L'exercice Oscillateur conservatif à un degré de liberté

5.1 Résolution avec un schéma de NEWMARK $\gamma = 0.5, \beta = 0.25$

5.1.1 D'après la relation de récurrence entre les valeurs du vecteurs d' état aux instants:

$$\begin{bmatrix} q_{j+1} \\ \dot{q}_{j+1} \end{bmatrix} = [A] \begin{bmatrix} q_j \\ \dot{q}_j \end{bmatrix}$$

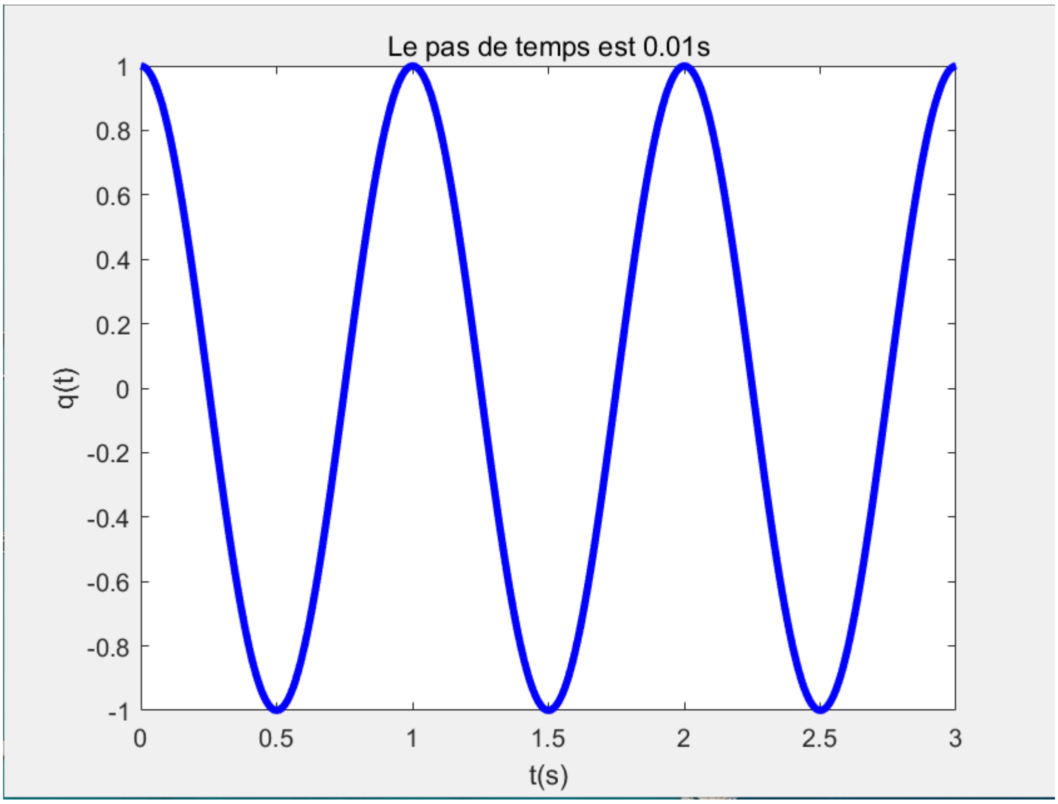
Avec :

$$[A] = [B]^{-1} \times [C],\ [B] = \begin{bmatrix} 1 + \beta \Delta t^2 {\omega_0}^2 & 0 \\ \gamma \Delta t {\omega_0}^2 & 1 \end{bmatrix},\ [C] = \begin{bmatrix} 1 - (0.5 - \beta)\Delta t^2 {\omega_0}^2 & \Delta t \\ -(1-\gamma)\Delta t {\omega_0}^2 & 1 \end{bmatrix}$$

Donc, le code en Matlab est :

```
dt5=0.01;T0=3;
q0=1;dq0=0;w0=2*pi;w0c=w0*w0;
be=0.25;ga=0.5;
t5=(0:dt5:T0)';
np5=size(t5,1);
energie5=zeros(np5,1);
q=[q0;dq0];
q5=zeros(np5,1);
dq5=zeros(np5,1);
dq5(1)=dq0;
q5(1)=q0;
C=[1-(dt5.^2)*(0.5-be)*w0c, dt5;-(1-ga)*dt5*w0c, 1];
B=[1+be*(dt5.^2)*w0c,0;ga*dt5*w0c,1];
A=inv(B)*C;
for inc=2:np5
    q=A*q
    q5(inc)=q(1)
    dq5(inc)=q(2)
end;
plot(t5,q5,'b-','Linewidth',3);
energie2=0.5*(dq5.*dq5+w0c*(q5.^2))
```

Le résultat est :



5.1.2 Code en Matlab :

```
clear all; close all; clc;
%Euler Implicite
%sans matrice d'amplification
dt2=0.01;T0=3;
q0=1;dq0=0;w0=2*pi;w0c=w0*w0;
t2=(0:dt2:T0)'
np2=size(t2,1);
q2=zeros(np2,1);
dq2=zeros(np2,1);
ddq2=zeros(np2,1);
energie2=zeros(np2,1);
q2(1)=q0;dq2(1)=dq0;
for inc=2:np2
    q2(inc)=(q2(inc-1)+dt2*dq2(inc-1))/(1+w0c*dt2*dt2);
    ddq2=-w0c*q2(inc);
    dq2(inc)=dq2(inc-1)+dt2*ddq2;
end;
energie2=0.5*(dq2.*dq2+w0c*(q2.^2));


%Euler Explicite
%sans matrice d'amplification
dt1=0.01;T0=3;
```

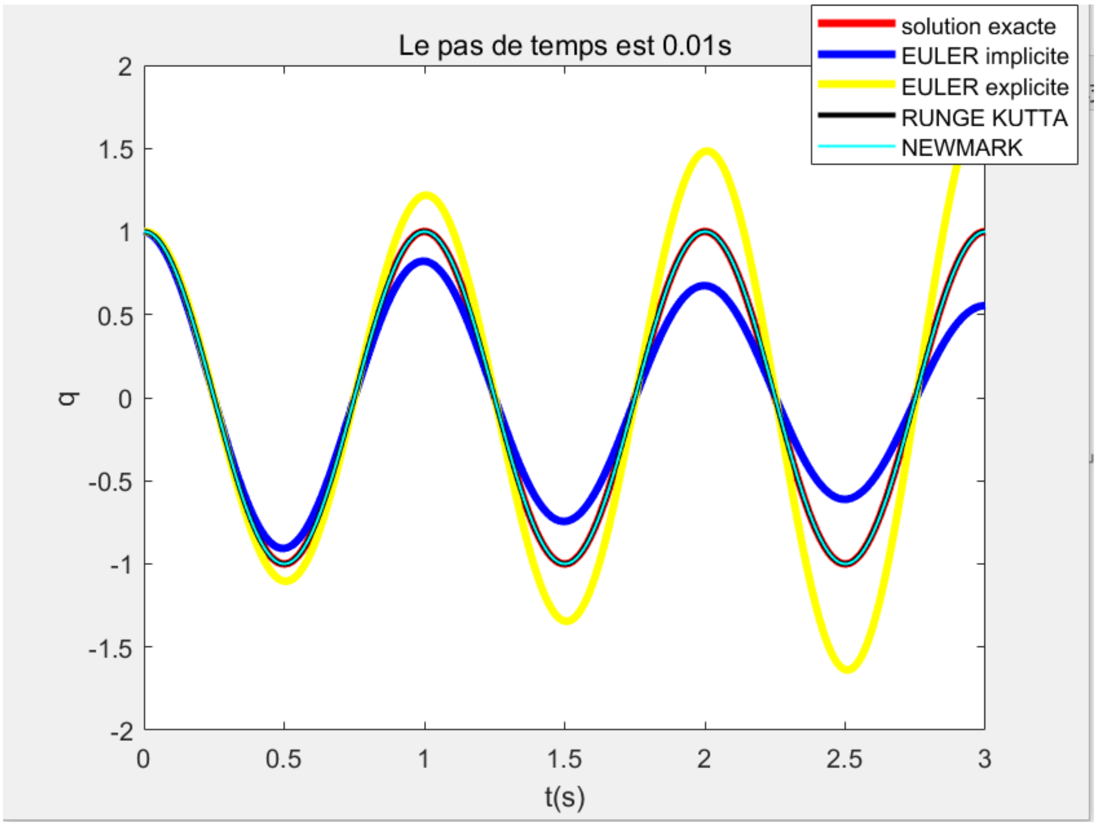
```
t1=(0:dt1:T0)'
np1=size(t1,1);
q1=zeros(np1,1);
dq1=zeros(np1,1);
ddq1=zeros(np1,1);
energie1=zeros(np1,1);
q1(1)=q0;dq1(1)=dq0;
ddq1(1)=-w0c*q1(1);
for inc=2:np1
    q1(inc)=q1(inc-1)+dt1*dq1(inc-1);
    dq1(inc)=dq1(inc-1)+dt1*ddq1(inc-1);
    ddq1(inc)=-w0c*q1(inc)
end;
energie1=0.5*(dq1.*dq1+w0c*(q1.^2));


%Solution exacte
dte=0.01
te=(0:dte:T0)';
npe=size(te,1);
qe=zeros(npe,1);
dqe=zeros(npe,1);
energie=zeros(npe,1);
tic;
qe=q0*cos(w0*te)+dq0/w0*sin(w0*te);
dqe=-w0*q0*sin(w0*te)+dq0*cos(w0*te);
ddqe=-w0c*qe;
energie=0.5*(dqe.*dqe+w0c.*(qe.^2));
toc;


%RUNGE KUTTA
dt4=0.01;T0=3;
t4=(0:dt4:T0)'
np4=size(t4,1)
q4=zeros(np4,1)
dq4=zeros(np4,1)
q4(1)=q0
dq4(1)=dq0
qj=[q0;dq0]
for inc=2:np4
    tc=t4(inc-1);
    xc=qj;
    k1=cal_fc(xc,tc);
```

```
    xc=qj+k1*dt4/2;
    k2=cal_fc(xc,tc+dt4/2);
    xc=qj+k2*dt4/2;
    k3=cal_fc(xc,tc + dt4/2);
    xc=qj + k3 * dt4;
    k4=cal_fc(xc,tc + dt4);
    dq=(k1 + 2 *k2 + 2 * k3 + k4)/6;
    qj=qj + dq * dt4;
    q4(inc)= qj(1);
    dq4(inc)=qj(2);
end
energie4=0.5*(dq4.*dq4+w0c*(q4.^2));


%NEWMARK
dt5=0.01;T0=3;
be=0.25;ga=0.5;
t5=(0:dt5:T0)';
np5=size(t5,1);
energie5=zeros(np5,1);
q=[q0;dq0];
q5=zeros(np5,1);
dq5=zeros(np5,1);
dq5(1)=dq0;
q5(1)=q0;
C=[1-(dt5.^2)*(0.5-be)*w0c, dt5;-(1-ga)*dt5*w0c, 1];
B=[1+be*(dt5.^2)*w0c,0;ga*dt5*w0c,1];
A=inv(B)*C;
for inc=2:np5
    q=A*q
    q5(inc)=q(1)
    dq5(inc)=q(2)
end;
energie5=0.5*(dq5.*dq5+w0c*(q5.^2));

plot(te,qe,'r-','Linewidth',3);
hold on
plot(t2,q2,'b','Linewidth',3);
hold on
plot(t1,q1,'y','Linewidth',3);
hold on
plot(t4,q4,'k-','Linewidth',2);
hold on
plot(t5,q5,'c-','Linewidth',1);
```

hold on

legend('solution exacte','EULER implicite','EULER explicite','RUNGEKUTTA','NEWMARK')
xlabel('t(s)')
ylabel('q')
title('Le pas de temps est 0.01s')

Et on obtient :



On peut voir que lorsque le pas de temps est de 0,01 s, le résultat de Runge Kuta et le résultat de NEWMARK est plus proche de la solution exacte que l'Euler explicite et l'Euler implicite (pour mieux afficher les résultats, on choisit que 'LineWidth' de la valeur exacte est 3 et 'LineWidth' de Runge Kuta est 2, 'LineWidth' de NEWMARK est 1).

5.1.3 Quand $\Delta t = 0.01s$, la quantité E^* de la solution exacte est : $E^* = 19.7392$

Quand $\Delta t = 0.01s$, la quantité E^* associée au schéma d'EULER explicite est :

19.7392 19.8171 19.8954 19.9739 20.0528 20.1319 20.2114 20.2912 20.3713 20.4517
20.5325 20.6135 20.6949 20.7766 20.8586 20.9410 21.0237 21.1066 21.1900 21.2736
21.3576 21.4419 21.5266 21.6116 21.6969 21.7825 21.8685 21.9549 22.0415 22.1286
22.2159 22.3036 22.3917 22.4801 22.5688 22.6579 22.7474 22.8372 22.9273 23.0178
…
62.1262 62.3714 62.6177 62.8649 63.1130 63.3622 63.6123 63.8635 64.1156 64.3687

Quand $\Delta t = 0.01s$, la quantité E^* associée au schéma d'EULER implicite est :

19.7392 19.6616 19.5843 19.5073 19.4306 19.3541 19.2780 19.2022 19.1267 19.0515
18.9766 18.9020 18.8276 18.7536 18.6799 18.6064 18.5332 18.4604 18.3878 18.3155
18.2434 18.1717 18.1002 18.0291 17.9582 17.8876 17.8172 17.7472 17.6774 17.6079

17.5386 17.4696 17.4009 17.3325 17.2644 17.1965 17.1289 17.0615 16.9944 16.9276

…

6.2717 6.2470 6.2225 6.1980 6.1736 6.1494 6.1252 6.1011 6.0771 6.0532

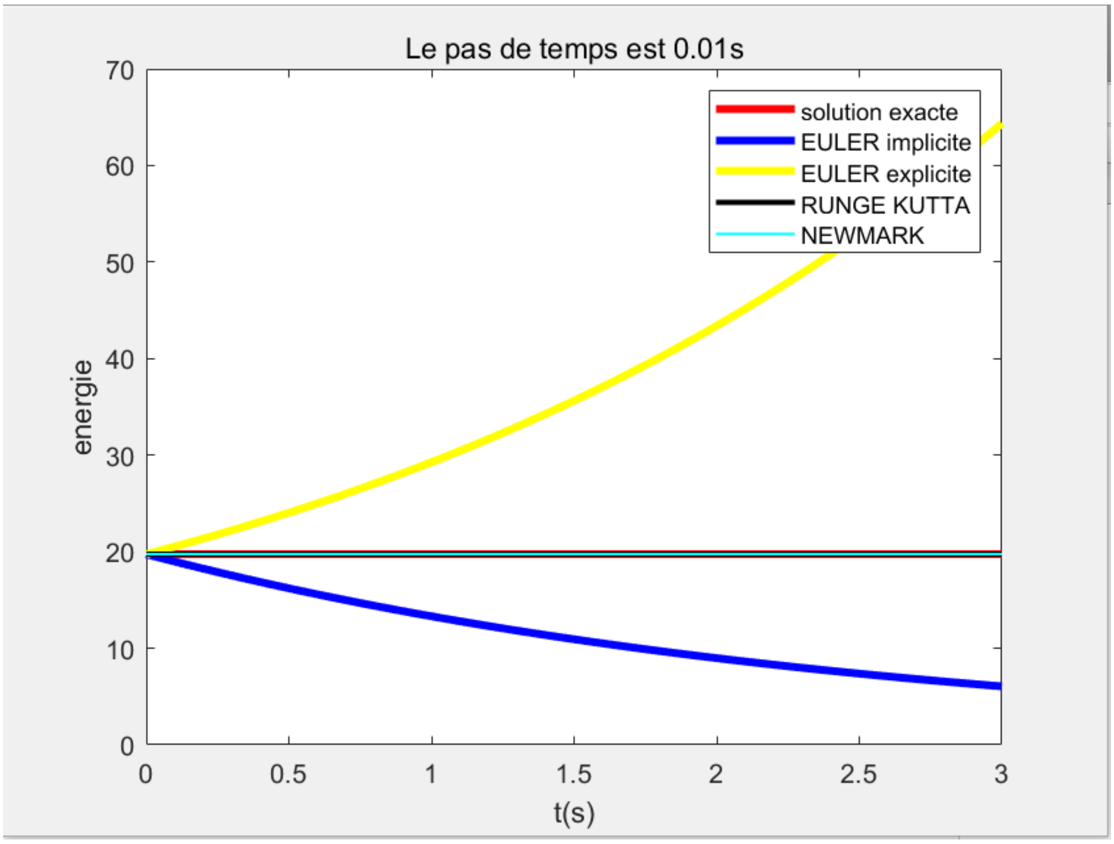
Quand $\Delta t = 0.01s$, la quantité E^* associée au schéma de Runge Kuta est :

19.739208802178716 19.739208785318539 19.739208768458361
19.739208751598191 19.739208734738014 19.739208717877833
19.739208701017660 19.739208684157486 19.739208667297312
19.739208650437142 19.739208633576965 19.739208616716788
…
19.739203777847141 19.739203760986967 19.739203744126797

Quand $\Delta t = 0.01s$, la quantité E^* associée au schéma de NEWMARK est :

19.739208802178716 19.739208802178720 19.739208802178723
19.739208802178723 19.739208802178727 19.739208802178730
19.739208802178737 19.739208802178737 19.739208802178744
19.739208802178752 19.739208802178752 19.739208802178752
…
19.739208802178933 19.739208802178936 19.739208802178940

La figure pour comparer ces quatre résultats est :



On peut voir que lorsque le pas de temps est de 0,01 s, le résultat de Runge Kuta et le résultat de NEWMARK est plus proche de la solution exacte que l'Euler explicite et l'Euler implicite (pour mieux afficher les résultats, on choisit que ‘LineWidth’ de la valeur exacte est 3 et ‘LineWidth’ de Runge Kuta est 2, ‘LineWidth’ de NEWMARK est 1). Le résultat de NEWMARK change moins que celui de RUNGE KUTTA. (Runge Kuta :

$|19.739208802178716 - 19.739203744126797| = 5.06 \times 10^{-6}$,
NEWMARK :$|19.739208802178940 - 19.739208802178716| = 2.2 \times 10^{-13}$)

Donc, on peut choisir le schéma de NEWMARK pour résoudre l' équation (1).

5.1.4 On calcule : $\det(A - \lambda I) = 0$, avec $A = \begin{bmatrix} 1 - \frac{\omega_0{}^2 \Delta t^2}{2(1+\beta\omega_0{}^2\Delta t^2)} & \frac{\Delta t}{1+\beta\omega_0{}^2\Delta t^2} \\ -\omega_0{}^2\Delta t\left[1 - \frac{\gamma\omega_0{}^2\Delta t^2}{2(1+\beta\omega_0{}^2\Delta t^2)}\right] & 1 - \frac{\gamma\omega_0{}^2\Delta t^2}{1+\beta\omega_0{}^2\Delta t^2} \end{bmatrix}$

On obtient que Les valeurs propres de la matrice d'amplification en fonction du pas de temps Δt ($\gamma = 0.5, \beta = 0.25$) sont :

$$\lambda_1 = -\frac{\omega_0\Delta t + 2i}{\omega_0\Delta t - 2i} \qquad \lambda_2 = -\frac{2 + i\omega_0\Delta t}{-2 + i\omega_0\Delta t}$$

Quand $\Delta t = 0s$, $\lambda_1 = \lambda_2 = 1$;

Quand $\Delta t = 0.001s$, $\lambda_1 = 9.9998 \times 10^{-1} + 6.2831 \times 10^{-3} i$
$\lambda_2 = 9.9998 \times 10^{-1} - 6.2831 \times 10^{-3} i$

Quand $\Delta t = 0.005s$, $\lambda_1 = 9.9951 \times 10^{-1} + 3.1408 \times 10^{-2} i$
$\lambda_2 = 9.9951 \times 10^{-1} - 3.1408 \times 10^{-2} i$

Quand $\Delta t = 0.01s$, $\lambda_1 = 9.9803 \times 10^{-1} + 6.2770 \times 10^{-2} i$
$\lambda_2 = 9.9803 \times 10^{-1} - 6.2770 \times 10^{-2} i$

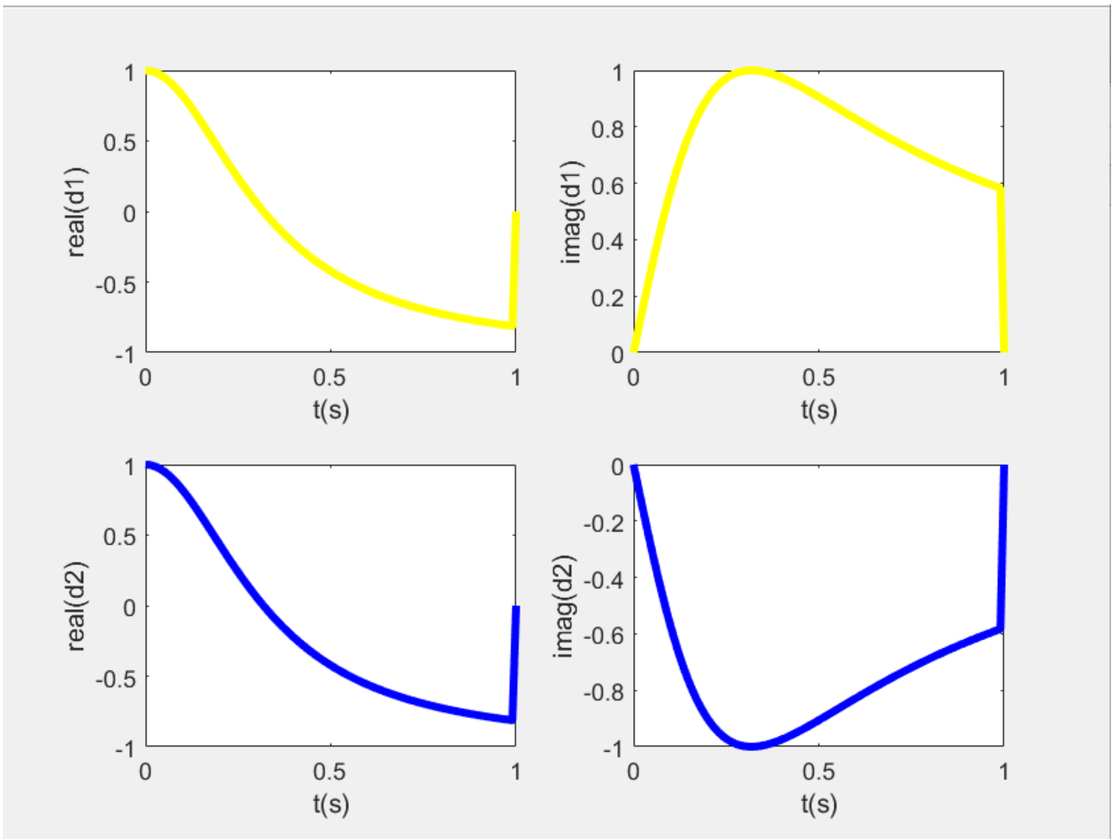
Quand $\Delta t = 0.05s$, $\lambda_1 = 9.5184 \times 10^{-1} + 3.0659 \times 10^{-1} i$
$\lambda_2 = 9.5184 \times 10^{-1} - 3.0659 \times 10^{-1} i$

Quand $\Delta t = 0.1s$, $\lambda_1 = 8.2034 \times 10^{-1} + 5.7188 \times 10^{-1} i$
$\lambda_2 = 8.2034 \times 10^{-1} - 5.7188 \times 10^{-1} i$

Quand $\Delta t = 0.5s$, $\lambda_1 = -4.2320 \times 10^{-1} + 9.0604 \times 10^{-1} i$
$\lambda_2 = -4.2320 \times 10^{-1} - 9.0604 \times 10^{-1} i$

Quand $\Delta t = 1s$, $\lambda_1 = -8.1600 \times 10^{-1} + 5.7805 \times 10^{-1} i$
$\lambda_2 = -8.1600 \times 10^{-1} - 5.7805 \times 10^{-1} i$

On utilise matlab pour tracer les parties réelles et imaginaires des deux valeurs propres au fil du temps :



On peut voir que les courbes de changement de parties réelles des deux valeurs propres sont cohérentes et que les courbes de changement de parties imaginaires sont symétriques.

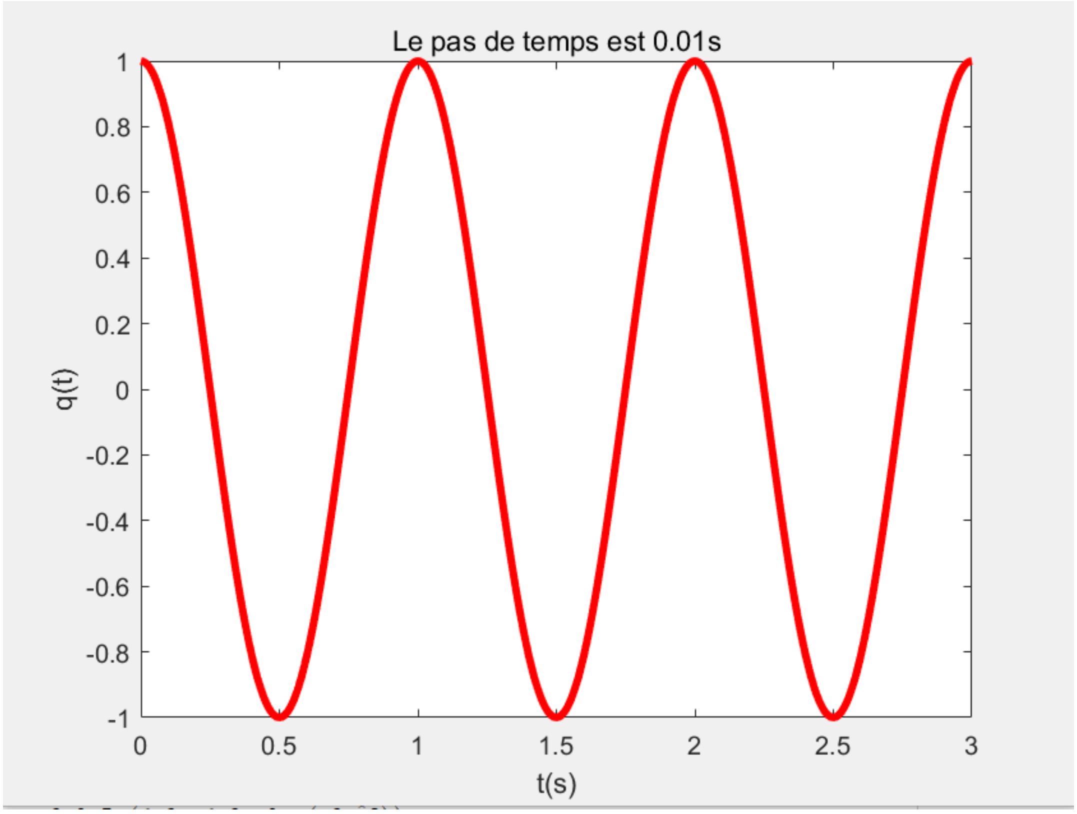
5.2 Résolution avec un schéma de NEWMARK $\gamma = 0.5, \beta = 0$

5.2.1 Code en Matlab :

```
w0=2*pi;
w0c=w0*w0;
dt6=0.001;
T0=3;
t6=(0:dt6:T0)';
np6=size(t6,1);
q6=zeros(np6,1);
dq6=zeros(np6,1);
energie6=zeros(np6,1);
q0=1;
dq0=0;
ddq0=-w0c*q0;
q6(1)=q0;
dq6(1)=dq0;
for inc=2:np6
    q6(inc)=q6(inc-1)+dt6*dq6(inc-1)+dt6*dt6*0.5*ddq0;
    ddq6=-w0c*q6(inc);
    dq6(inc)=dq6(inc-1)+0.5*dt6*(ddq0+ddq6);
```

```
        ddq0=ddq6;
end
energie6=0.5*(dq6.*dq6+w0c*(q6.^2));
plot(t6,q6,'r-','Linewidth',3);
```

Le résultat est :



5.2.2 Code en Matlab :

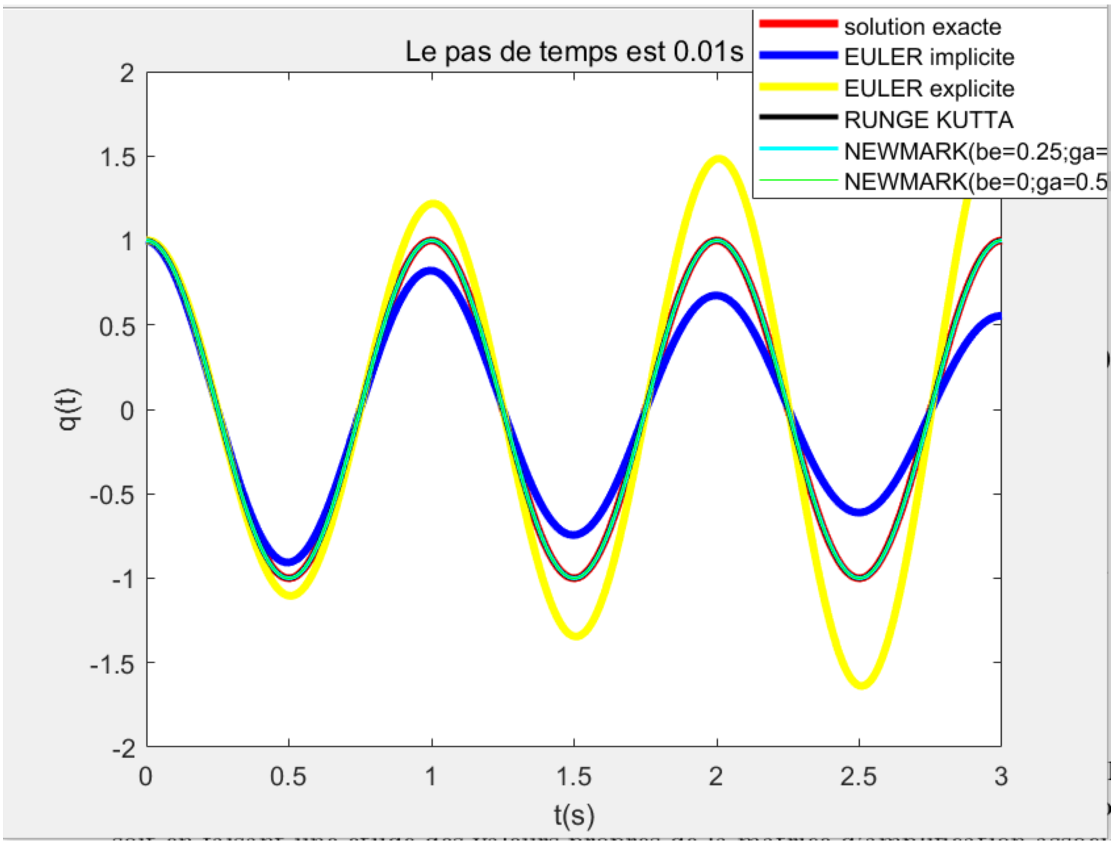
```
clear all; close all; clc;
%Euler Implicite
%sans matrice d'amplification
dt2=0.01;T0=3;
q0=1;dq0=0;w0=2*pi;w0c=w0*w0;
t2=(0:dt2:T0)'
np2=size(t2,1);
q2=zeros(np2,1);
dq2=zeros(np2,1);
ddq2=zeros(np2,1);
energie2=zeros(np2,1);
q2(1)=q0;dq2(1)=dq0;
for inc=2:np2
        q2(inc)=(q2(inc-1)+dt2*dq2(inc-1))/(1+w0c*dt2*dt2);
        ddq2=-w0c*q2(inc);
        dq2(inc)=dq2(inc-1)+dt2*ddq2;
end;
```

```
energie2=0.5*(dq2.*dq2+w0c*(q2.^2));


%Euler Explicite
%sans matrice d'amplification
dt1=0.01;T0=3;
t1=(0:dt1:T0)'
np1=size(t1,1);
q1=zeros(np1,1);
dq1=zeros(np1,1);
ddq1=zeros(np1,1);
energie1=zeros(np1,1);
q1(1)=q0;dq1(1)=dq0;
ddq1(1)=-w0c*q1(1);
for inc=2:np1
    q1(inc)=q1(inc-1)+dt1*dq1(inc-1);
    dq1(inc)=dq1(inc-1)+dt1*ddq1(inc-1);
    ddq1(inc)=-w0c*q1(inc)
end;
energie1=0.5*(dq1.*dq1+w0c*(q1.^2));


%Solution exacte
dte=0.01
te=(0:dte:T0)';
npe=size(te,1);
qe=zeros(npe,1);
dqe=zeros(npe,1);
energie=zeros(npe,1);
tic;
qe=q0*cos(w0*te)+dq0/w0*sin(w0*te);
dqe=-w0*q0*sin(w0*te)+dq0*cos(w0*te);
ddqe=-w0c*qe;
energie=0.5*(dqe.*dqe+w0c.*(qe.^2));
toc;


%RUNGE KUTTA
dt4=0.01;T0=3;
t4=(0:dt4:T0)'
np4=size(t4,1)
q4=zeros(np4,1)
dq4=zeros(np4,1)
q4(1)=q0
```

```
dq4(1)=dq0
qj=[q0;dq0]
for inc=2:np4
    tc=t4(inc-1);
    xc=qj;
    k1=cal_fc(xc,tc);
    xc=qj+k1*dt4/2;
    k2=cal_fc(xc,tc+dt4/2);
    xc=qj+k2*dt4/2;
    k3=cal_fc(xc,tc + dt4/2);
    xc=qj + k3 * dt4;
    k4=cal_fc(xc,tc + dt4);
    dq=(k1 + 2 *k2 + 2 * k3 + k4)/6;
    qj=qj + dq * dt4;
    q4(inc)= qj(1);
    dq4(inc)=qj(2);
end
energie4=0.5*(dq4.*dq4+w0c*(q4.^2));


%NEWMARK(be=0.25;ga=0.5)
dt5=0.01;T0=3;
be=0.25;ga=0.5;
t5=(0:dt5:T0)';
np5=size(t5,1);
energie5=zeros(np5,1);
q=[q0;dq0];
q5=zeros(np5,1);
dq5=zeros(np5,1);
dq5(1)=dq0;
q5(1)=q0;
C=[1-(dt5.^2)*(0.5-be)*w0c, dt5;-(1-ga)*dt5*w0c, 1];
B=[1+be*(dt5.^2)*w0c,0;ga*dt5*w0c,1];
A=inv(B)*C;
for inc=2:np5
    q=A*q
    q5(inc)=q(1)
    dq5(inc)=q(2)
end;
energie5=0.5*(dq5.*dq5+w0c*(q5.^2));


% NEWMARK(be=0;ga=0.5)
dt6=0.01;T0=3;
```

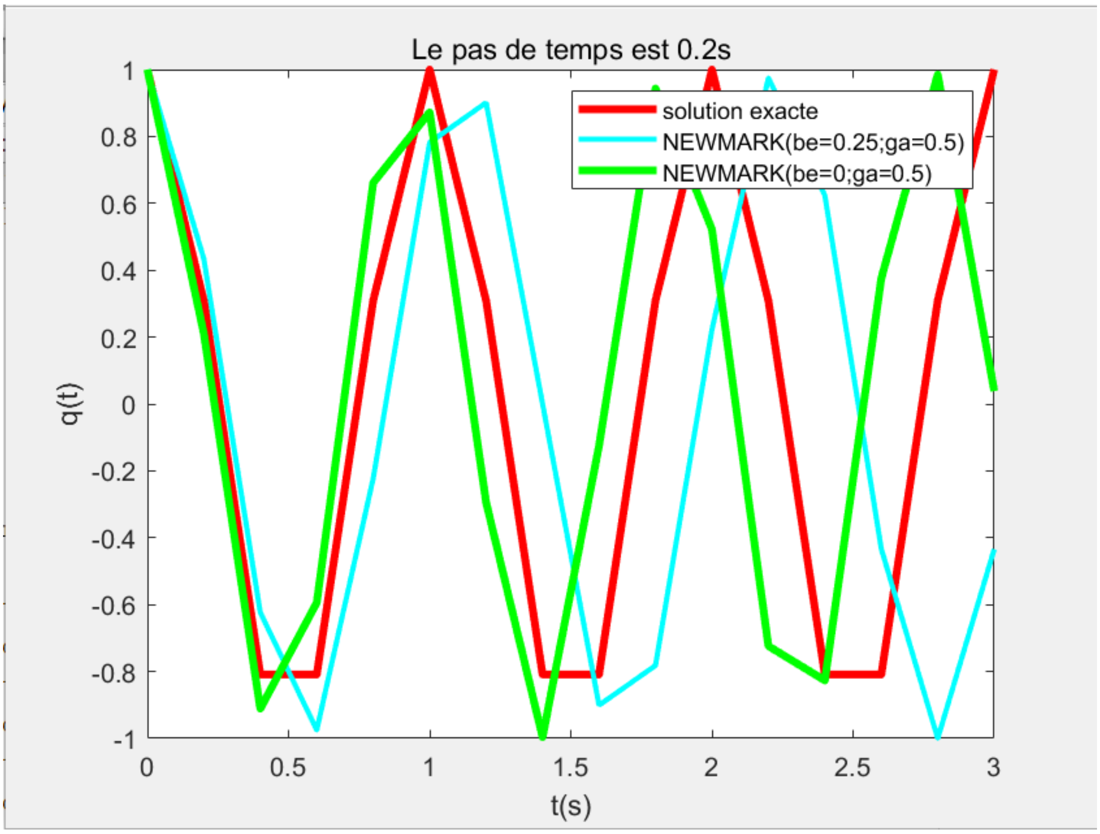
```
t6=(0:dt6:T0)';
np6=size(t6,1);
q6=zeros(np6,1);
dq6=zeros(np6,1);
energie6=zeros(np6,1);
ddq0=-w0c*q0;
q6(1)=q0;
dq6(1)=dq0;
for inc=2:np6
    q6(inc)=q6(inc-1)+dt6*dq6(inc-1)+dt6*dt6*0.5*ddq0;
    ddq6=-w0c*q6(inc);
    dq6(inc)=dq6(inc-1)+0.5*dt6*(ddq0+ddq6);
    ddq0=ddq6;
end
energie6=0.5*(dq6.*dq6+w0c*(q6.^2));


plot(te,qe,'r-','Linewidth',3);
hold on
plot(t2,q2,'b','Linewidth',3);
hold on
plot(t1,q1,'y','Linewidth',3);
hold on
plot(t4,q4,'k-','Linewidth',2);
hold on
plot(t5,q5,'c-','Linewidth',1.5);
hold on
plot(t6,q6,'g-','Linewidth',0.5);
hold on
legend('solution exacte','EULER implicite','EULER explicite','RUNGE KUTTA', 'NEWMARK(be=0.25;ga=0.5)', 'NEWMARK(be=0;ga=0.5)')
xlabel('t(s)')
ylabel('q(t)')
title('Le pas de temps est 0.01s')
```

Et on obtient :

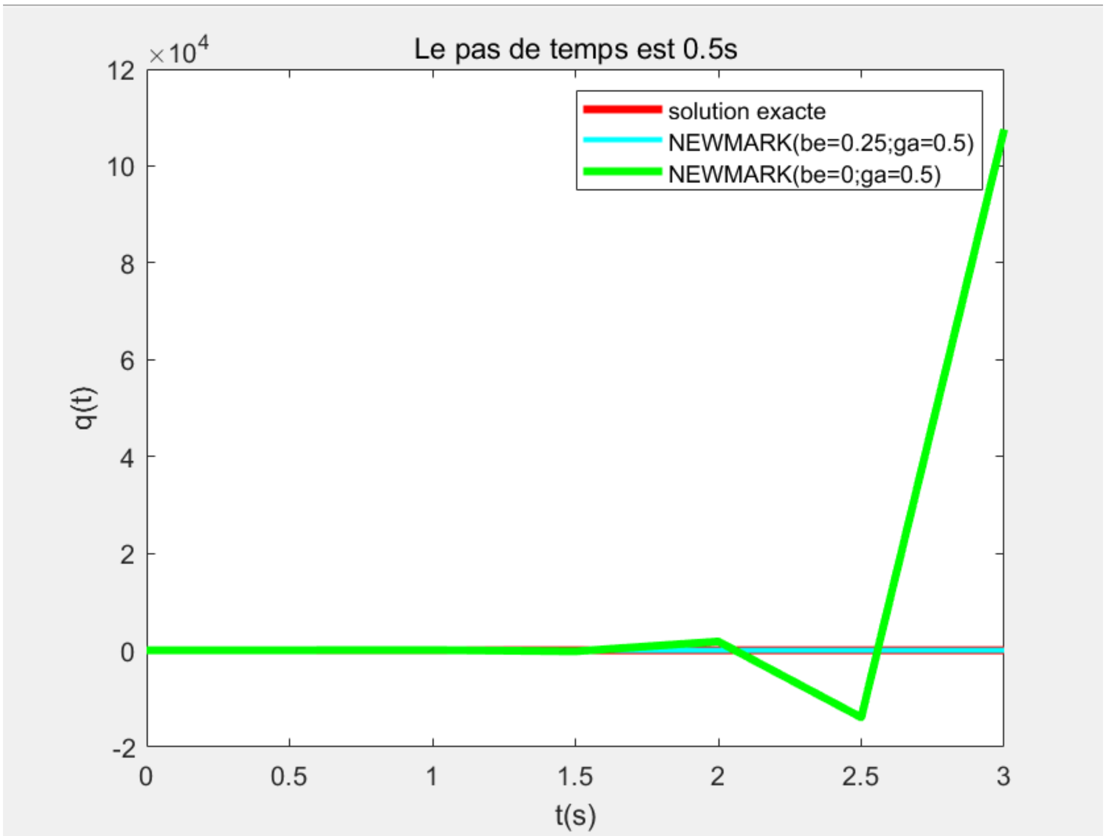


On peut voir que lorsque le pas de temps est de 0,01 s, le résultat de Runge Kuta et le résultat de NEWMARK est plus proche de la solution exacte que l'Euler explicite et l'Euler implicite (pour mieux afficher les résultats, on choisit que ‘LineWidth’ de la valeur exacte est 3 et ‘LineWidth’ de Runge Kuta est 2, ‘LineWidth’ de NEWMARK(be=0.25;ga=0.5) est 1.5, ‘LineWidth’ de NEWMARK(be=0;ga=0.5) est 0.5).

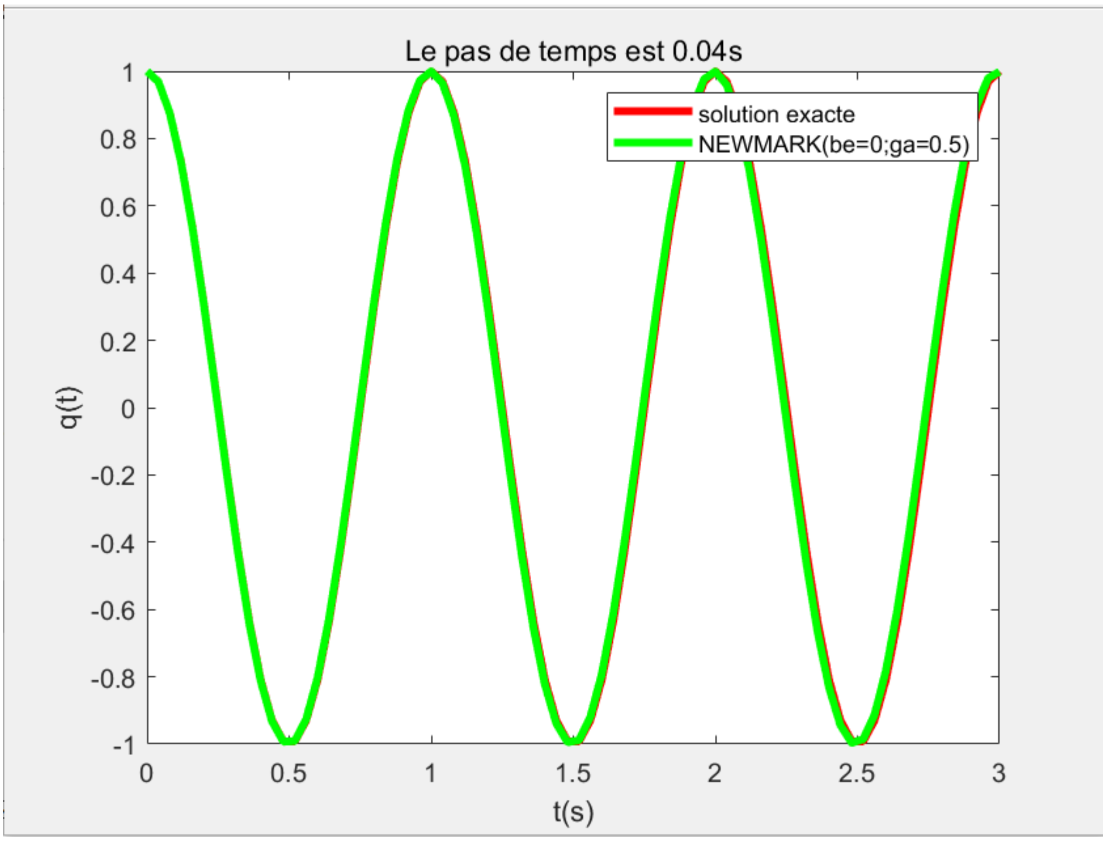
5.2.3 Quand $\Delta t = 0.2s$, le résultat est :



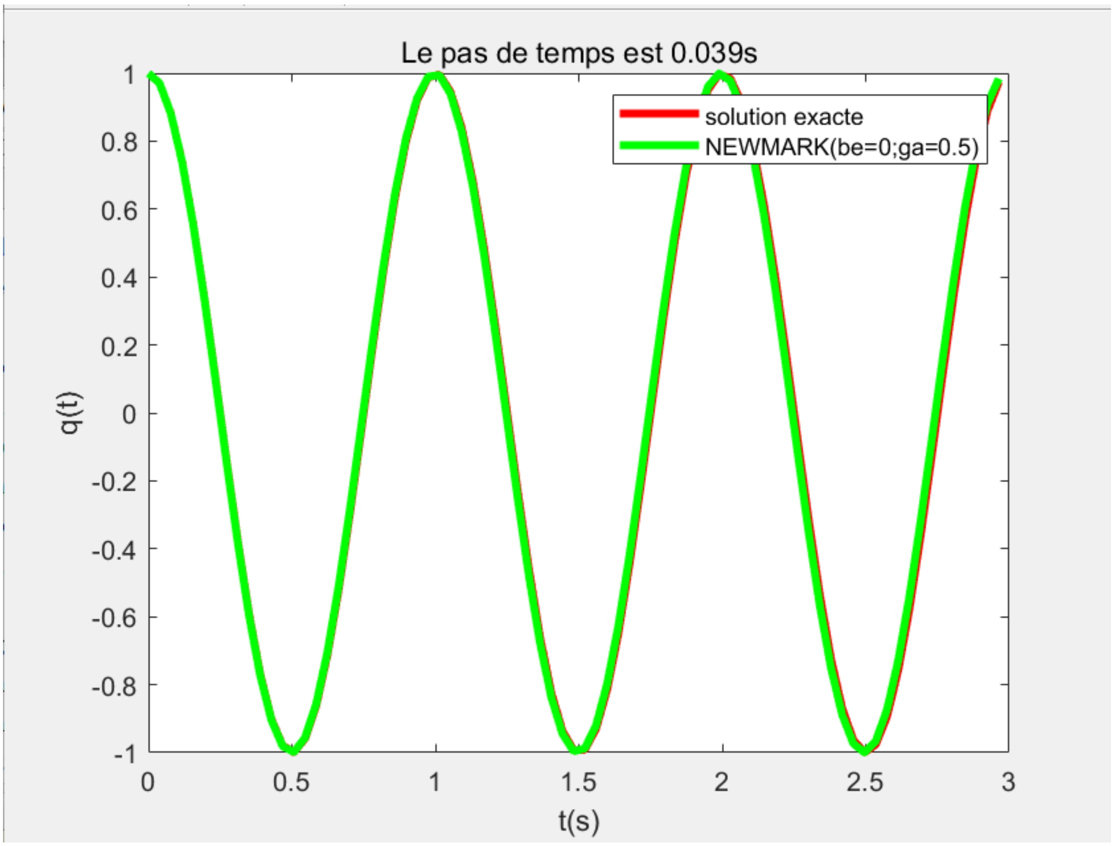
Quand $\Delta t = 0.5s$, le résultat est :



5.2.4 Quand $\Delta t = 0.04s$, le résultat est :



Quand $\Delta t = 0.039s$, le résultat est :



Par dichotomie, on peut choisir $\Delta t = 0.0395s, \alpha = 0.124$

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

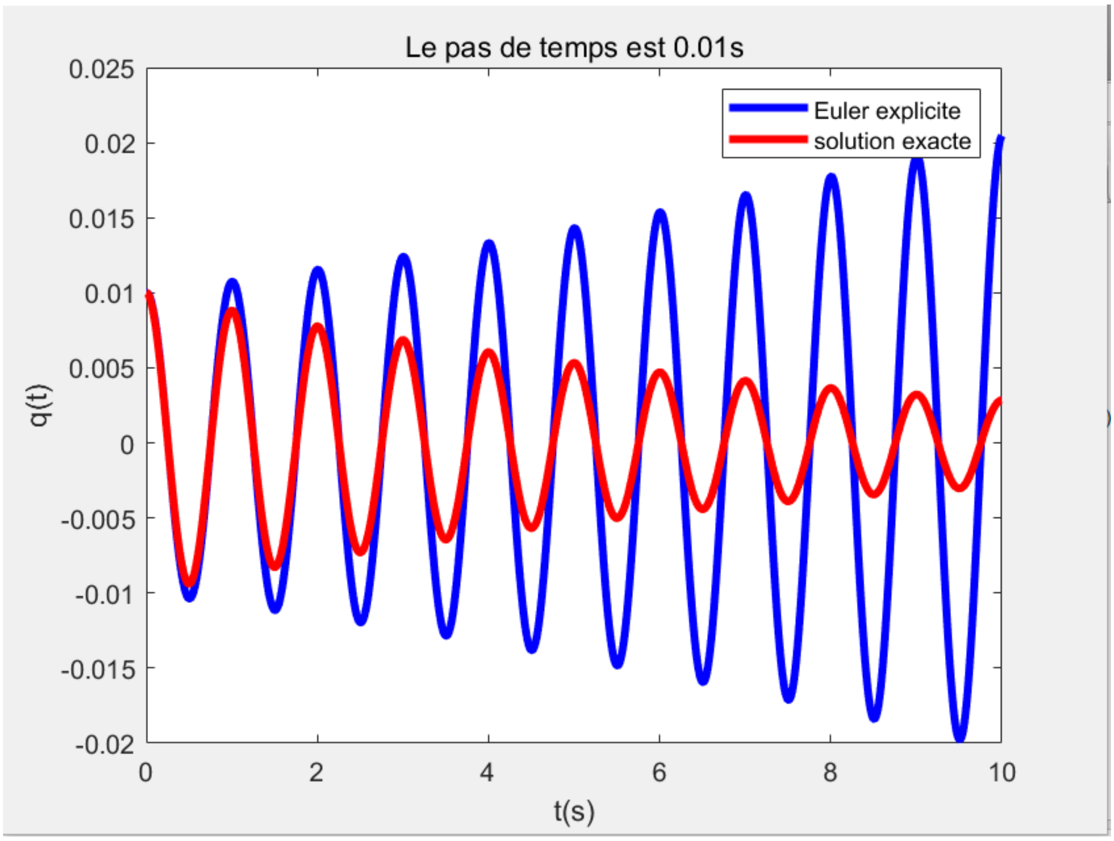
1.1 Code en Matlab :

```
dt1=0.01;T0=10;
x0=0.01;dx0=0;w0=2*pi;w0c=w0*w0;epi=0.02;
t1=(0:dt1:T0)';
np1=size(t1,1);
energie1=zeros(np1,1);
x=[x0;dx0];
x1=zeros(np1,1);
dx1=zeros(np1,1);
dx1(1)=dx0
x1(1)=x0
A=[1,dt1;-w0c*dt1,1-2*epi*w0*dt1]
for inc=2:np1
    x=A*x;
    x1(inc)=x(1);
    dx1(inc)=x(2);
end;


%solution exacte
% b='D2x = -2*0.02*(2*pi)*Dx - ((2*pi)^2)* x';
% x = dsolve(b,'x(0)=0.01,Dx(0)=0');
% x=simplify(x);
t=linspace(0,T0,1000)';
x=(exp(-(pi*t)/25).*(357*cos((7*pi*51^(1/2)*t)/25)
+51^(1/2).*sin((7*pi*51^(1/2)*t)/25)))/35700

plot(t1,x1,'b-','Linewidth',3);
hold on
plot(t,x,'r-','Linewidth',3);
hold on

legend('Euler explicite','solution exacte')
xlabel('t(s)')
ylabel('q(t)')
title('Le pas de temps est 0.01s')
```

$T_0 = 1s$ avec $T_0 = \frac{2\pi}{\omega_0}$, donc $\omega_0 = 2\pi$

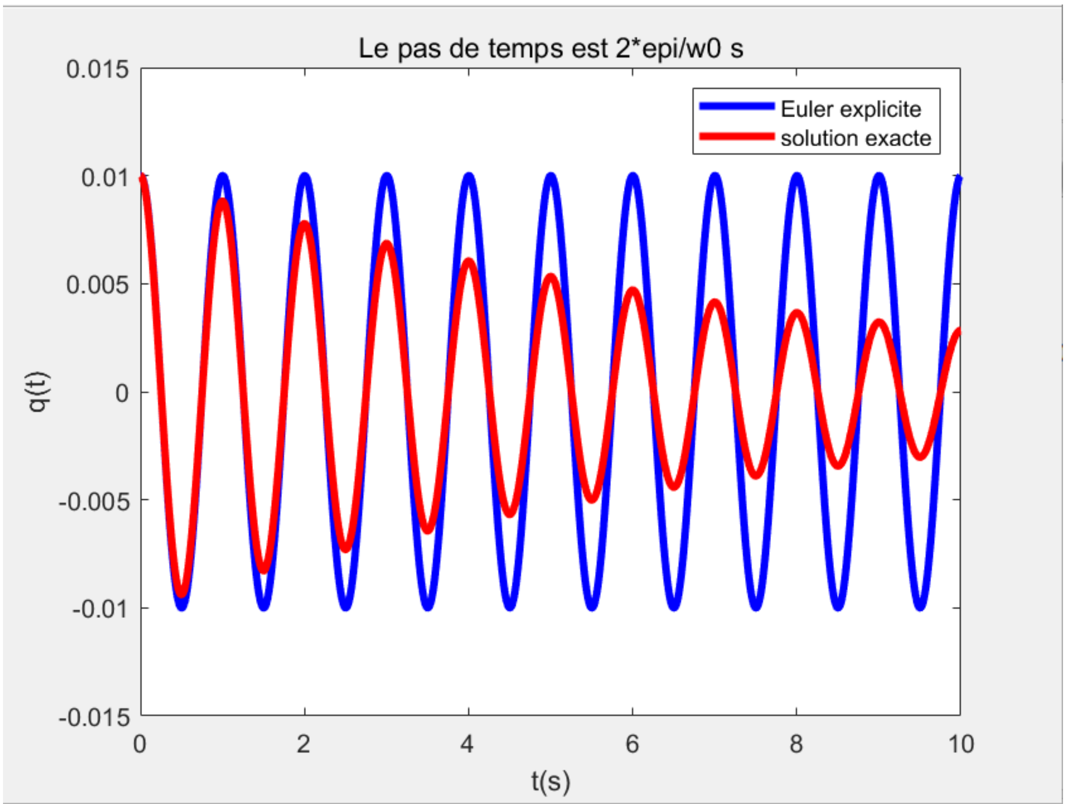
Car $\varepsilon = 0.02$, donc $\frac{2\varepsilon}{\omega_0} \approx 0.00637$

1.1.a) quand $\Delta t > \frac{2\varepsilon}{\omega_0}$, on choisit $\Delta t = 0.01s$ et on obtient :



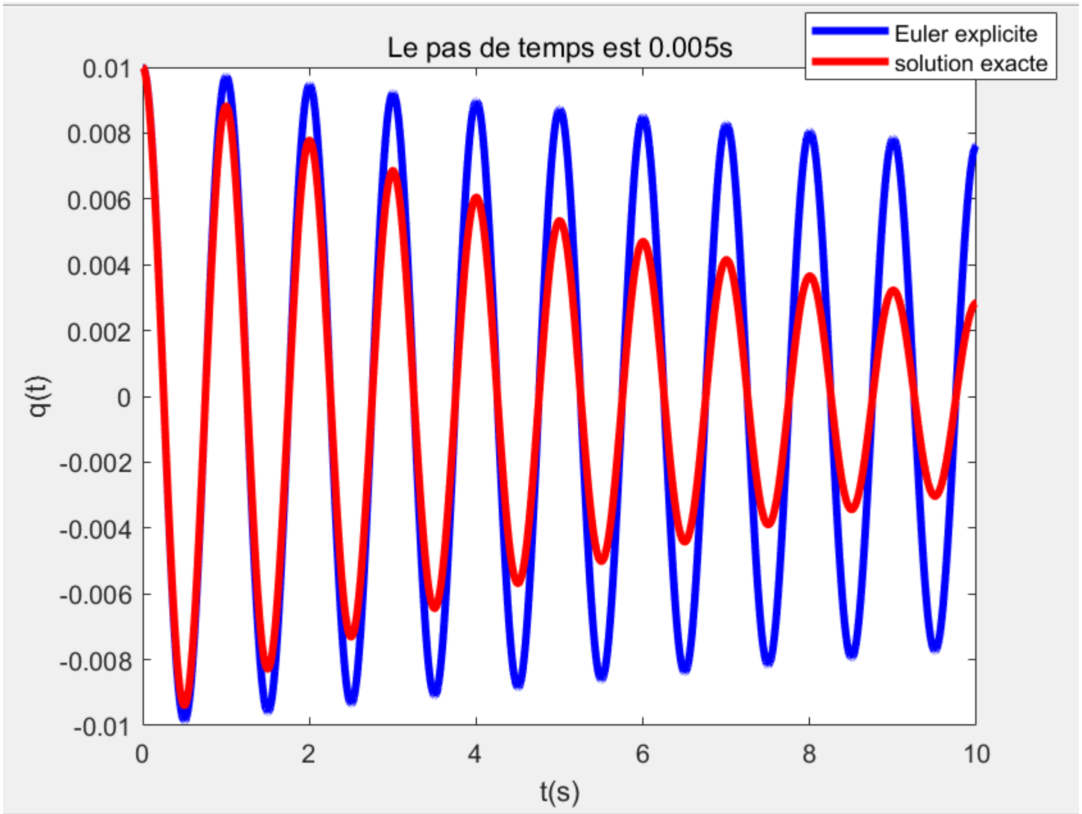
Donc, la solution numérique obtenue avec ce schéma d'intégration est divergente. Au début, Le résultat d'Euler explicite coïncide avec la solution exacte.

1.1.b) quand $\Delta t = \frac{2\varepsilon}{\omega_0}$, on obtient :



Donc, la solution numérique obtenue avec ce schéma d'intégration est périodique.

1.1.c) quand $\Delta t = 0.8 \times \frac{2\varepsilon}{\omega_0} \approx 0.005s$, on obtient :



Donc, ce schéma d'intégration d'EULER explicite introduit un amortissement numérique.

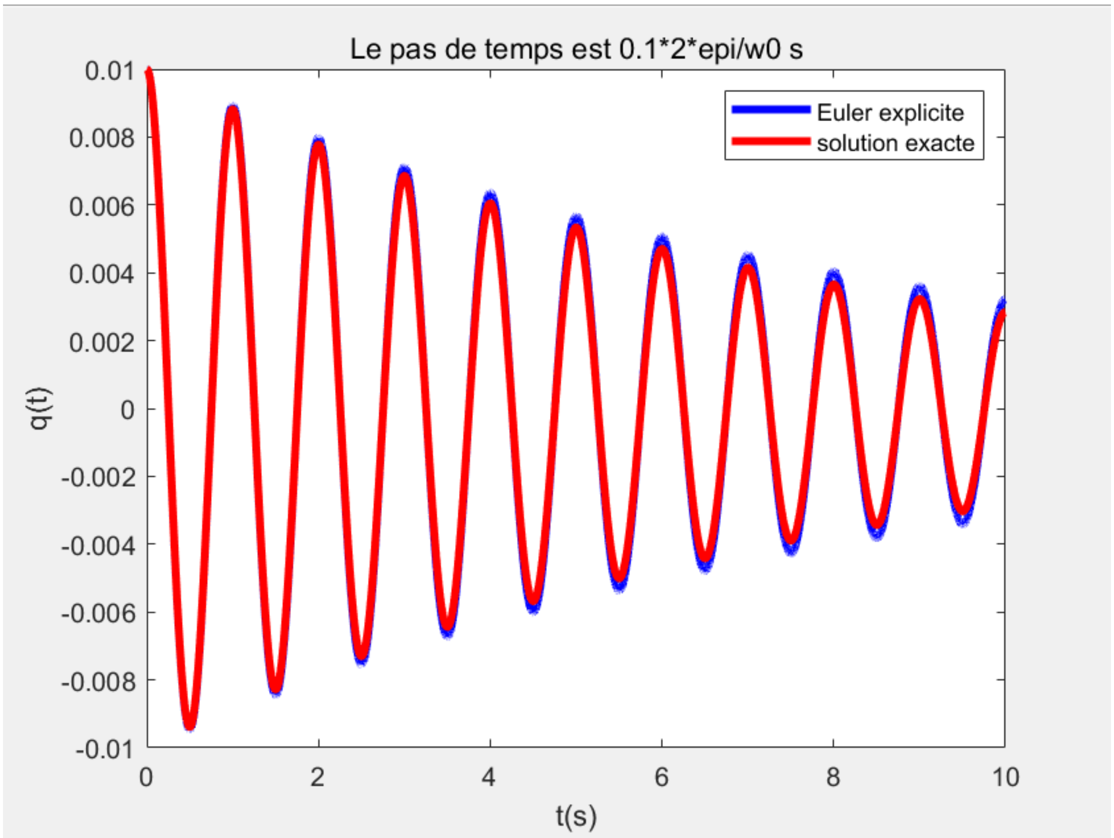
1.1.d) -On calcule : $\det(A - \lambda I) = 0$, avec $A = \begin{bmatrix} 1 & \Delta t \\ -{\omega_0}^2\Delta t^2 & 1 - 2\varepsilon\omega_0\Delta t \end{bmatrix}$

On a : $\lambda = 1 - \varepsilon\omega_0\Delta t \pm i\omega_0\Delta t[1 - \varepsilon^2]^{1/2}$

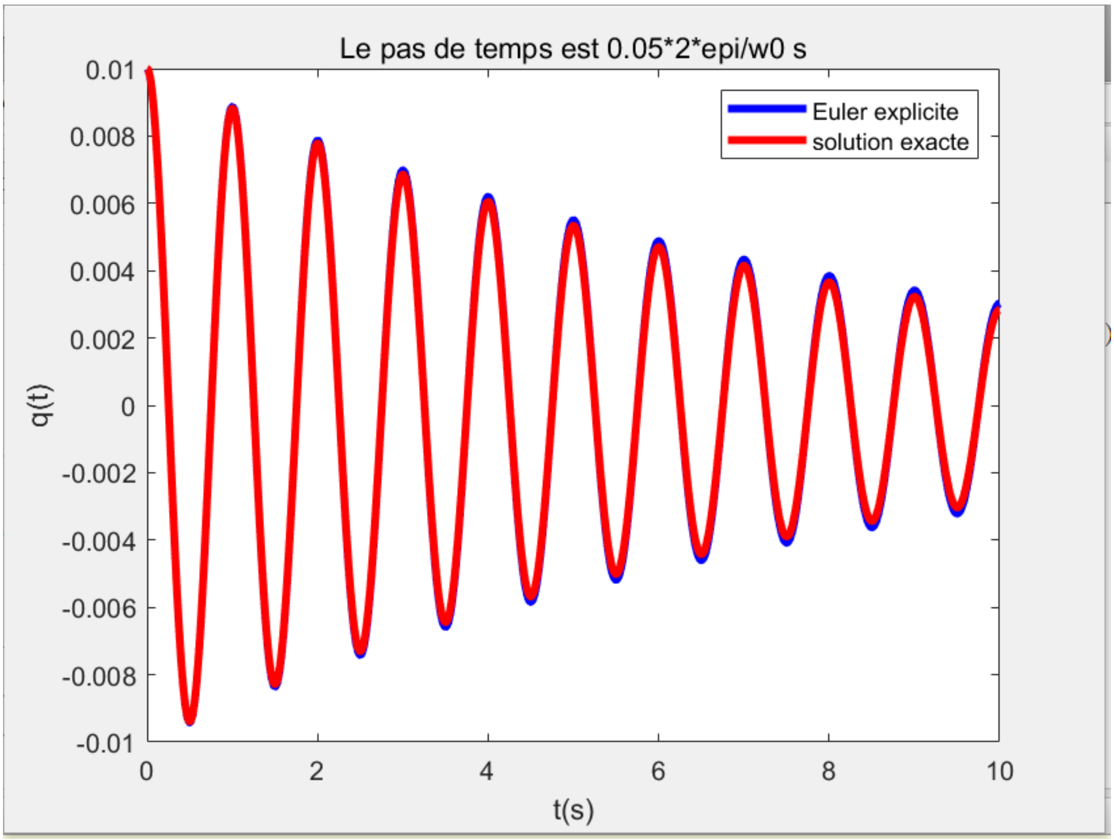
Pour $|\lambda| < 1$, on a : $\Delta t < \frac{2\varepsilon}{\omega_0}$

Donc, le critère permettant d'étudier la précision de la solution est : $\Delta t < \frac{2\varepsilon}{\omega_0}$

- Quand $\Delta t = 0.1 \times \frac{2\varepsilon}{\omega_0} \approx 0.00063s$, on obtient :



Quand $\Delta t = 0.05 \times \frac{2\varepsilon}{\omega_0} \approx 0.00032s$, on obtient :



On peut voir que $\Delta t \approx 0.00032s$, la solution de EULER explicite converge vers la solution exacte. Donc, quand $\frac{\Delta t}{\frac{2\varepsilon}{\omega_0}} < 0.05$, la solution calculée présente-t-elle une précision suffisant.

1.2 Pour calculer le temps critique, On calcule : $\det(A - \lambda I) = 0$,

avec $A^{-1} = \begin{bmatrix} 1 & -\Delta t \\ {\omega_0}^2 \times \Delta t & 1 + 2\varepsilon\omega_0\Delta t \end{bmatrix}$

On utilise matlab pour calculer les valeurs propres de A, et on obtient :

$$\lambda_{1=}0.998388336952360\ +\ 0.031339329879928i$$

$$\lambda_2 = 0.998388336952360\ -\ 0.031339329879928i$$

$$|\lambda_1 < 1 \quad |\lambda_2| < 1$$

En fait, pour calculer le temps critique, on met $|\lambda| < 1$ et calcule Δt. Mais $|\lambda_1 < 1$ $|\lambda_2| < 1$, donc, il n'y a pas de temps de critique.

Code en Matlab :

```
clear all; close all; clc;
dt2=0.005;T0=10;
x0=0.01;dx0=0;w0=2*pi;w0c=w0*w0;epi=0.02;
A=[1,-dt2;w0c*dt2,1+2*epi*w0*dt2]
A=inv(A)
eig(A)
```

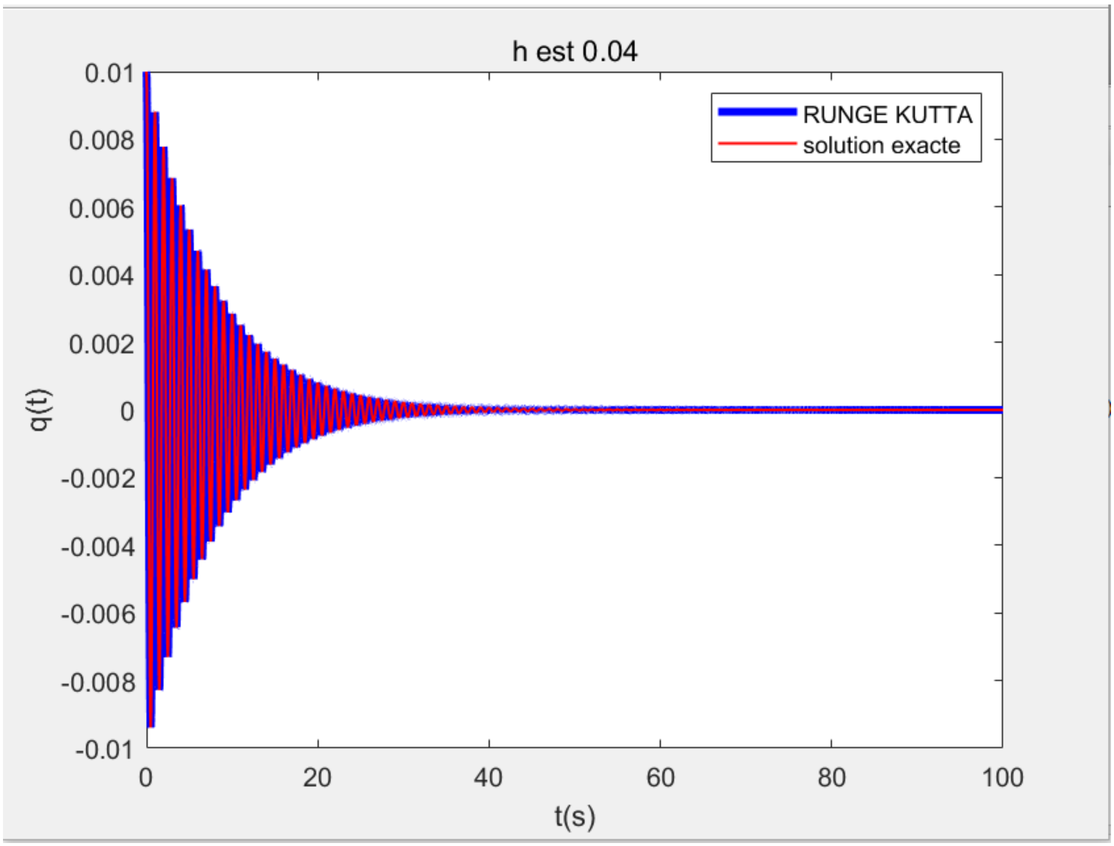
1.3 Résolution avec un schéma de RUNGE KUTTA

1.3.a) Code en Matlab :

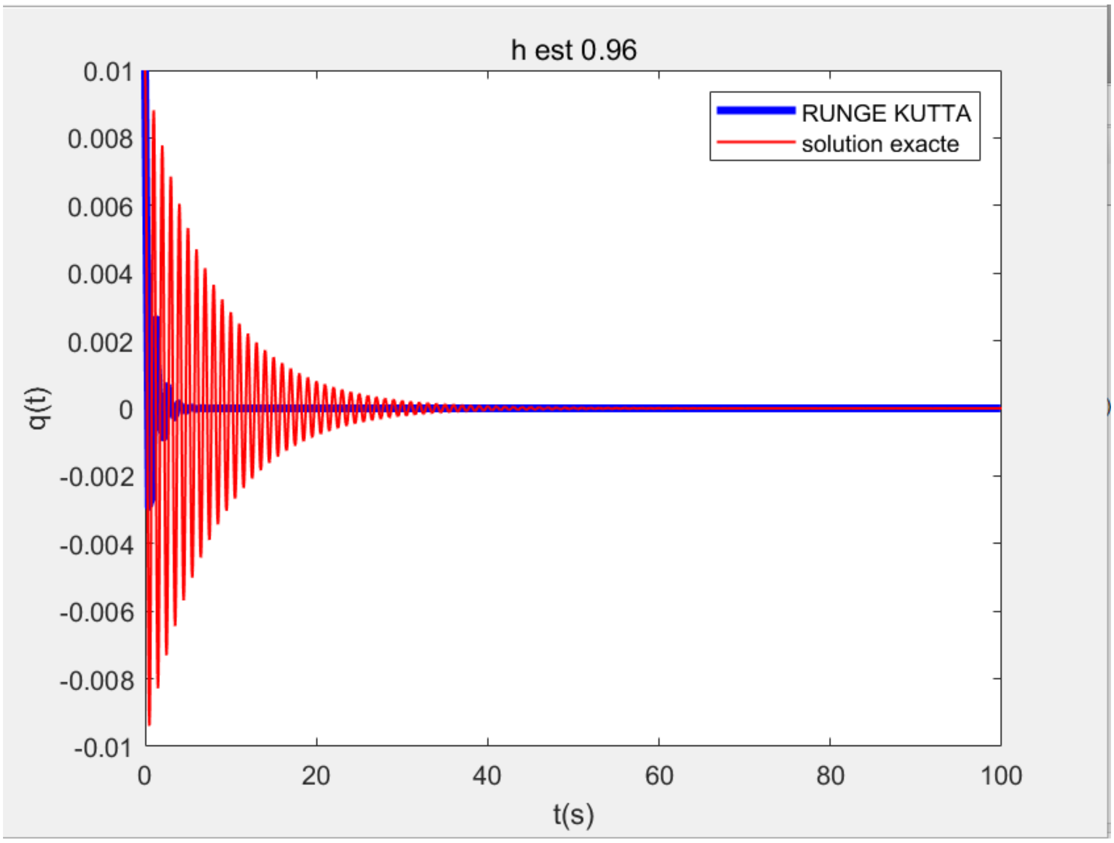
```
clear all; close all; clc;
w0=2*pi;w0c=w0*w0;x0=0.01;dx0=0;
h=0.04;
dt3=h*2*(2^0.5)/w0;T0=100;
t3=(0:dt3:T0)'
np3=size(t3,1)
x3=zeros(np3,1)
dx3=zeros(np3,1)
x3(1)=x0
dx3(1)=dx0
xj=[x0;dx0]
for inc=2:np3
    tc=t3(inc-1);
    xc=xj;
    k1=cal_f(xc,tc);
    xc=xj+k1*dt3/2;
    k2=cal_f(xc,tc+dt3/2);
    xc=xj+k2*dt3/2;
    k3=cal_f(xc,tc + dt3/2);
    xc=xj + k3 * dt3;
    k4=cal_f(xc,tc + dt3);
    dx=(k1 + 2 *k2 + 2 * k3 + k4)/6;
```

```
        xj=xj + dx * dt3;
        x3(inc)= xj(1);
        dx3(inc)=xj(2);
end

%%solution exacte
% b='D2x = -2*0.02*(2*pi)*Dx - ((2*pi)^2)* x';
% x = dsolve(b,'x(0)=0.01,Dx(0)=0');
% x=simplify(x);
t=linspace(0,T0,2000)';
x=(exp(-(pi*t)/25).*(357*cos((7*pi*51^(1/2)*t)/25)                    +
51^(1/2).*sin((7*pi*51^(1/2)*t)/25)))/35700;

plot(t3,x3,'b-','Linewidth',3);
hold on
plot(t,x,'r-','Linewidth',1);
hold on
legend('RUNGE KUTTA','solution exacte')
xlabel('t(s)')
ylabel('q(t)')
title('h est 0.04')

%function cal_f
function dx= cal_f(x,tc)
w0=2*pi;
epi=0.02;
w0c=w0*w0;
dx=zeros(2,1);
dx(1)=x(2);
dx(2)=-2*epi*w0*x(2)-w0c*x(1);
end
```

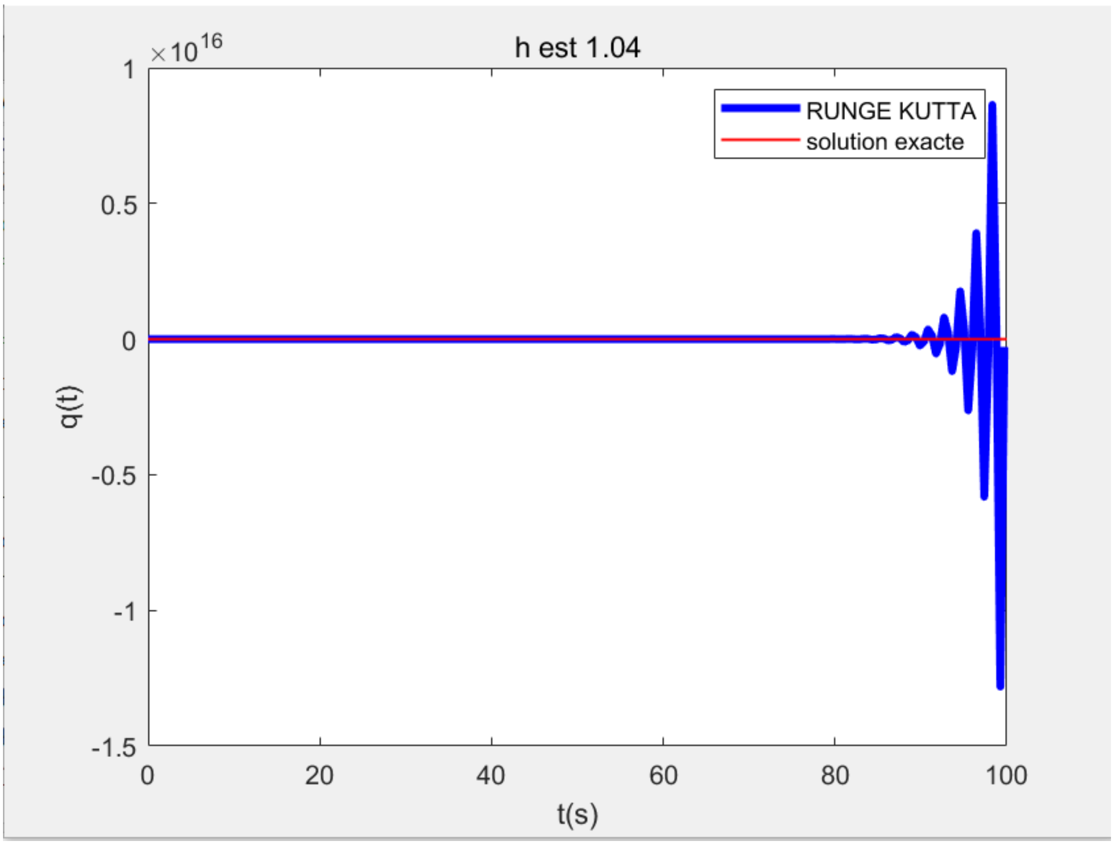
Quand $h = 0.04s$, on obtient :



Quand $h = 0.96s$, on obtient :

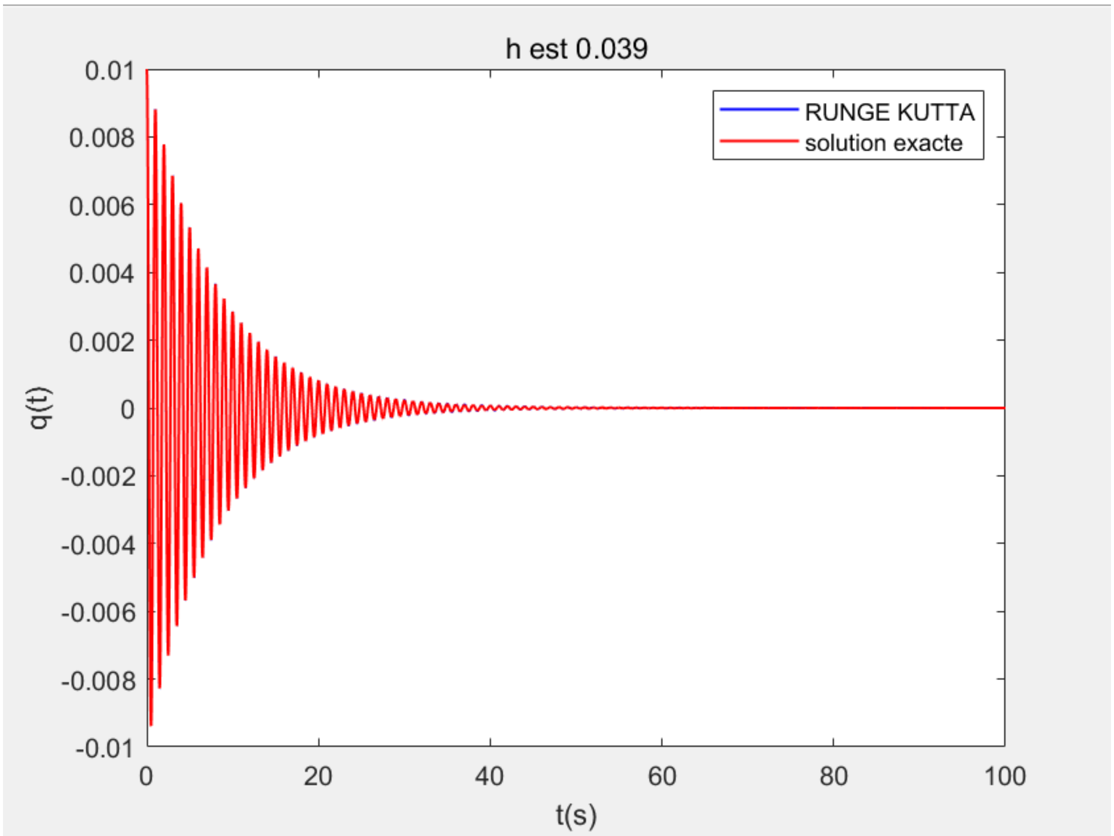


Quand $h = 1.04s$, on obtient :



Quand $h = 0.04s$, la solution de RUNGE KUTTA converge vers la solution exacte. Plus h est petie, plus la solution est proche de la solution exacte.

1.3.b) Quand $h = 0.039s$, on obtient :



$\left|0.039 \times \frac{2\varepsilon}{w0} - 0.04 \times \frac{2\varepsilon}{w0}\right| = |0.01755 - 0.018| = 0.00045 < 0.001$, donc, on peut choisir hc=0.0395.

Etude d'un double pendule avec l'hypothèse des petits mouvements

1. Résolution avec un schéma de NEWMARK explicite

1.1) On choisit :$q = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$, donc $\dot{q} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$ et $\ddot{q} = \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix}$

D'après l'équation (1), on note : $M = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}, N = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}, L = \begin{bmatrix} a \\ \frac{a}{\sqrt{2}} \end{bmatrix}$

On obtient :

$$ma^2M\ddot{q} + mgaNq = F_0 sin(\omega t)L$$

C'est-à-dire :

$$\ddot{q} = \frac{M^{-1}}{ma^2}(F_0 sin(\omega t)L - mgaNq)$$

On note $B = \frac{M^{-1}}{ma^2}F_0 sin(\omega t)L$ et $C = \frac{M^{-1}}{a}gN$, donc :

$$\ddot{q} = B - Cq \qquad (4)$$

D'après l'équation(4) ,(2) et (3), on obtient :

$$q_{n+1} = q_n + \Delta t\dot{q}_n + \Delta t^2(0.5 - \beta)(B - Cq_n) + \Delta t^2\beta(B - Cq_{n+1})$$
$$\dot{q}_{n+1} = \dot{q}_n + \Delta t(1 - \gamma)(B - Cq_n) + \Delta t\gamma(B - Cq_{n+1})$$

Donc :

$$\begin{bmatrix} I + \Delta t^2\beta C & 0 \\ \Delta t\gamma C & I \end{bmatrix}\begin{bmatrix} q_{n+1} \\ \dot{q}_{n+1} \end{bmatrix} = \begin{bmatrix} I - \Delta t^2(0.5 - \beta)C & \Delta t \\ -\Delta t(1 - \gamma)C & I \end{bmatrix}\begin{bmatrix} q_n \\ \dot{q}_n \end{bmatrix} + \begin{bmatrix} 0.5\Delta t^2 B \\ \Delta t B \end{bmatrix}$$

Et puis, on utilise Matlab pour obtenir la matrice d'amplification :

```
clear all; close all; clc;
dt1= sym('dt1','real');
% w0= sym('w0','real');
w0=2*pi;
F0=20
ga=0.5;
be=0;
m=2;
a=0.5;
g=9.81;
I=[1,0;0,1];M=[2,1;1,1];N=[2,0;0,1];L=[a;a/sqrt(2)];
B=inv(M)*(F0*sin(w0*dt1)*L)/(m*a*a);
C=+inv(M)*(g*N)/a;
P1=I+dt1*dt1*be*C;
P2=dt1*ga*C;
P=[P1,[0,0;0,0];P2,I];
Q1=I-dt1*dt1*(0.5-be)*C;
Q2=-dt1*(1-ga)*C;
Q=[Q1,dt1*I;Q2,I];
R1=0.5*dt1*dt1*B;
```

```
R2=dt1*B;
R=[R1;R2];
A=inv(P)*Q
```

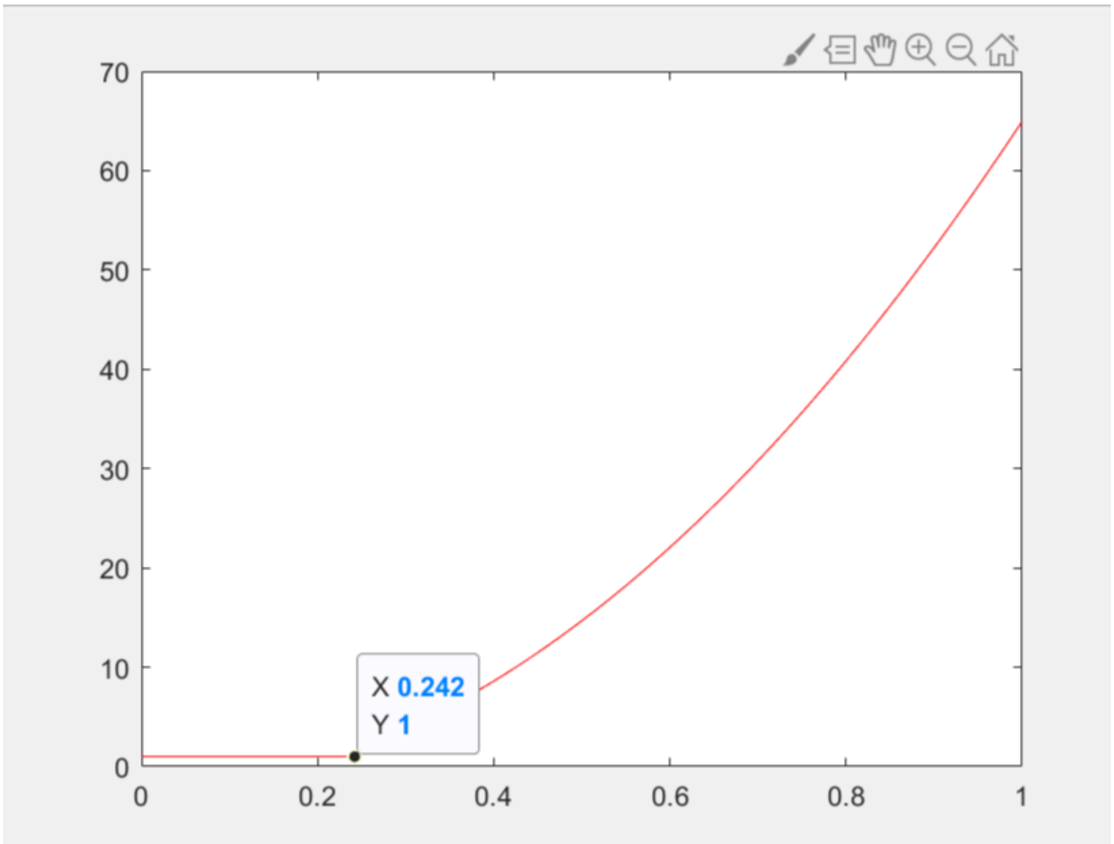
Le résultat est :

A=[1-(981*dt1^2)/50,(981*dt1^2)/100, dt1,0 ; (981*dt1^2)/50, 1-(981*dt1^2)/50, 0,dt1;(981*dt1*((981*dt1^2)/50-1))/50 - (981*dt1)/50 + (962361*dt1^3)/5000, (981*dt1)/100 -(981*dt1*((981*dt1^2)/50-1))/100-(962361*dt1^3)/5000,1-(981*dt1^2)/50, (981*dt1^2)/100 ; (981*dt1)/50 - (981*dt1*((981*dt1^2)/50 - 1))/50 - (962361*dt1^3)/2500, (981*dt1*((981*dt1^2)/50 - 1))/50 - (981*dt1)/50 + (962361*dt1^3)/5000, (981*dt1^2)/50, 1 - (981*dt1^2)/50]

1.2) On utilise Matlab pour calculer le temps de critique :

```
eigmax=[];
for dt1=0:0.001:1;
    eigmax=[eigmax,max(abs(eig(eval(A))))];
end
dt1=0:0.001:1;
plot(dt1,eigmax,'r')
```

On obtient:



Donc , le pas de temps critique est 0.242s.

1.3) $ma^2\begin{bmatrix}2 & 1\\1 & 1\end{bmatrix}\ddot{q}_0 + mga\begin{bmatrix}2 & 0\\0 & 1\end{bmatrix}q_0 = F_0\sin(\omega t)\begin{bmatrix}a\\ \frac{a}{\sqrt{2}}\end{bmatrix}$

1.4) $q_{j+1} = q_j + \Delta t\dot{q}_j + \frac{\Delta t^2}{2}\ddot{q}_j$

$$ma^2\begin{bmatrix}2 & 1\\1 & 1\end{bmatrix}\ddot{q}_j + mga\begin{bmatrix}2 & 0\\0 & 1\end{bmatrix}q_j = F_0\sin(\omega t)\begin{bmatrix}a\\ \frac{a}{\sqrt{2}}\end{bmatrix}$$

$$\dot{q}_{j+1} = \dot{q}_j + \frac{\Delta t}{2}(\ddot{q}_j + \ddot{q}_{j+1})$$

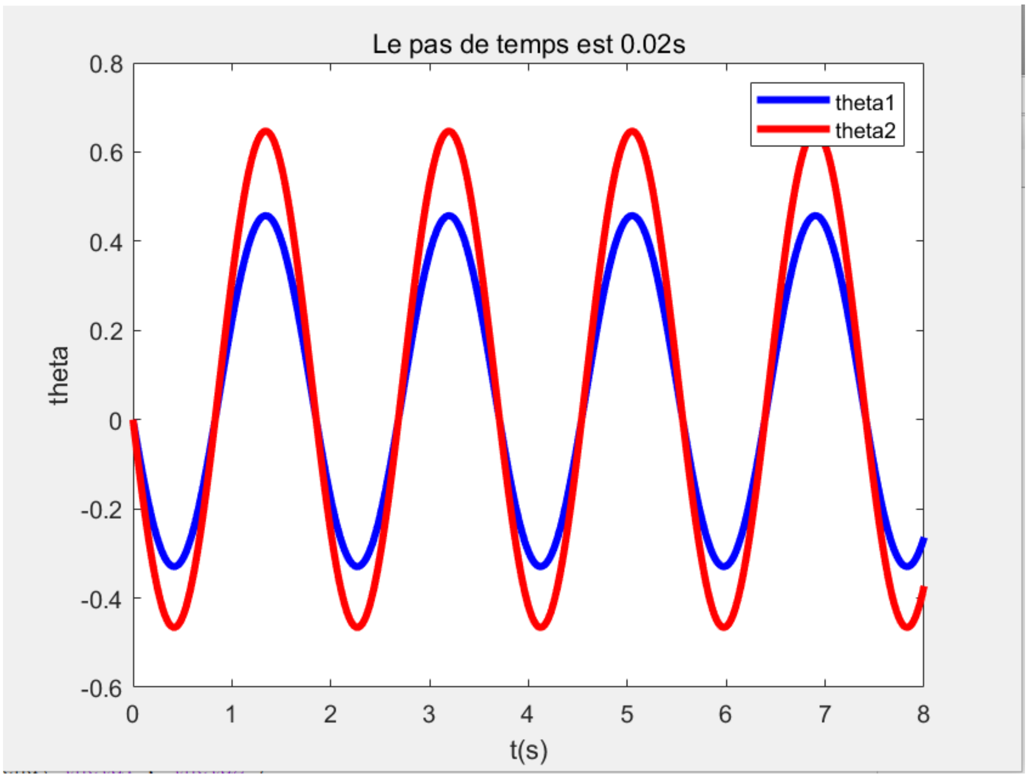
$$ma^2\begin{bmatrix}2 & 1\\1 & 1\end{bmatrix}\ddot{q}_{j+1} + mga\begin{bmatrix}2 & 0\\0 & 1\end{bmatrix}q_{j+1} = F_0\sin(\omega t)\begin{bmatrix}a\\ \frac{a}{\sqrt{2}}\end{bmatrix}$$

1.5) Code en Matlab :

```
clear all; close all; clc;
%w0= sym('w0','real');
dt1=0.02;T0=8;
w0=2*pi;
F0=20
ga=0.5;
be=0;
m=2;
a=0.5;
g=9.81;
I=[1,0;0,1];M=[2,1;1,1];N=[2,0;0,1];L=[a;a/sqrt(2)];
B=inv(M)*(F0*sin(w0*dt1)*L)/(m*a*a);
C=+inv(M)*(g*N)/a;
P1=I+dt1*dt1*be*C;
P2=dt1*ga*C;
P=[P1,[0,0;0,0];P2,I];
Q1=I-dt1*dt1*(0.5-be)*C;
Q2=-dt1*(1-ga)*C;
Q=[Q1,dt1*I;Q2,I];
R1=0.5*dt1*dt1*B;
R2=dt1*B;
R=[R1;R2];
A=inv(P)*Q

theta1_0=0;theta2_0=0;dtheta1_0=-1.31519275;dtheta2_0=-
1.85996342;w0=2*pi;w0c=w0*w0;
ddtheta1_0=0;ddtheta2_0=0;%par les calculs a mains
t1=(0:dt1:T0)';
np1=size(t1,1);
q(1,1)=theta1_0;
q(2,1)=theta2_0;
q(3,1)=dtheta1_0;
q(4,1)=dtheta2_0;
theta1=zeros(np1,1);
```

```
theta2=zeros(np1,1);
dtheta1=zeros(np1,1);
dtheta2=zeros(np1,1);
ddtheta1=zeros(np1,1);
ddtheta2=zeros(np1,1);
theta1(1)=theta1_0;
theta2(1)=theta2_0;
dtheta1(1)=dtheta1_0;
dtheta2(1)=dtheta2_0;
ddtheta1(1)=ddtheta1_0;
ddtheta1(1)=ddtheta2_0;
d(1,1)=ddtheta1_0;
d(2,1)=ddtheta2_0;
for inc=2:np1
    q=A*q+R;
    theta1(inc)=q(1,1);
    theta2(inc)=q(2,1);
    dtheta1(inc)=q(3,1);
    dtheta2(inc)=q(4,1);
    d(:,1)=-C*[q(1,1);q(2,1)]+B;
    ddtheta1(inc)=d(1,1);
    ddtheta2(inc)=d(2,1);
end;
%
plot(t1,theta1,'b-','Linewidth',3);
hold on;
plot(t1,theta2,'r-','Linewidth',3);
hold on;

legend('theta1','theta2')
xlabel('t(s)')
ylabel('theta')
title('Le pas de temps est 0.02s')
```

On obtient :



1.6) Code en Matlab : theta1([1,2,3,26],1)

theta2([1,2,3,26],1)

dtheta1([1,2,3,26],1)

dtheta2([1,2,3,26],1)

ddtheta1([1,2,3,26],1)

ddtheta2([1,2,3,26],1)

Quand $\Delta t = 0s,\ q = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \dot{q} = \begin{bmatrix} -1.3152 \\ -1.8600 \end{bmatrix}, \ddot{q} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$

Quand $\Delta t = 0.02s,\ q = \begin{bmatrix} -0.0262 \\ -0.0370 \end{bmatrix}, \dot{q} = \begin{bmatrix} -1.2975 \\ -1.8349 \end{bmatrix}, \ddot{q} = \begin{bmatrix} 1.0348 \\ 1.4634 \end{bmatrix}$

Quand $\Delta t = 0.04s,\ q = \begin{bmatrix} -0.0519 \\ -0.0734 \end{bmatrix}, \dot{q} = \begin{bmatrix} -1.2738 \\ -1.8014 \end{bmatrix}, \ddot{q} = \begin{bmatrix} 1.3307 \\ 1.8819 \end{bmatrix}$

Quand $\Delta t = 0.5s,\ q = \begin{bmatrix} -0.3133 \\ -0.4430 \end{bmatrix}, \dot{q} = \begin{bmatrix} 0.3785 \\ 0.5352 \end{bmatrix}, \ddot{q} = \begin{bmatrix} 4.3345 \\ 6.1299 \end{bmatrix}$

2. Résolution avec un schéma de NEWMARK implicite

2.1) On choisit :$q = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$, donc $\dot{q} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$ et $\ddot{q} = \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix}$

D'après l'équation (1), on note : $M = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}, N = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}, L = \begin{bmatrix} a \\ \frac{a}{\sqrt{2}} \end{bmatrix}$

On obtient :

$$ma^2M\ddot{q} + mgaNq = F_0 sin(\omega t)L$$

C'est-à-dire :

$$\ddot{q} = \frac{M^{-1}}{ma^2}(F_0 sin(\omega t)L - mgaNq)$$

On note $B = \frac{M^{-1}}{ma^2} F_0 sin(\omega t) L$ et $C = \frac{M^{-1}}{a} gN$, donc :

$$\ddot{q} = B - Cq \qquad (4)$$

D'après l'équation(4) ,(2) et (3), on obtient :

$$q_{n+1} = q_n + \Delta t \dot{q}_n + \Delta t^2 (0.5 - \beta)(B - Cq_n) + \Delta t^2 \beta (B - Cq_{n+1})$$

$$\dot{q}_{n+1} = \dot{q}_n + \Delta t (1 - \gamma)(B - Cq_n) + \Delta t \gamma (B - Cq_{n+1})$$

Donc :

$$\begin{bmatrix} I + \Delta t^2 \beta C & 0 \\ \Delta t \gamma C & I \end{bmatrix} \begin{bmatrix} q_{n+1} \\ \dot{q}_{n+1} \end{bmatrix} = \begin{bmatrix} I - \Delta t^2 (0.5 - \beta) C & \Delta t \\ -\Delta t (1 - \gamma) C & I \end{bmatrix} \begin{bmatrix} q_n \\ \dot{q}_n \end{bmatrix} + \begin{bmatrix} 0.5 \Delta t^2 B \\ \Delta t B \end{bmatrix}$$

Et puis, on utilise Matlab pour obtenir la matrice d'amplification :

```
clear all; close all; clc;
% w0= sym('w0','real');
dt1=0.02;T0=8;
w0=2*pi;
F0=20
ga=0.5;
be=0.24;
m=2;
a=0.5;
g=9.81;
I=[1,0;0,1];M=[2,1;1,1];N=[2,0;0,1];L=[a;a/sqrt(2)];
B=inv(M)*(F0*sin(w0*dt1)*L)/(m*a*a);
C=+inv(M)*(g*N)/a;
P1=I+dt1*dt1*be*C;
P2=dt1*ga*C;
P=[P1,[0,0;0,0];P2,I];
Q1=I-dt1*dt1*(0.5-be)*C;
Q2=-dt1*(1-ga)*C;
Q=[Q1,dt1*I;Q2,I];
R1=0.5*dt1*dt1*B;
R2=dt1*B;
R=[R1;R2];
A=inv(P)*Q
```

Et on obtient:

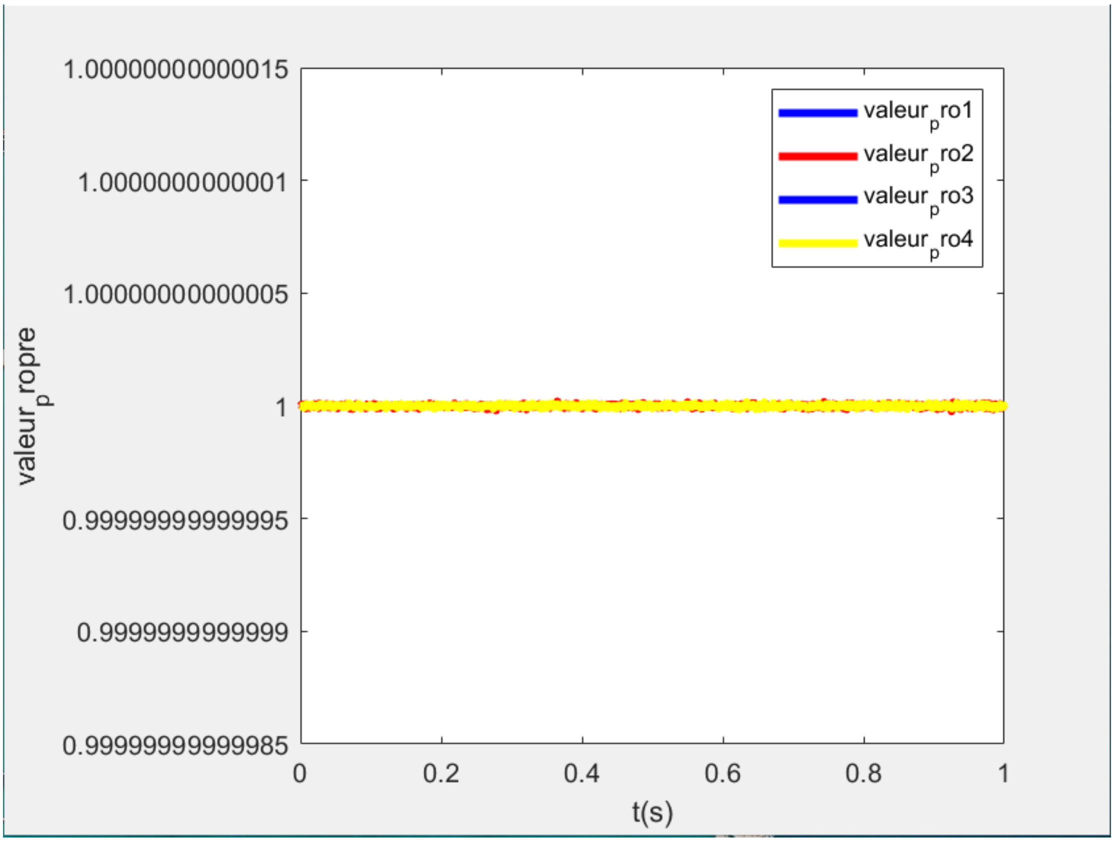
A=[(962361*dt1^4)/(962361*dt1^4+392400*dt1^2+20000)-(200*(981*dt1^2+100)*((981*dt1^2)/100 - 1))/(962361*dt1^4 + 392400*dt1^2 + 20000), (981*dt1^2*(981*dt1^2 + 100))/(962361*dt1^4 + 392400*dt1^2 + 20000) - (98100*dt1^2*((981*dt1^2)/100 - 1))/(962361*dt1^4 + 392400*dt1^2 + 20000), (200*dt1*(981*dt1^2 + 100))/(962361*dt1^4 + 392400*dt1^2 + 20000), (98100*dt1^3)/(962361*dt1^4 + 392400*dt1^2 + 20000) ; (1962*dt1^2*(981*dt1^2 + 100))/(962361*dt1^4 + 392400*dt1^2 + 20000) - (196200*dt1^2*((981*dt1^2)/100 - 1))/(962361*dt1^4 + 392400*dt1^2 + 20000), (962361*dt1^4)/(962361*dt1^4 + 392400*dt1^2 + 20000) - (200*(981*dt1^2 + 100)*((981*dt1^2)/100 - 1))/(962361*dt1^4 + 392400*dt1^2 + 20000), (196200*dt1^3)/(962361*dt1^4 + 392400*dt1^2 + 20000), (200*dt1*(981*dt1^2 + 100))/(962361*dt1^4 + 392400*dt1^2 + 20000) ; (1924722*dt1^3)/(962361*dt1^4 + 392400*dt1^2 + 20000) - (981*dt1)/50 + (1962*(981*dt1^3 + 200*dt1)*((981*dt1^2)/100 - 1))/(962361*dt1^4 + 392400*dt1^2 + 20000), (981*dt1)/100 - (196200*dt1*((981*dt1^2)/100 - 1))/(962361*dt1^4 + 392400*dt1^2 + 20000) - (962361*dt1^2*(981*dt1^3 + 200*dt1))/(100*(962361*dt1^4 + 392400*dt1^2 + 20000)), 1 - (1962*dt1*(981*dt1^3 + 200*dt1))/(962361*dt1^4 + 392400*dt1^2 + 20000), (196200*dt1^2)/(962361*dt1^4 + 392400*dt1^2 + 20000) ; (981*dt1)/50 - (392400*dt1*((981*dt1^2)/100 - 1))/(962361*dt1^4 + 392400*dt1^2 + 20000) - (962361*dt1^2*(981*dt1^3 + 200*dt1))/(50*(962361*dt1^4 + 392400*dt1^2 + 20000)), (1924722*dt1^3)/(962361*dt1^4 + 392400*dt1^2 + 20000) - (981*dt1)/50 + (1962*(981*dt1^3 + 200*dt1)*((981*dt1^2)/100 - 1))/(962361*dt1^4 + 392400*dt1^2 + 20000), (392400*dt1^2)/(962361*dt1^4 + 392400*dt1^2 + 20000), 1 - (1962*dt1*(981*dt1^3 + 200*dt1))/(962361*dt1^4 + 392400*dt1^2 + 20000)]

2.2) On utilise Matlab pour représenter sur un graphique l'allure de l' évolution des valeurs propres de cette matrice en fonction du pas de temps Δt :

```
clear all; close all; clc;
w0= sym('w0','real');
% dt1=0.02;T0=8;
dt1=sym('dt1','real')
% w0=2*pi;
F0=20
ga=0.5;
be=0.25;
m=2;
a=0.5;
g=9.81;
I=[1,0;0,1];M=[2,1;1,1];N=[2,0;0,1];L=[a;a/sqrt(2)];
t1=(0:0.001:1)';
np1=size(t1,1);
valeur_pro1=zeros(np1,1);
valeur_pro2=zeros(np1,1);
valeur_pro3=zeros(np1,1);
valeur_pro4=zeros(np1,1);
K=zeros(4,1);
dt1=0:0.001:1;
for i=1:length(dt1)
    B=inv(M)*(F0*sin(w0*dt1(i))*L)/(m*a*a);
    C=+inv(M)*(g*N)/a;
    P1=I+dt1(i)*dt1(i)*be*C;
    P2=dt1(i)*ga*C;
    P=[P1,[0,0;0,0];P2,I];
    Q1=I-dt1(i)*dt1(i)*(0.5-be)*C;
    Q2=-dt1(i)*(1-ga)*C;
    Q=[Q1,dt1(i)*I;Q2,I];
```

```
        R1=0.5*dt1(i)*dt1(i)*B;
        R2=dt1(i)*B;
        R=[R1;R2];
        A=inv(P)*Q;
        K=abs(eig(A));
        valeur_pro1(i,1)=K(1,1);
        valeur_pro2(i,1)=K(2,1);
        valeur_pro3(i,1)=K(3,1);
        valeur_pro4(i,1)=K(4,1);
    end
    plot(dt1,valeur_pro1,'b-','Linewidth',3);
    hold on;
    plot(dt1,valeur_pro2,'r-','Linewidth',3);
    hold on;
    plot(dt1,valeur_pro3,'b-','Linewidth',3);
    hold on;
    plot(dt1,valeur_pro4,'y-','Linewidth',3);
    hold on;
    legend('valeur_pro1','valeur_pro2','valeur_pro3','valeur_pro4')
    xlabel('t(s)')
    ylabel('valeur_propre')
```

On obtient:



Il n’y a pas le pas de temps critique, car $|\lambda_{1,2,3,4}| \approx 1$.

2.3) $ma^2\begin{bmatrix}2 & 1\\1 & 1\end{bmatrix}\ddot{q}_0 + mga\begin{bmatrix}2 & 0\\0 & 1\end{bmatrix}q_0 = F_0\sin(\omega t)\begin{bmatrix}a\\\frac{a}{\sqrt{2}}\end{bmatrix}$

2.4) $q_{j+1} = q_j + \frac{\Delta t}{2}\dot{q}_j + \frac{\Delta t}{2}\dot{q}_{j+1}$

$$ma^2\begin{bmatrix}2 & 1\\1 & 1\end{bmatrix}\ddot{q}_j + mga\begin{bmatrix}2 & 0\\0 & 1\end{bmatrix}q_j = F_0\sin(\omega t)\begin{bmatrix}a\\\frac{a}{\sqrt{2}}\end{bmatrix}$$

$$\dot{q}_{j+1} = \dot{q}_j + \frac{\Delta t}{2}(\ddot{q}_j + \ddot{q}_{j+1})$$

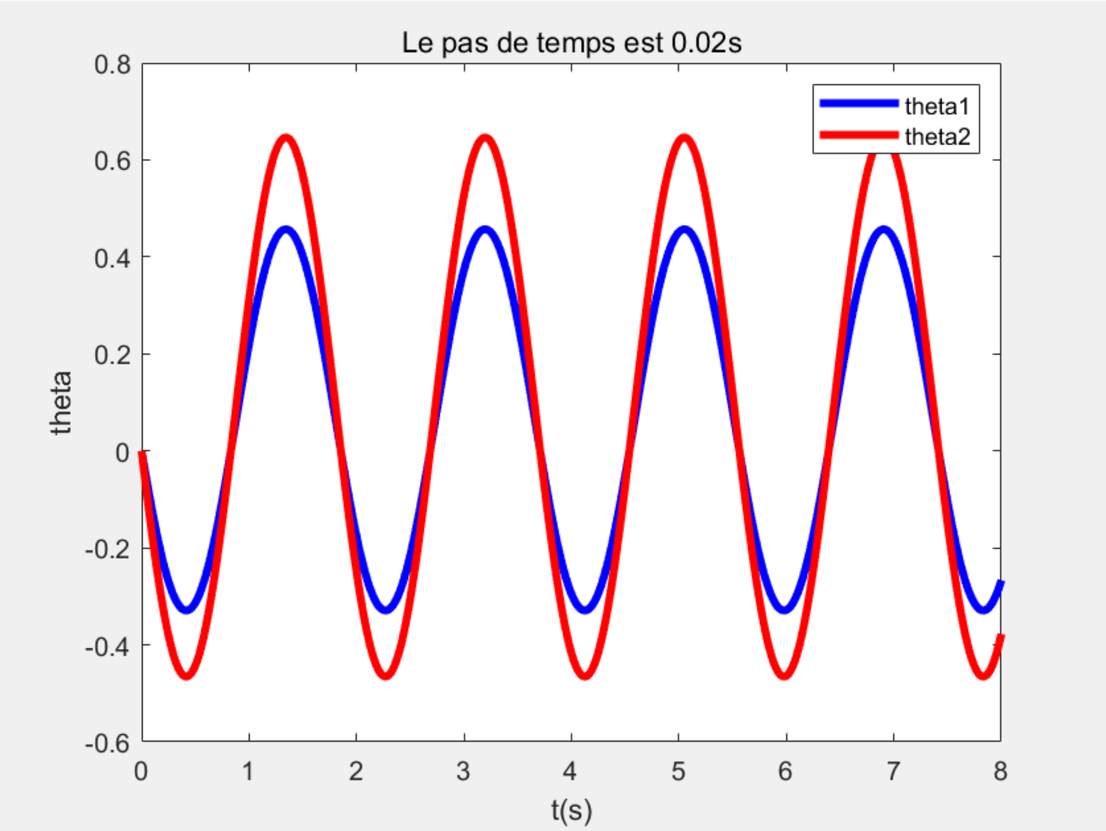
$$ma^2\begin{bmatrix}2 & 1\\1 & 1\end{bmatrix}\ddot{q}_{j+1} + mga\begin{bmatrix}2 & 0\\0 & 1\end{bmatrix}q_{j+1} = F_0\sin(\omega t)\begin{bmatrix}a\\\frac{a}{\sqrt{2}}\end{bmatrix}$$

2.5) Code en Matlab :

```
clear all; close all; clc;
% w0= sym('w0','real');
dt1=0.02;T0=8;
% dt1=sym('dt1','real')
w0=2*pi;
F0=20
ga=0.5;
be=0.25;
m=2;
a=0.5;
g=9.81;
I=[1,0;0,1];M=[2,1;1,1];N=[2,0;0,1];L=[a;a/sqrt(2)];
B=inv(M)*(F0*sin(w0*dt1)*L)/(m*a*a);
C=+inv(M)*(g*N)/a;
P1=I+dt1*dt1*be*C;
P2=dt1*ga*C;
P=[P1,[0,0;0,0];P2,I];
Q1=I-dt1*dt1*(0.5-be)*C;
Q2=-dt1*(1-ga)*C;
Q=[Q1,dt1*I;Q2,I];
R1=0.5*dt1*dt1*B;
R2=dt1*B;
R=[R1;R2];
A=inv(P)*Q;

theta1_0=0;theta2_0=0;dtheta1_0=-1.31519275;dtheta2_0=-
1.85996342;w0=2*pi;w0c=w0*w0;
ddtheta1_0=0;ddtheta2_0=0; %C'est le resultat qu'on obtient par calcule
t1=(0:dt1:T0)';
np1=size(t1,1);
q(1,1)=theta1_0;
```

```
q(2,1)=theta2_0;
q(3,1)=dtheta1_0;
q(4,1)=dtheta2_0;
theta1=zeros(np1,1);
theta2=zeros(np1,1);
dtheta1=zeros(np1,1);
dtheta2=zeros(np1,1);
ddtheta1=zeros(np1,1);
ddtheta2=zeros(np1,1);
theta1(1)=theta1_0;
theta2(1)=theta2_0;
dtheta1(1)=dtheta1_0;
dtheta2(1)=dtheta2_0;
ddtheta1(1)=ddtheta1_0;
ddtheta1(1)=ddtheta2_0;
d(1,1)=ddtheta1_0;
d(2,1)=ddtheta2_0;
for inc=2:np1
    q=A*q+R;
    theta1(inc)=q(1,1);
    theta2(inc)=q(2,1);
    dtheta1(inc)=q(3,1);
    dtheta2(inc)=q(4,1);
    d(:,1)=-C*[q(1,1);q(2,1)]+B;
    ddtheta1(inc)=d(1,1);
    ddtheta2(inc)=d(2,1);
end;
plot(t1,theta1,'b-','Linewidth',3);
hold on;
plot(t1,theta2,'r-','Linewidth',3);
hold on;
legend('theta1','theta2')
xlabel('t(s)')
ylabel('theta')
title('Le pas de temps est 0.02s')
```

Et on obtient :



2.6) Code en Matlab : theta1([1,2,3,26],1)

```
theta2([1,2,3,26],1)
dtheta1([1,2,3,26],1)
dtheta2([1,2,3,26],1)
ddtheta1([1,2,3,26],1)
ddtheta2([1,2,3,26],1)
```

Quand $\Delta t = 0s,\ q = \begin{bmatrix}0\\0\end{bmatrix}, \dot{q} = \begin{bmatrix}-1.3152\\-1.8600\end{bmatrix}, \ddot{q} = \begin{bmatrix}0\\0\end{bmatrix}$

Quand $\Delta t = 0.02s,\ q = \begin{bmatrix}-0.0261\\-0.0369\end{bmatrix}, \dot{q} = \begin{bmatrix}-1.2975\\-1.8349\end{bmatrix}, \ddot{q} = \begin{bmatrix}1.0345\\1.4630\end{bmatrix}$

Quand $\Delta t = 0.04s,\ q = \begin{bmatrix}-0.0518\\-0.0733\end{bmatrix}, \dot{q} = \begin{bmatrix}-1.2738\\-1.8015\end{bmatrix}, \ddot{q} = \begin{bmatrix}1.3300\\1.8809\end{bmatrix}$

Quand $\Delta t = 0.5s,\ q = \begin{bmatrix}-0.3131\\-0.4428\end{bmatrix}, \dot{q} = \begin{bmatrix}0.3773\\0.5336\end{bmatrix}, \ddot{q} = \begin{bmatrix}4.3332\\6.1281\end{bmatrix}$

Oscillateur non linéaire à un degré de liberté

1. Résolution avec un schéma de NEWMARK explicite

1.1) $q_{j+1} = q_j + \Delta t \dot{q}_j + \frac{\Delta t^2}{2} \ddot{q}_j$

$\ddot{q}_j = -\omega_0{}^2 q_j (1 + a \times q_j{}^2)$

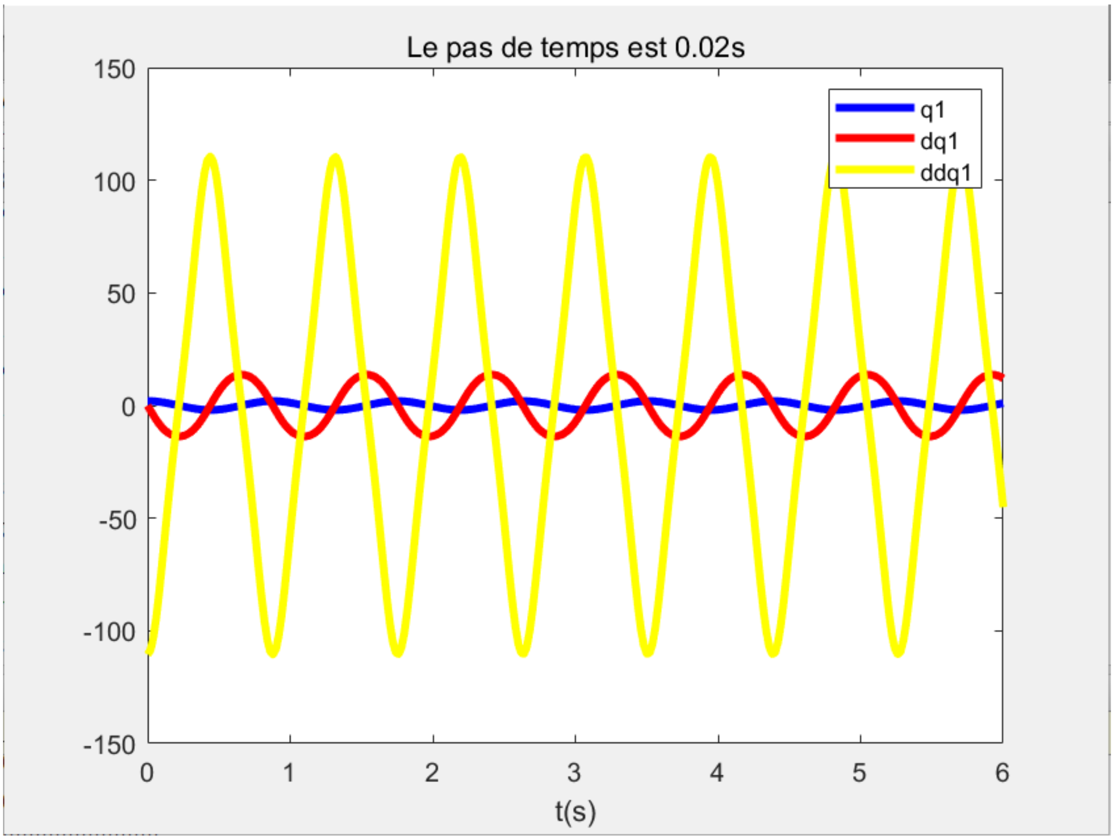
$\dot{q}_{j+1} = \dot{q}_j + \frac{\Delta t}{2}(\ddot{q}_j + \ddot{q}_{j+1})$

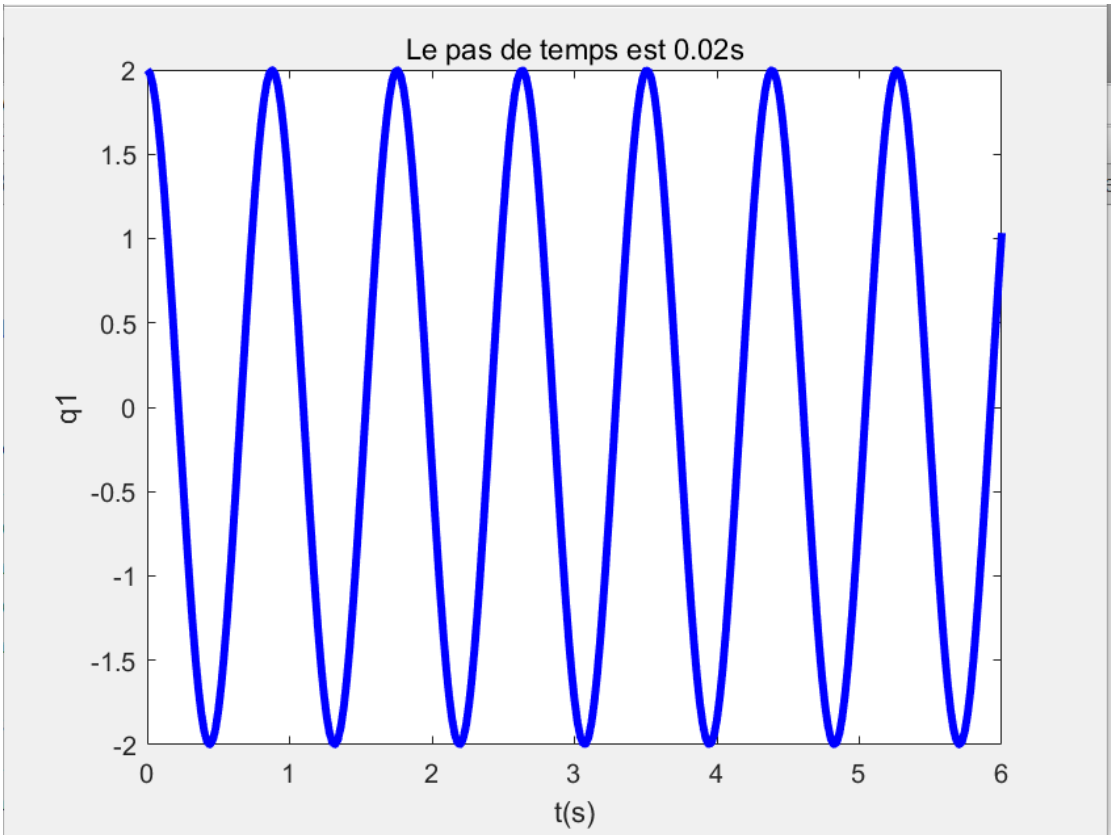
$\ddot{q}_{j+1} = -\omega_0{}^2 q_{j+1} (1 + a \times q_{j+1}{}^2)$

1.2) Code en Matlab :

```
clear all;close all;clc;
dt1=0.02;T0=6;
q0=2;dq0=0;w0=2*pi;w0c=w0*w0;a=0.1;
t1=(0:dt1:T0)'
np1=size(t1,1);
q1=zeros(np1,1);
dq1=zeros(np1,1);
ddq1=zeros(np1,1);
q1(1)=q0;dq1(1)=dq0;
ddq1(1)=-w0c*q0*(1+a*q0*q0);
%NEWMARK explicite
for inc=2:np1
    q1(inc)=q1(inc-1)+dt1*dq1(inc-1)+dt1*dt1*0.5*ddq1(inc-1);
    ddq1(inc)=-w0c*q1(inc)*(1+a*q1(inc)*q1(inc));
    dq1(inc)=dq1(inc-1)+dt1*0.5*ddq1(inc-1)+dt1*0.5*ddq1(inc);
end;
plot(t1,q1,'b-','Linewidth',3);
hold on;
plot(t1,dq1,'r-','Linewidth',3);
hold on;
plot(t1,ddq1,'y-','Linewidth',3);
hold on;
title("Le pas de temps est 0.02s")
legend('q1','dq1','ddq1')
xlabel('t(s)')
% ylabel('q2')
title('Le pas de temps est 0.02s')
```

On obtient :





1.3) Code : q1([1,2,3,301],1);

Quand $\Delta t = 0s,\ q = 2.0000$

Quand $\Delta t = 0.02s,\ q = 1.9779$
Quand $\Delta t = 0.04s,\ q = 1.9123$
Quand $\Delta t = 6s,\ q = 1.0329$

2. Résolution avec un schéma de NEWMARK implicite

2.1) Il faut minimiser le résidu.

2.2) $\ddot{q}_{j+1} = \ddot{q}^*{}_{j+1} + \Delta\ddot{q}_{j+1}, q_{j+1} = q^*{}_{j+1} + \Delta q_{j+1}$

$$f\left(\ddot{q}^*{}_{j+1} + \Delta\ddot{q}_{j+1}, q^*{}_{j+1} + \Delta q_{j+1}\right) = 0$$

$$\ddot{q}^*{}_{j+1} + \Delta\ddot{q}_{j+1} + \omega_0{}^2\left(q^*{}_{j+1} + \Delta q_{j+1}\right)\left(1 + a\left(q^*{}_{j+1} + \Delta q_{j+1}\right)^2\right) = 0$$

$$\Delta q_{j+1} = \beta\Delta t^2\Delta\ddot{q}_{j+1}$$

Donc, l'expression analytique de la correction à appliquer à $\ddot{q}^*{}_{j+1}$ est :

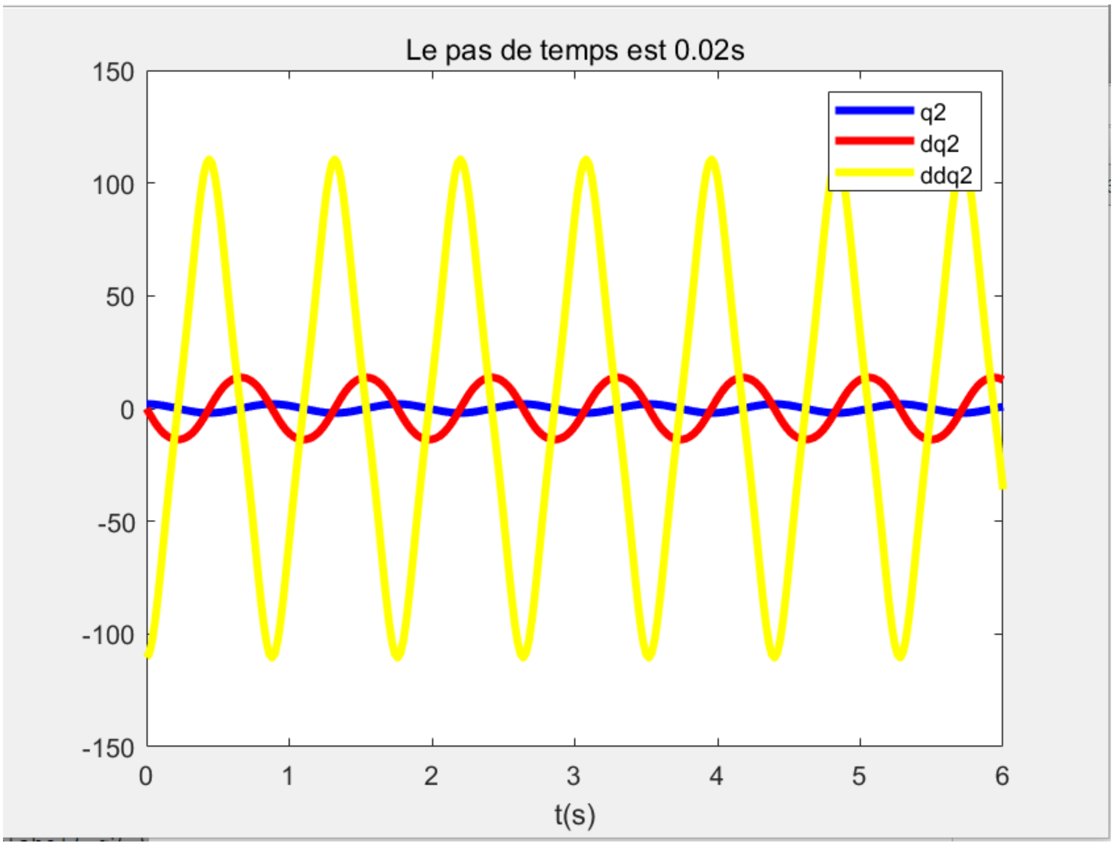
$$\Delta\ddot{q}_{j+1} = -\frac{f(\ddot{q}^*{}_{j+1}, \dot{q}^*{}_{j+1}, q^*{}_{j+1})}{\dfrac{\partial f}{\partial\ddot{q}^*{}_{j+1}} + \dfrac{\partial f}{\partial q^*{}_{j+1}}\beta\Delta t^2}$$

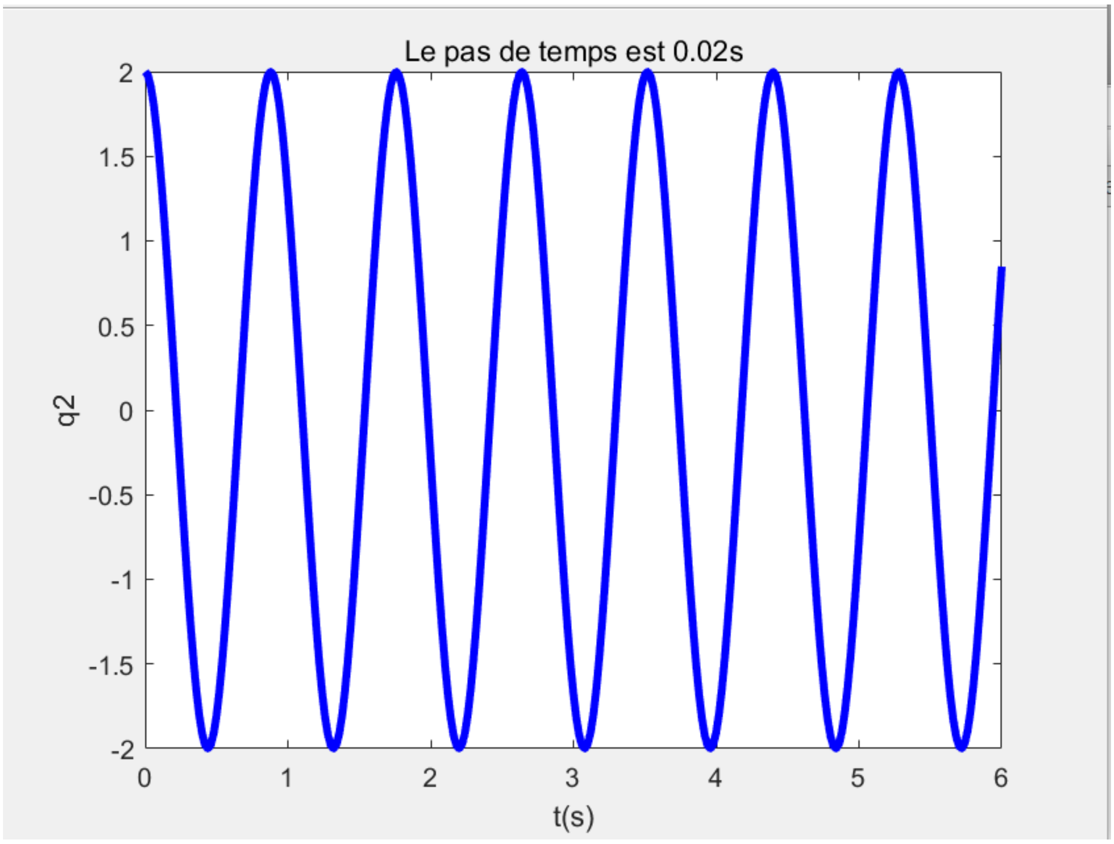
2.3) Code en Matlab :

```
clear all;close all;clc;
dt2=0.02;T0=6;
q0=2;dq0=0;w0=2*pi;w0c=w0*w0;a=0.1;
ga=0.5;be=0.25;
t2=(0:dt2:T0)'
np2=size(t2,1);
q2=zeros(np2,1);
dq2=zeros(np2,1);
ddq2=zeros(np2,1);
q2(1)=q0;dq2(1)=dq0;
ddq2(1)=-w0c*q0*(1+a*q0*q0);
e=0.0001;
%NEWMARK Implicite
for inc=2:np2
    ddq2(inc)=0;
    dq2(inc)=dq2(inc-1)+dt2*(1-ga)*ddq2(inc-1);
    q2(inc)=q2(inc-1)+dt2*dq2(inc-1)+dt2*dt2*(0.5-be)*ddq2(inc-1);
    residu=abs(ddq2(inc)+w0c*q2(inc)*(1+a*q2(inc)*q2(inc)));
    while residu >=e
        f=ddq2(inc)+w0c*q2(inc)*(1+a*q2(inc)*q2(inc));
        deld2q=-f/(1+be*dt2*dt2*(w0c+3*w0c*a*q2(inc)*q2(inc)));
        deldq=ga*dt2*deld2q;
        delq=be*dt2*dt2*deld2q;
        q2(inc)=q2(inc)+delq;
        dq2(inc)=dq2(inc)+deldq;
        ddq2(inc)=ddq2(inc)+deld2q;
```

```
        residu=abs(ddq2(inc)+w0c*q2(inc)*(1+a*q2(inc)*q2(inc)));
    end


end;
plot(t2,q2,'b-','Linewidth',3);
hold on;
% plot(t1,dq1,'r-','Linewidth',3);
% hold on;
% plot(t1,ddq1,'y-','Linewidth',3);
% hold on;
title("Le pas de temps est 0.02s")
% legend('q2','dq2','ddq2')
xlabel('t(s)')
ylabel('q2')
title('Le pas de temps est 0.02s')
```

On obtient :





2.4) Code : q2([1,2,3,301],1);

Quand $\Delta t = 0s$, $q = 2.0000$

Quand $\Delta t = 0.02s$, $q = 1.9781$

Quand $\Delta t = 0.04s$, $q = 1.9131$

Quand $\Delta t = 6s$, $q = 0.8485$

3. Energie mécanique

3.1) $E_c = \frac{1}{2}m\dot{q}^2$, $E_p = \int Fdq$

Donc, $E = \frac{1}{2}m\dot{q}^2 + \frac{1}{2}kq^2 + \frac{1}{4}kaq^4 + cte$

3.2) On choisit que cte=0 et m=1, donc,

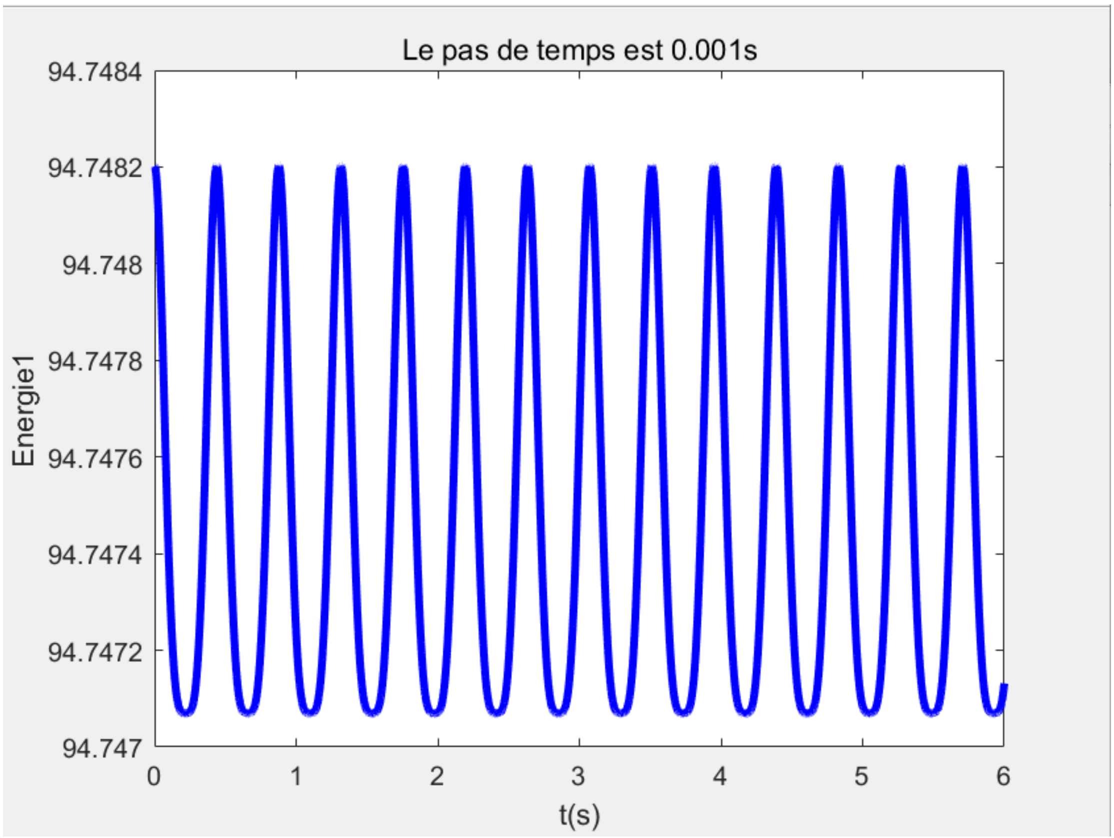
$$E = m\left(\frac{1}{2}\dot{q}^2 + \frac{1}{2}\frac{k}{m}q^2 + \frac{1}{4}\frac{k}{m}aq^4\right) = \frac{1}{2}\dot{q}^2 + \frac{1}{2}{\omega_0}^2 q^2 + \frac{1}{4}{\omega_0}^2 aq^4$$

-Code en Matlab par NEWMARK expicite :

```
clear all;close all;clc;
dt1=0.001;T0=6;
q0=2;dq0=0;w0=2*pi;w0c=w0*w0;a=0.1;
t1=(0:dt1:T0)'
np1=size(t1,1);
q1=zeros(np1,1);
dq1=zeros(np1,1);
ddq1=zeros(np1,1);
Energie1=zeros(np1,1);
```

```
q1(1)=q0;dq1(1)=dq0;
ddq1(1)=-w0c*q0*(1+a*q0*q0);
for inc=2:np1
    q1(inc)=q1(inc-1)+dt1*dq1(inc-1)+dt1*dt1*0.5*ddq1(inc-1);
    ddq1(inc)=-w0c*q1(inc)*(1+a*q1(inc)*q1(inc));
    dq1(inc)=dq1(inc-1)+dt1*0.5*ddq1(inc-1)+dt1*0.5*ddq1(inc);
end;
Energie1=0.5*dq1.*dq1+0.5*w0c*q1.*q1+0.25*w0c*a*(q1.^4);
plot(t1,Energie1,'b-','Linewidth',3);
hold on;
xlabel('t(s)')
ylabel('Energie1')
title('Le pas de temps est 0.001s')
```

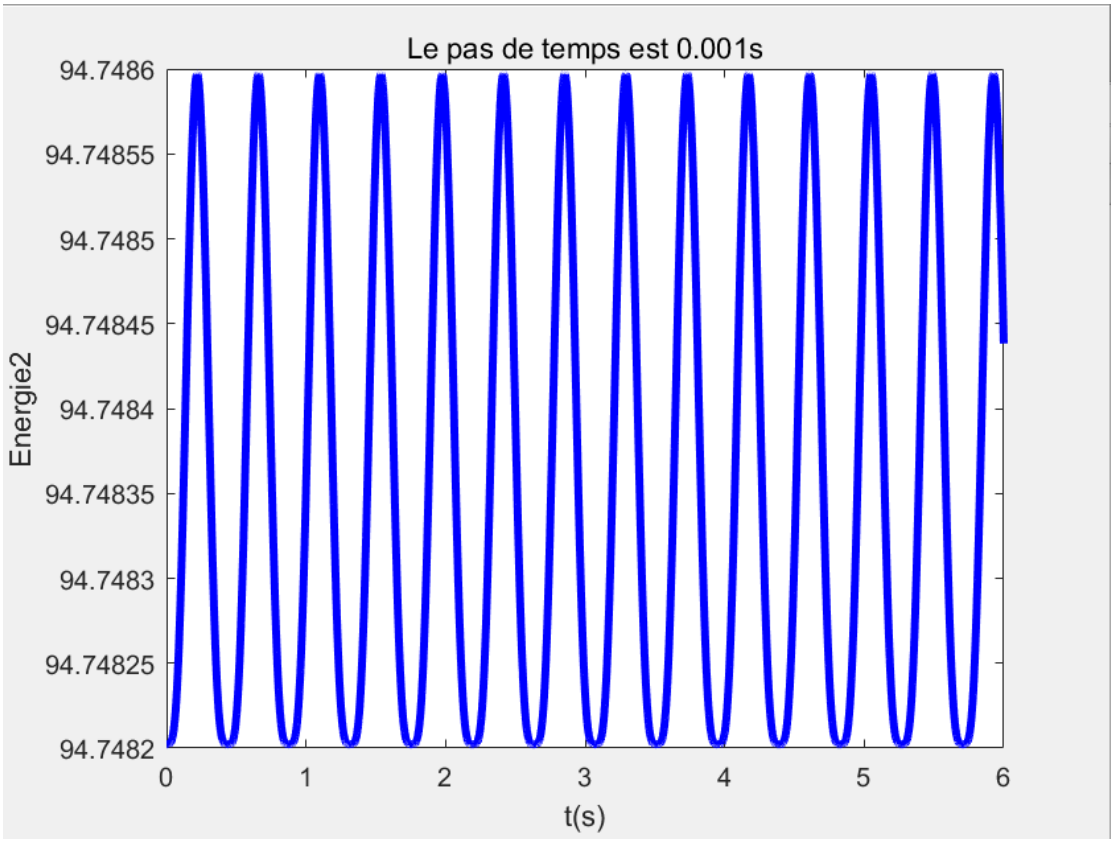
On obtient :



-Code en Matlab par NEWMARK impicite :

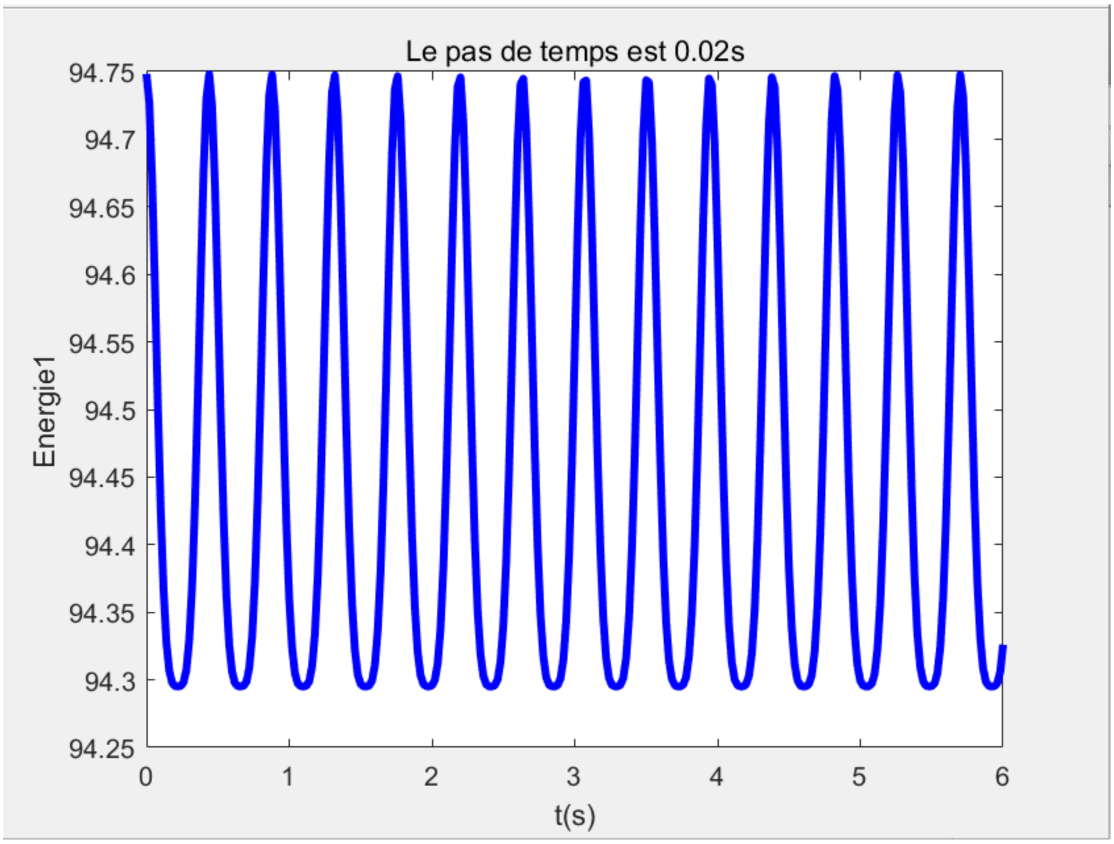
```
clear all;close all;clc;
dt2=0.001;T0=6;
q0=2;dq0=0;w0=2*pi;w0c=w0*w0;a=0.1;
ga=0.5;be=0.25;
t2=(0:dt2:T0)'
np2=size(t2,1);
q2=zeros(np2,1);
```

```
dq2=zeros(np2,1);
ddq2=zeros(np2,1);
Energie2=zeros(np2,1);
q2(1)=q0;dq2(1)=dq0;
ddq2(1)=-w0c*q0*(1+a*q0*q0);
e=0.0001;
for inc=2:np2
    ddq2(inc)=0;
    dq2(inc)=dq2(inc-1)+dt2*(1-ga)*ddq2(inc-1);
    q2(inc)=q2(inc-1)+dt2*dq2(inc-1)+dt2*dt2*(0.5-be)*ddq2(inc-1);
    residu=abs(ddq2(inc)+w0c*q2(inc)*(1+a*q2(inc)*q2(inc)));
    while residu >=e
        f=ddq2(inc)+w0c*q2(inc)*(1+a*q2(inc)*q2(inc));
        deld2q=-f/(1+be*dt2*dt2*(w0c+3*w0c*a*q2(inc)*q2(inc)));
        deldq=ga*dt2*deld2q;
        delq=be*dt2*dt2*deld2q;
        q2(inc)=q2(inc)+delq;
        dq2(inc)=dq2(inc)+deldq;
        ddq2(inc)=ddq2(inc)+deld2q;
        residu=abs(ddq2(inc)+w0c*q2(inc)*(1+a*q2(inc)*q2(inc)));
    end


end;
Energie2=0.5*dq2.*dq2+0.5*w0c*q2.*q2+0.25*w0c*a*(q2.^4);
plot(t2,Energie2,'b-','Linewidth',3);
hold on;
title("Le pas de temps est 0.001s")
xlabel('t(s)')
ylabel('Energie2')
```
On obtient:



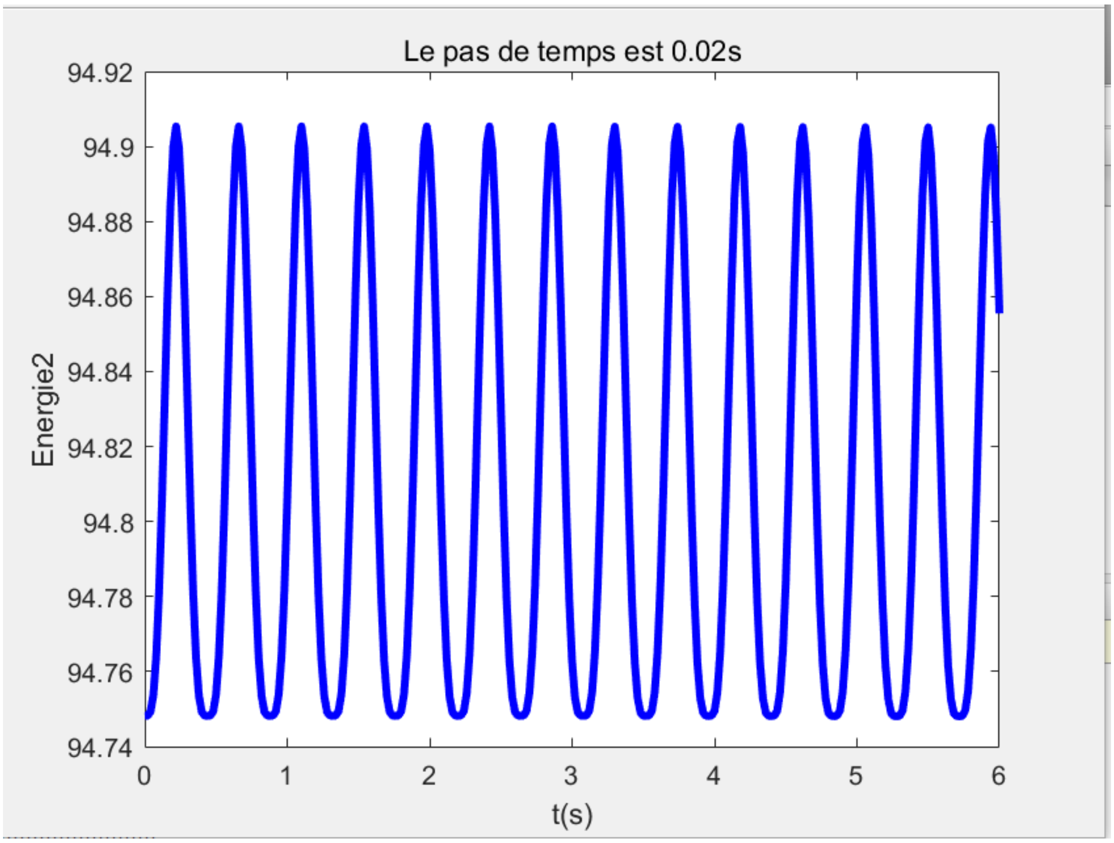
3.3) Quand $\Delta t = 0.02s$, on obtient :

En NEWMARK explicite :



Mean(Energie1)= 94.4561

En NEWMARK implicite :



Mean(Energie2)= 94.8102

On peut voir qu'il n'a y pas d'amortissements. Donc, l' énergie ne diminue pas.