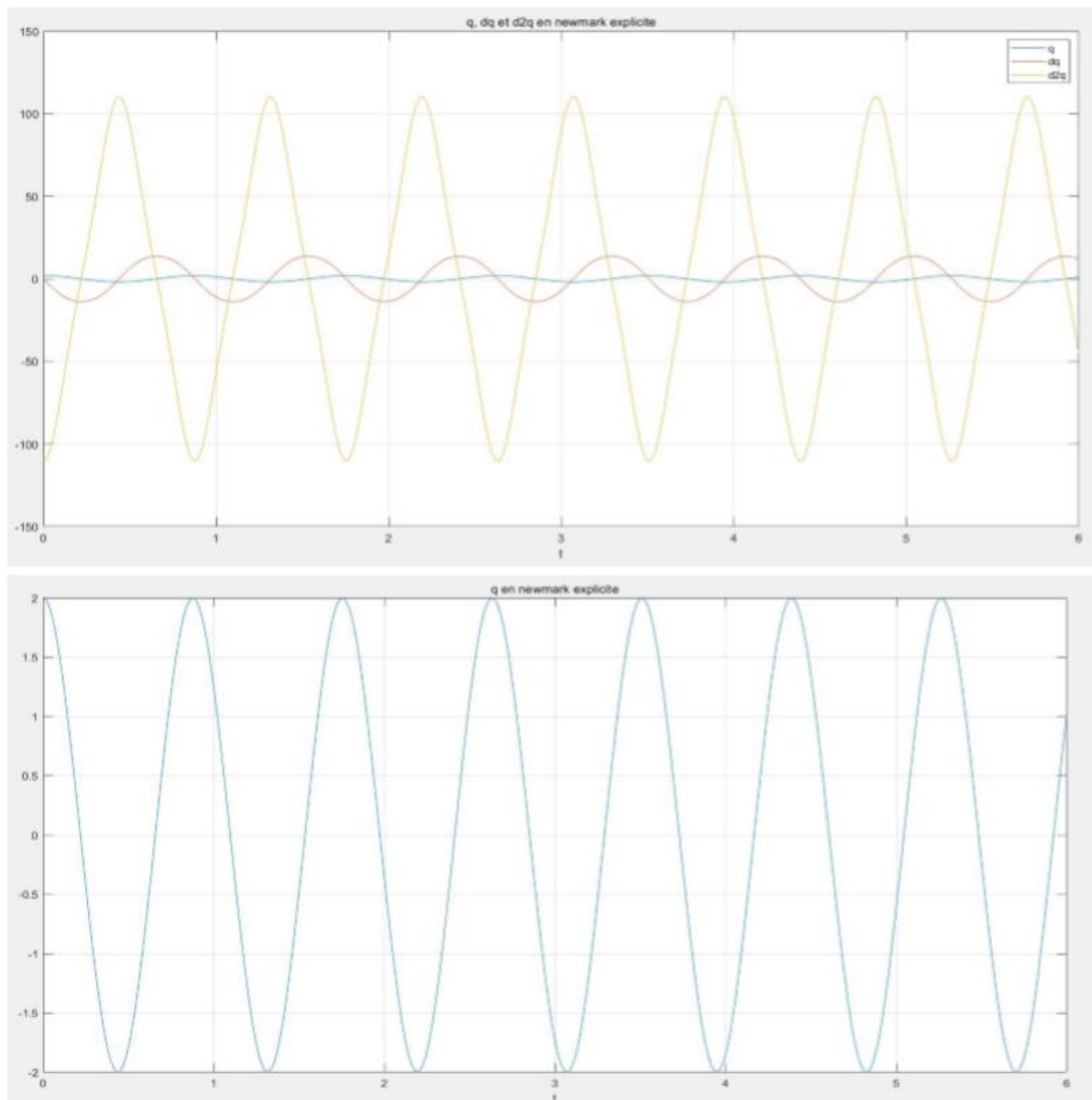
1.1

$$q(j+1)=q(j)+dt*q'(j)+dt^2/2*q''(j);$$
$$q''(j+1)=-w0^2*q(j+1)*(1+a*q(j+1)^2);$$
$$q'(j+1)=q'(j)+dt/2*q''(j)+dt/2*q''(j+1);$$

1.2

Programmation en MATLAB est comme ca:

```
clc
clear all
close all
%% Condition initiale
a=0.1;
w0=2*pi;
q0=2;
dq0=0;
T0=6;
dt=0.02;
gamma=0.5;
beta=0;
%% Calcul de q en Newmark explicite
U_Newexp(1,1)=q0;
U_Newexp(2,1)=dq0;
U_Newexp(3,1)=-w0^2*q0*(1+a*q0^2);
E(1,1)=1/2*w0^2*U_Newexp(1,1)^2+1/4*w0^2*a*U_Newexp(1,1)^4+1/2*U_Newexp(
    2,1)^2;
n=floor(T0/dt);
for i=2:n+1
    U_Newexp(1,i)=U_Newexp(1,i-1)+dt*U_Newexp(2,i-1)+dt^2/2*U_Newexp(3,i-1);
    U_Newexp(3,i)=-w0^2*U_Newexp(1,i)*(1+a*(U_Newexp(1,i))^2);
    U_Newexp(2,i)=U_Newexp(2,i-1)+dt/2*U_Newexp(3,i-1)+dt/2*U_Newexp(3,i);
    E(1,i)=1/2*w0^2*U_Newexp(1,i)^2+1/4*w0^2*a*U_Newexp(1,i)^4+1/2*U_Newexp(
        2,i)^2;
end
1.3
% q en 0s, dt, 2*dt, T0
U=U_Newexp(1,[1 2 3 n+1])
U = [2.0000, 1.9779, 1.9123, 1.0329]
```



EX2 Résolution avec un schéma de Newmark implicite

2.1

Il faut minimiser le résidu, $\text{residu} = \text{abs}((q''^*) + w_0^2(q^*)(1 + a(q^*)^2))$;

2.2

$$(q''^*(j+1)) = (q''^*(j+1)) + \Delta t q''(j+1);$$

$$f(q''^*(j+1), q^*(j+1)) = (q''^*) + w_0^2(q^*)(1 + a(q^*)^2);$$

$$\Delta t q''(j+1) = -f(q''^*(j+1), q^*(j+1)) / (df/dq''^*(j+1) + df/dq^*(j+1) * \beta * \Delta t^2);$$

2.3

Programmation en MATLAB esi comme ca:

```
clc
```

```
clear all
```

```
close all
```

```
%% Condition initiale
```

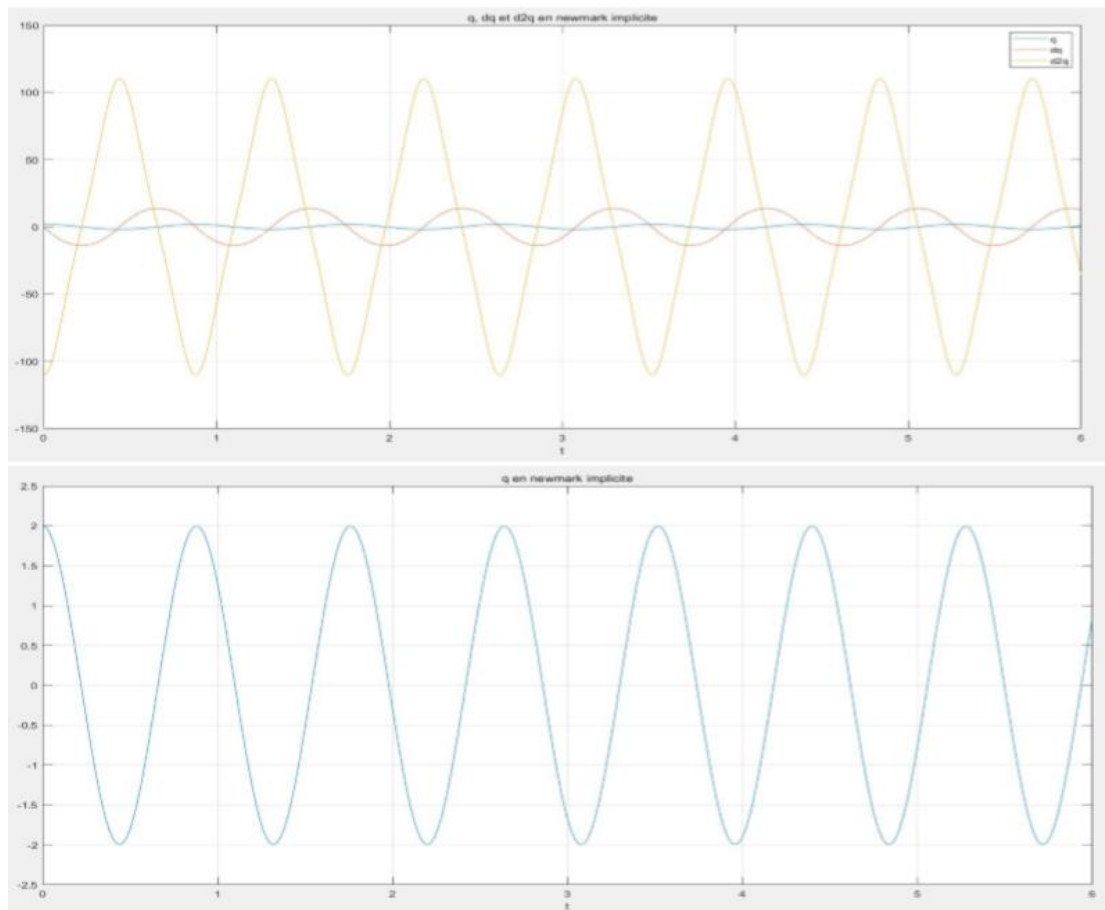
```
a=0.1;
```

```
w0=2*pi;
```

```

q0=2;
dq0=0;
T0=6;
dt=0.02;
gamma=0.5;
beta=0.25;
%% Etat initial
U_Newimp(1,1)=q0;
U_Newimp(2,1)=dq0;
U_Newimp(3,1)=-w0^2*q0*(1+a*q0^2);
E(1,1)=1/2*w0^2*U_Newimp(1,1)^2+1/4*w0^2*a*U_Newimp(1,1)^4+1/2*U_Newi
mp(2,1)^2;
n=floor(T0/dt);
e=0.0001;
%% Calcul de q* en Newmark implicite
for i=2:n+1
    U_Newimp(3,i)=0;
    U_Newimp(2,i)=U_Newimp(2,i-1)+dt*(1-gamma)*U_Newimp(3, i-1);
    U_Newimp(1,i)=U_Newimp(1,i-1)+dt*U_Newimp(2,i-1)+dt^2 *(0.5-
    beta)*U_Newimp(3,i-1);
    residu=abs(U_Newimp(3,i)+w0^2*U_Newimp(1,i)*(1+a*(U_N ewimp(1,i))^2));
    while residu>=e
        f=U_Newimp(3,i)+w0^2*U_Newimp(1,i)*(1+a*(U_Newimp(1,i) )^2);
        deltad2q=-f/(1+beta*dt^2*(w0^2+3*w0^2*a*(U_Newimp(1,i) )^2));
        deltaq=beta*dt^2*deltad2q;
        deltadq=gamma*dt*deltad2q;
        U_Newimp(1,i)=U_Newimp(1,i)+deltaq;
        U_Newimp(2,i)=U_Newimp(2,i)+deltadq;
        U_Newimp(3,i)=U_Newimp(3,i)+deltad2q;
        residu=abs(U_Newimp(3,i)+w0^2*U_Newimp(1,i)*(1+a*(U_N ewimp(1,i))^2));
        E(1,i)=1/2*w0^2*U_Newimp(1,i)^2+1/4*w0^2*a*U_Newimp(1,i)^4+1/2*U_Newimp
        (2,i)^2;
    end
end
end

```



2.4

% q en 0s, dt, 2*dt, T0

U=U_Newimp(1,[1 2 3 n+1])

U = [2.0000, 1.9781, 1.9131, 0.8492]

EX3 Energie mécanique

3.1

$$E_c = \frac{1}{2} m \ddot{q}^2$$

$$E_p = \int F dq$$

$$E = E_c + E_p = \frac{1}{2} m \ddot{q}^2 + \frac{1}{2} k q^2 + \frac{1}{4} k a q^4 + \text{cte.}$$

3.2

On suppose que cte=0 et m=1, en newmark explicite

```
E(1,1)=1/2*w0^2*U_Newexp(1,1)^2+1/4*w0^2*a*U_Newexp(1,1)^4+1/2*U_Newexp(2,1)^2;
```

```
for i=2:n+1
```

```
E(1,i)=1/2*w0^2*U_Newexp(1,i)^2+1/4*w0^2*a*U_Newexp(1,i)^4+1/2*U_Newexp(2,i)^2;
```

```
end
```

```
Emoyenne=mean(E(1,:))
```

On suppose que $\text{cte}=0$ et $m=1$, en newmark implicite

```
E(1,1)=1/2*w0^2*U_Newimp(1,1)^2+1/4*w0^2*a*U_Newimp(1,1)^4+1/2*U_Newimp(2,1)^2;
```

```
for i=2:n+1
```

```
E(1,i)=1/2*w0^2*U_Newimp(1,i)^2+1/4*w0^2*a*U_Newimp(1,i)^4+1/2*U_Newimp(2,i)^2;
```

```
end
```

```
Emoyenne=mean(E(1,:))
```

3.3

Quand $\text{dt}=0.02$, en newmark explicite : $\text{Emoyenne}=94.4561$

Quand $\text{dt}=0.02$, en newmark implicite : $\text{Emoyenne}=94.8102$

Conclusion: L'énergie mécanique est presque une constante, car il n'y a pas d'amortissement dans ce modèle. Parce que les solutions des deux schémas ne sont pas précis, donc l'énergie mécanique oscille un peu, et l'énergie moyenne en newmark implicite est un peu plus grande que celle en newmark explicite

