Bonjour monsieur,

Je suis désolée que je n'ai pas encore obtenu le résultat de la qustion2 de <u>*Etude d'un oscillateur non linéaire à un degré de liberté*</u>, mais la date limite pour soumettre mon travail est aujourd'hui. Donc je vous envoie d'abord les réponses aux questions précédentes.

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

1.1 Euler explicite

1.1.a

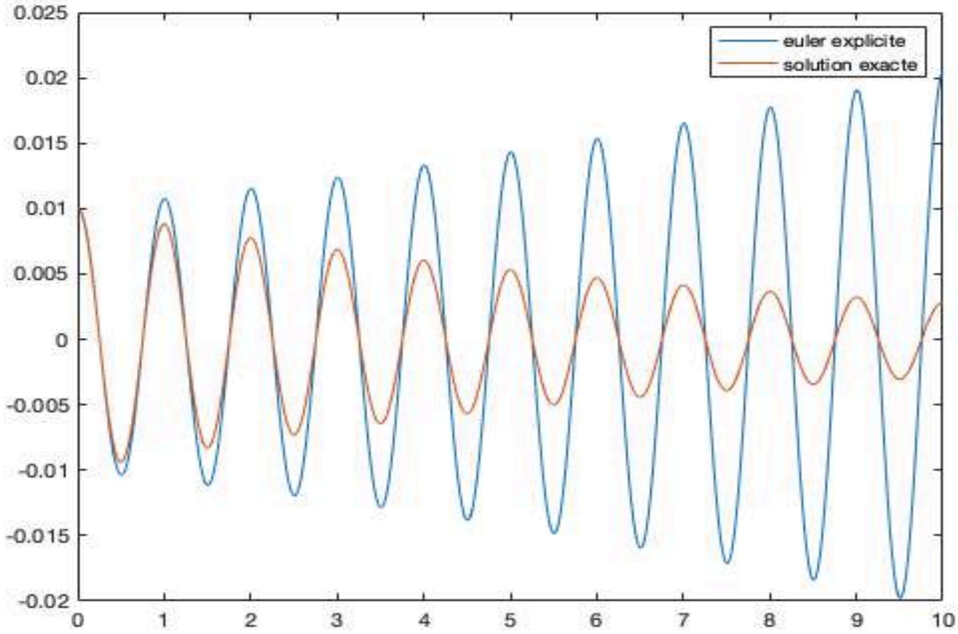
Quand on choisit dt > $2\varepsilon/\omega_0$ (dt=0.01) :

```
clear all;
T0 = 1;
w0 = 2*pi/T0;
eps = 0.02;
Tc = 2*eps/w0;
x0 = 0.01;
Dx0 = 0;
omega = w0*sqrt(1-eps^2);
```

1.1

```
%on choisit dt>Tc
n = 1000;
dt = 10*T0/n;
t = 0:dt:10*T0;
A1 = [1,dt;-w0^2*dt,1-2*eps*w0*dt];
U1(:,1) = [x0;Dx0];
for j = 1:length(t)-1
    U1(:,j+1) = A1*U1(:,j);
end
plot(t,U1(1,:))
hold on;
plot(t,exp(-eps*w0*t).*(x0*cos(omega*t)+(eps*w0*x0+Dx0)/omega.*sin(omega*t)));
legend('euler explicite','solution exacte')
```

On peut obtenir le résultat comme ci-dessous :



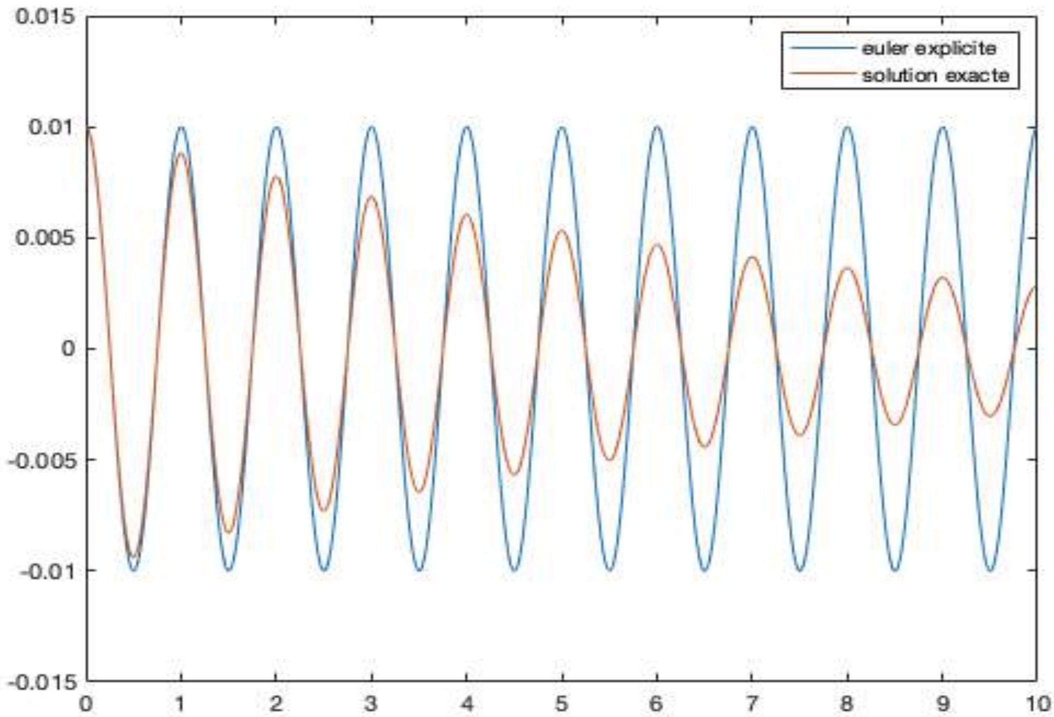
On peut trouver que quand le pas de temps est assez grand (supérieur à $2\varepsilon/\omega_0$), la solution d'Euler explicite est croissante, alors que la solution exacte est amortie.

1.1.b

Quand on choisit dt = $2\varepsilon/\omega_0$:

```
%on choisit dt=Tc
dt = Tc;
t = 0:dt:10*T0;
A1 = [1,dt;-w0^2*dt,1-2*eps*w0*dt];
U1(:,1) = [x0;Dx0];
for j = 1:length(t)-1
    U1(:,j+1) = A1*U1(:,j);
end
plot(t,U1(1,:))
hold on;
plot(t,exp(-eps*w0*t).*(x0*cos(omega*t)+(eps*w0*x0+Dx0)/omega.*sin(omega*t)));
legend('euler explicite','solution exacte')
```

On peut obtenir le résultat comme ci-dessous :

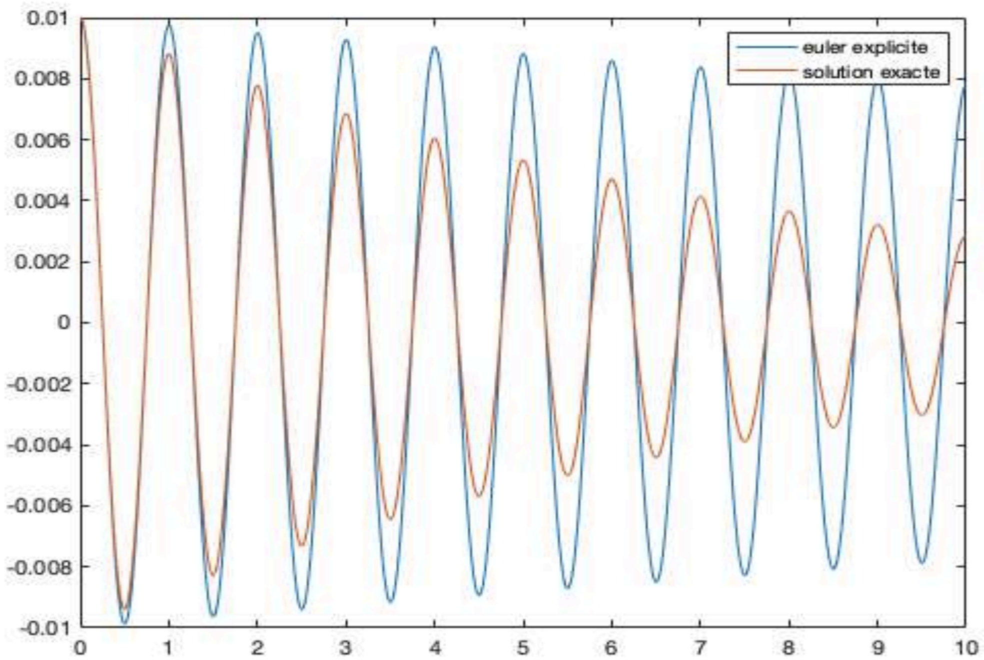


D'après la figure, quand le pas de temps égale $2\varepsilon/\omega_0$, la solution d'Euler explicite est l'oscillation sans amortissement, alors que la solution exacte est amortie.

1.1.c

Quand on choisit dt = 0.8*$2\varepsilon/\omega_0$:

```
%on choisit dt=0.8*Tc
dt = 0.8*Tc;
t = 0:dt:10*T0;
A1 = [1,dt;-w0^2*dt,1-2*eps*w0*dt];
U1(:,1) = [x0;Dx0];
for j = 1:length(t)-1
    U1(:,j+1) = A1*U1(:,j);
end
plot(t,U1(1,:))
hold on;
plot(t,exp(-eps*w0*t).*(x0*cos(omega*t)+(eps*w0*x0+Dx0)/omega.*sin(omega*t)));
legend('euler explicite','solution exacte')
```



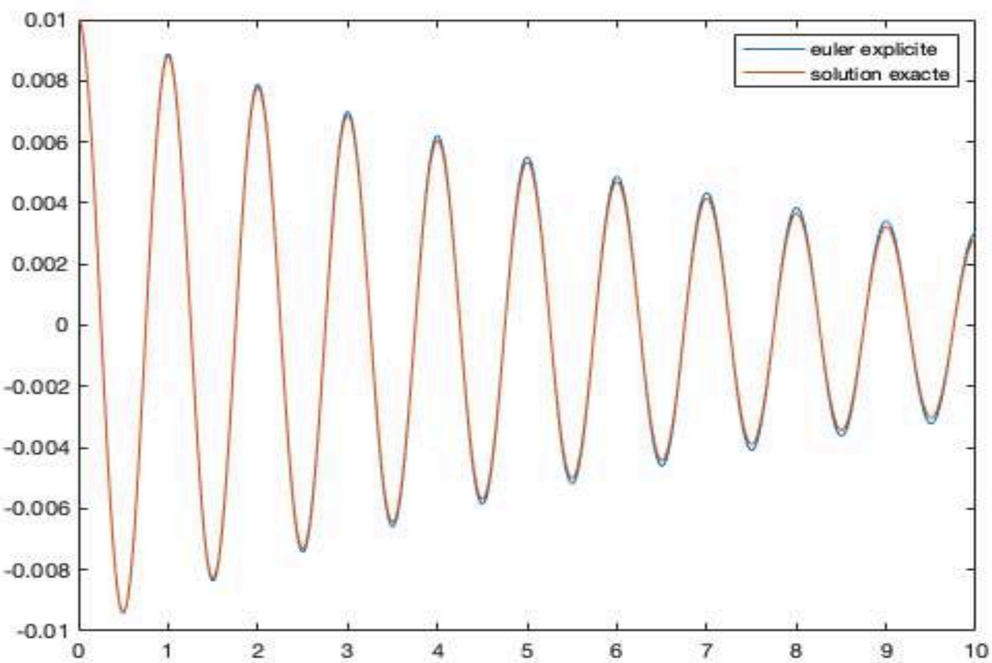
On peut voir que quand le pas de temps est petit à $2\varepsilon/\omega_0$, la solution d'Euler explicite est amortie comme la solution exacte.

1.1.d

- Les critères permettant d'étudier la précision da la solution sont: la cohérence de croissance, la propagation des erreurs numériques, etc.
- Quand on met le rapport dt/($2\varepsilon/\omega_0$) inférieur à 0.05, la solution calculée présente une précision suffisante.

```
%on choisit dt=0.8*Tc
dt = 0.05*Tc;
t = 0:dt:10*T0;
A1 = [1,dt;-w0^2*dt,1-2*eps*w0*dt];
%calculer le module des valeurs propres
vp = eig(A1);
module_vp = abs(vp(1))
U1(:,1) = [x0;Dx0];
for j = 1:length(t)-1
    U1(:,j+1) = A1*U1(:,j);
end
plot(t,U1(1,:))
hold on;
plot(t,exp(-eps*w0*t).*(x0*cos(omega*t)+(eps*w0*x0+Dx0)/omega.*sin(omega*t)));
legend('euler explicite','solution exacte')
```

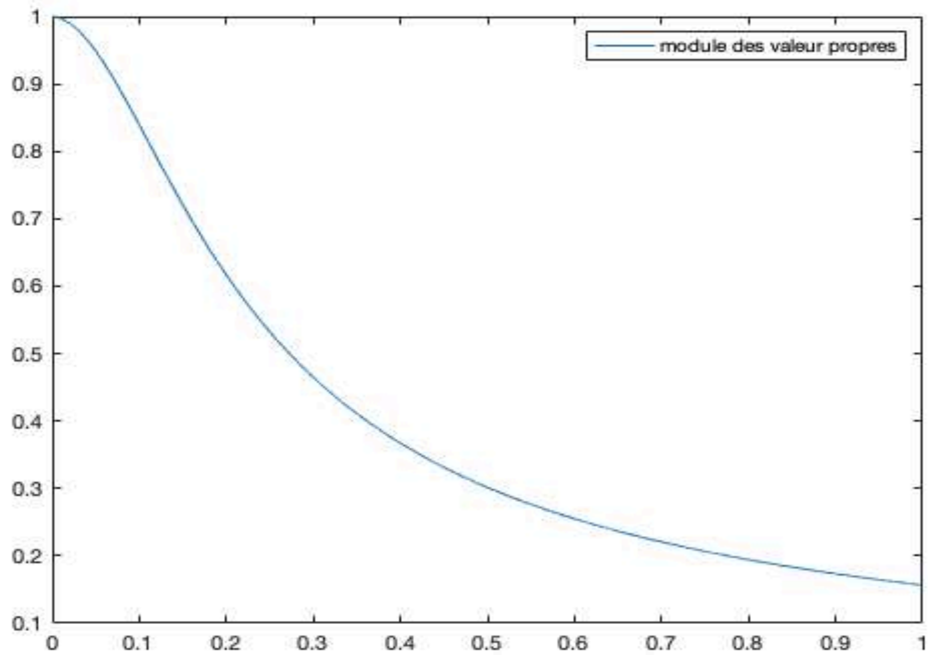
```
module_vp =

    1.0000
```



1.2 Euler implicite

Si on choisit la méthode d'Euler implicite, la solution calculée est toujours amortie. Et on calcule le module des valeurs propres de la matrice d'amplification :

```
clf;
%déterminer la matrice d'amplification A2
B = [1,-dt;w0^2*dt,1+2*eps*w0*dt];
A2 = inv(B);
dt_test = 0:0.0001:1;
%trouver le pas de temps critique
for i = 1:length(dt_test)
    B_test(1,i) = [1,-dt_test(i);w0^2*dt_test(i),1+2*eps*w0*dt_test(i)];
    A_test(1,i) = inv(B_test(1,i));
    vpA_test(:,i) = eig(A_test(1,i));
    module_test(i) = sqrt(real(vpA_test(1,i))^2+imag(vpA_test(1,i))^2);
end
plot(dt_test,module_test);
legend('module des valeur propres');
```



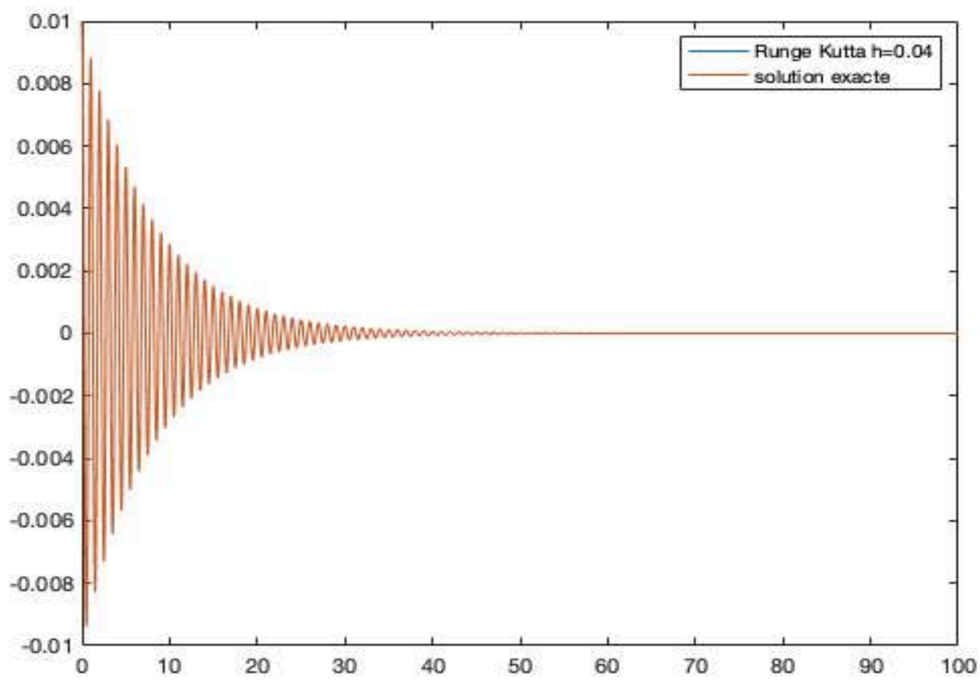
Le module est toujours inférieur à 1, donc il n'y a pas de pas de temps critique.

1.3 Runge-Kutta

1.3.a

Quand h = 0.04 :

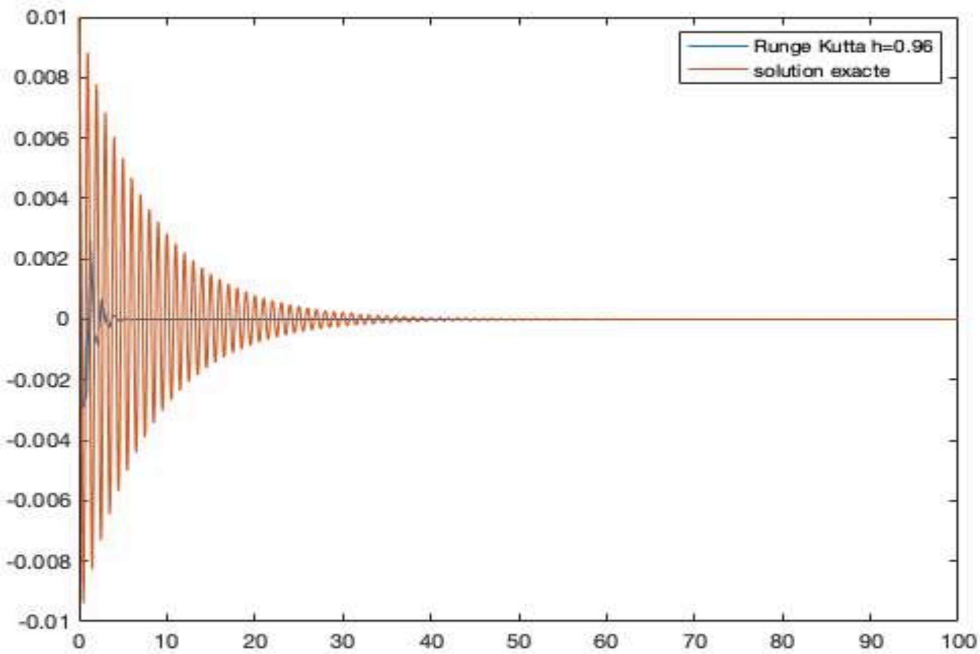
```
clf;
%quand h=0.04
h = 0.04;
dt_rk = h*2*sqrt(2)/w0;
t_rk = 0:dt_rk:100*T0;
C = [0,1;-w0^2,-2*eps*w0];
Urk(:,1) = [x0;Dx0];
for i = 1:length(t_rk)-1
    k1 = C*Urk(:,i);
    k2 = C*(Urk(:,i)+1/2*dt_rk*k1);
    k3 = C*(Urk(:,i)+1/2*dt_rk*k2);
    k4 = C*(Urk(:,i)+dt_rk*k3);
    K = 1/6*(k1+2*k2+2*k3+k4);
    Urk(:,i+1) = Urk(:,i)+K*dt_rk;
end
%comparer les figures
plot(t_rk,Urk(1,:))
hold on;
plot(t,exp(-eps*w0*t).*(x0*cos(omega*t)+(eps*w0*x0+Dx0)/omega.*sin(omega*t)));
legend('Runge Kutta h=0.04','solution exacte');
```



Le résultat est très proche à la solution exacte, stable et précis.

Quand h = 0.96 :

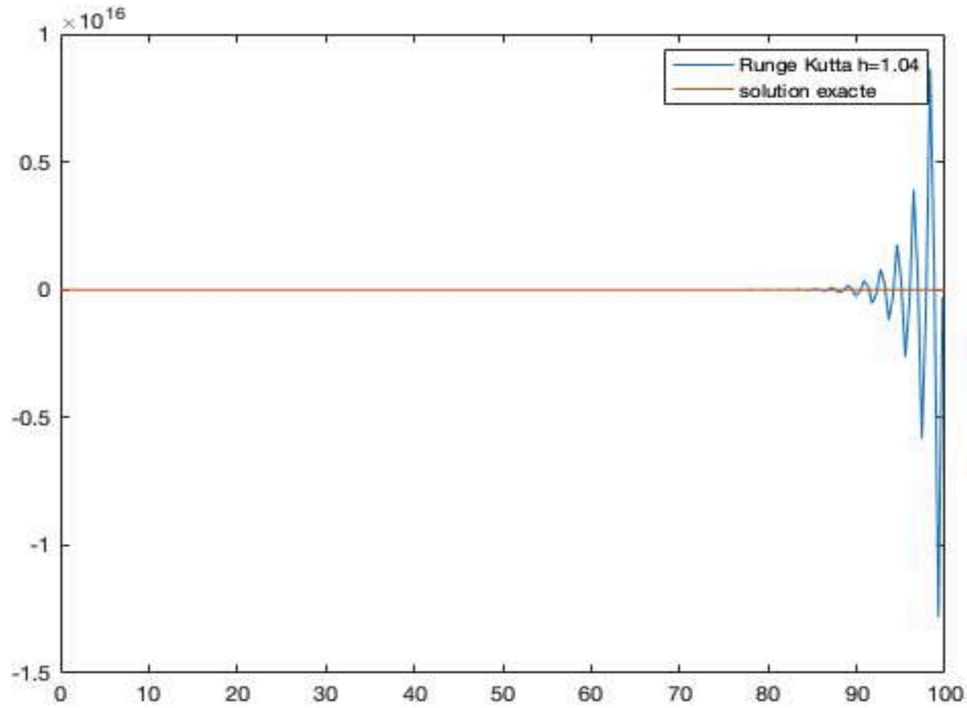
```
clf;
%quand h=0.96
h = 0.96;
dt_rk = h*2*sqrt(2)/w0;
t_rk = 0:dt_rk:100*T0;
C = [0,1;-w0^2,-2*eps*w0];
Urk(:,1) = [x0;Dx0];
for i = 1:length(t_rk)-1
    k1 = C*Urk(:,i);
    k2 = C*(Urk(:,i)+1/2*dt_rk*k1);
    k3 = C*(Urk(:,i)+1/2*dt_rk*k2);
    k4 = C*(Urk(:,i)+dt_rk*k3);
    K = 1/6*(k1+2*k2+2*k3+k4);
    Urk(:,i+1) = Urk(:,i)+K*dt_rk;
end
%comparer les figures
plot(t_rk,Urk(1,:))
hold on;
plot(t,exp(-eps*w0*t).*(x0*cos(omega*t)+(eps*w0*x0+Dx0)/omega.*sin(omega*t)));
legend('Runge Kutta h=0.96','solution exacte');
```



Le résultat est stable mais pas précis de la solution exacte.

Quand h= 1.04 :

```
clf;
%quand h=1.04
h = 1.04;
dt_rk = h*2*sqrt(2)/w0;
t_rk = 0:dt_rk:100*T0;
C = [0,1;-w0^2,-2*eps*w0];
Urk(:,1) = [x0;Dx0];
for i = 1:length(t_rk)-1
    k1 = C*Urk(:,i);
    k2 = C*(Urk(:,i)+1/2*dt_rk*k1);
    k3 = C*(Urk(:,i)+1/2*dt_rk*k2);
    k4 = C*(Urk(:,i)+dt_rk*k3);
    K = 1/6*(k1+2*k2+2*k3+k4);
    Urk(:,i+1) = Urk(:,i)+K*dt_rk;
end
%comparer les figures
plot(t_rk,Urk(1,:))
hold on;
plot(t,exp(-eps*w0*t).*(x0*cos(omega*t)+(eps*w0*x0+Dx0)/omega.*sin(omega*t)));
legend('Runge Kutta h=1.04','solution exacte');
```

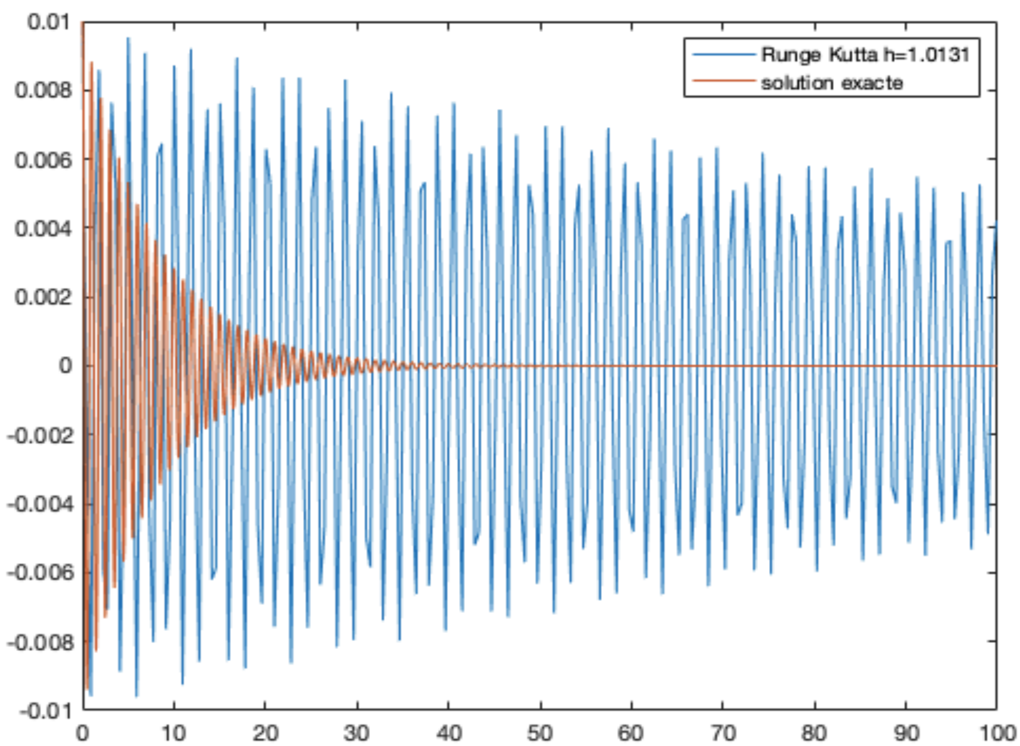


Le résultat n’est pas stable ni précis de la solution exacte.

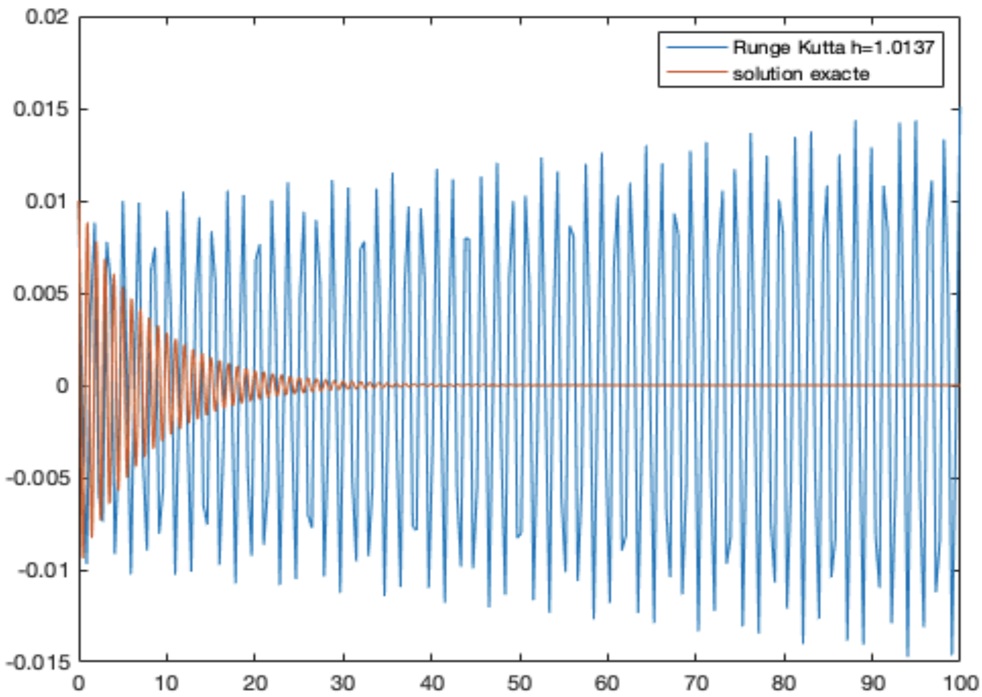
1.3.b

On utilise la méthode de dichotomie, et puis on peut confirm h_c est dans le intervalle à partir de la question 1.3.a : $1.0131 < h_c < 1.0137$

```
clf;
%quand h=1.013125
%h = 1.01375
h = 1.013125;
dt_rk = h*2*sqrt(2)/w0;
t_rk = 0:dt_rk:100*T0;
C = [0,1;-w0^2,-2*eps*w0];
Urk(:,1) = [x0;Dx0];
for i = 1:length(t_rk)-1
    k1 = C*Urk(:,i);
    k2 = C*(Urk(:,i)+1/2*dt_rk*k1);
    k3 = C*(Urk(:,i)+1/2*dt_rk*k2);
    k4 = C*(Urk(:,i)+dt_rk*k3);
    K = 1/6*(k1+2*k2+2*k3+k4);
    Urk(:,i+1) = Urk(:,i)+K*dt_rk;
end
%comparer les figures
plot(t_rk,Urk(1,:))
hold on;
plot(t,exp(-eps*w0*t).*(x0*cos(omega*t)+(eps*w0*x0+Dx0)/omega.*sin(omega*t)));
legend('Runge Kutta h=1.0131','solution exacte');
```



Quand h = 1.0131, la solution du schéma Runge-Kutta est amortie et stable.



En même temps, si h = 1.0137, la solution du schéma Runge-Kutta n'est plus stable. Donc on peut conclure: dt = $h_c \times \frac{2\sqrt{2}}{\omega_0}$ où

$1.0131 < h_c < 1.0137$.

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

1.Newmark explicite, gamma=0.5 beta=0

1.1

La matrice d’amplification est :

$$q_{n+1} = q_n + \Delta t\,\dot{q}_n + \Delta t^2 (0.5-\beta)\,\ddot{q}_n + \Delta t^2 \beta\,\ddot{q}_{n+1}$$

$$\dot{q}_{n+1} = \dot{q}_n + \Delta t\,(1-\gamma)\,\ddot{q}_n + \Delta t\,\gamma\,\ddot{q}_{n+1}$$

et $\ddot{q}_{n+1} = -w_0^2\, q_{n+1}$

$$\Rightarrow$$

$$\begin{bmatrix} 1+\beta\Delta t^2 w_0^2 & 0 \\ \gamma\Delta t\, w_0^2 & 1 \end{bmatrix} \begin{vmatrix} q_{n+1} \\ \dot{q}_{n+1} \end{vmatrix} = \begin{bmatrix} 1-\Delta t^2(0.5-\beta)w_0^2 & \Delta t \\ -(1-\gamma)\Delta t\, w_0^2 & 1 \end{bmatrix} \begin{vmatrix} q_n \\ \dot{q}_n \end{vmatrix}$$

$$[B] \begin{vmatrix} q_{n+1} \\ \dot{q}_{n+1} \end{vmatrix} = [C] \begin{vmatrix} q_n \\ \dot{q}_n \end{vmatrix}$$

la matrice d'amplification $A = [B]^{-1} \times [C]$

$$A = \begin{bmatrix} 1-\dfrac{w_0^2\Delta t^2}{2(1+\beta w_0^2\Delta t^2)} & \dfrac{\Delta t}{1+\beta w_0^2\Delta t^2} \\ -w_0^2\Delta t\left[\dfrac{\gamma w_0^2\Delta t^2}{2(1+\beta w_0^2\Delta t^2)}+1\right] & 1-\dfrac{\gamma w_0^2\Delta t^2}{1+\beta w_0^2\Delta t^2} \end{bmatrix}$$

1.2

On va determiner le pas de temps critique à partir de cette matrice d'amplification. Comme on a étudié pendant le cours, quand gamma egale 0.5, il faut que$\left(\gamma+\frac{1}{2}\right)^{2}-4\beta \leq \frac{4}{{\omega_0}^{2}\Delta t^{2}}$. On a déjà $\gamma = 0.5$ et $\beta = 0$, donc le pas de temps critique est :

```
clear all;
m = 2;
a = 0.5;
g = 9.81;
F0 = 20;
%calculer w0
w0 = sqrt(g/a);
w = 2*pi;
theta10 = 0;
theta20 = 0;
Dtheta10 = -1.31519275;
Dtheta20 = -1.85996342;
gamma1 = 0.5;
beta1 = 0;
```

```
Tc = sqrt(4/(w0^2*((gamma1+1/2)^2-4*beta1)))
```

```
Tc =

    0.4515
```

1.3

$$ma^2 \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \ddot{q}_0^{\,2} + mga \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} q_0 = 0$$

1.4

En présence d’une force dépendant du temps, on a :

$$q_{n+1} = q_n + \Delta t\,\dot{q}_n + \Delta t^2 (0.5-\beta)\,\ddot{q}_n + \Delta t^2 \beta\,\ddot{q}_{n+1}$$

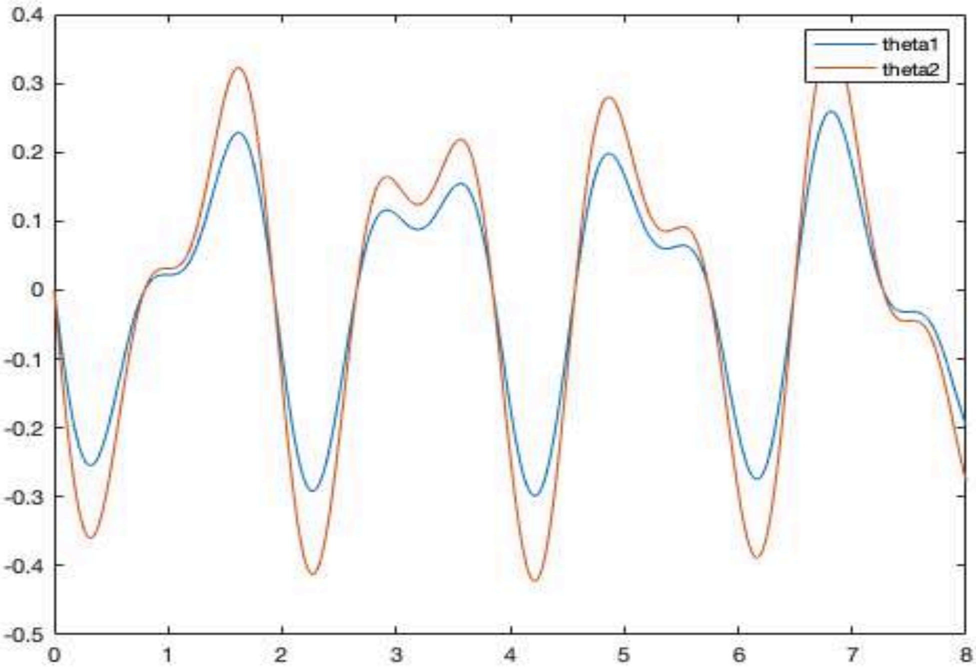
$$\dot{q}_{n+1} = \dot{q}_n + \Delta t\,(1-\gamma)\,\ddot{q}_n + \Delta t\,\gamma\,\ddot{q}_{n+1}$$

$$ma^2 \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \ddot{q}_n + mga \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} q_n = F_0 \sin\omega t \left| \begin{matrix} a \\ \frac{a}{\sqrt{2}} \end{matrix} \right.$$

1.5

```
%D2q+omega0*omega0*q = f*sin(w*t)
omega02 = inv([2,1;1,1])*m*g*a*[2,0;0,1]/(m*a^2);
f = F0*inv([2,1;1,1])*[a;a/sqrt(2)]/(m*a^2);

dt = 0.02;
t = 0:dt:T0;
B = [[1,0;0,1]+beta1*dt^2*omega02,0*[1,0;0,1];gamma1*dt*omega02,[1,0;0,1]];
C = [[1,0;0,1]-(0.5-beta1)*dt^2*omega02,dt*[1,0;0,1];-(1-gamma1)*dt*omega02,[1,0;0,1]];
A = inv(B)*C;
%calculer les valeurs propres
vp = eig(A);
re = [real(vp(2));real(vp(4))];
im = [imag(vp(2));imag(vp(4))];
mo = [abs(vp(2));abs(vp(4))];

U(:,1) = [q0;Dq0];
for i = 1:length(t)-1
    F = [dt^2*(0.5-beta1)*f*sin(w*i*dt);dt*(1-gamma1)*f*sin(w*i*dt)];
    U(:,i+1) = A*U(:,i)+inv(B)*F;
end
plot(t,U(1,:));
hold on;
plot(t,U(2,:));
legend('theta1','theta2')
```

Le résultat est comme ci-dessous :



1.6

Pour la valeur de t egale à 0s :

```
%t = 0s
q = U(1:2,1)
Dq = U(3:4,1)
D2q = f*sin(w*0*dt)-omega02*q
```

```
q =

     0
     0


Dq =

   -1.3152
   -1.8600


D2q =

     0
     0
```

Pour la valeur de t egale à dt :

```
%t = dt
q = U(1:2,2)
Dq = U(3:4,2)
D2q = f*sin(w*1*dt)-omega02*q
```

```
q =

   -0.0262
   -0.0370


Dq =

   -1.3048
   -1.8453


D2q =

    1.0348
    1.4634
```

Pour la valeur de t egale à 2*dt :

```
%t = 2*dt
q = U(1:2,3)
Dq = U(3:4,3)
D2q = f*sin(w*2*dt)-omega02*q
```

```
q =

  -0.0519
  -0.0734


Dq =

  -1.2813
  -1.8120


D2q =

   2.0533
   2.9038
```

Pour la valeur de t egale à 0.5s :

```
%t = 0.5s
q = U(1:2,26)
Dq = U(3:4,26)
D2q = f*sin(w*25*dt)-omega02*q
```

```
q =

  -0.1817
  -0.2569


Dq =

   0.6934
   0.9806


D2q =

   2.0879
   2.9527
```

2.Newmark implicite, gamma=0.5 beta=0.25

2.1

La matrice d'amplification est :

$$q_{n+1} = q_n + \Delta t\, \dot{q}_n + \Delta t^2 (0.5-\beta)\, \ddot{q}_n + \Delta t^2 \beta\, \ddot{q}_{n+1}$$

$$\dot{q}_{n+1} = \dot{q}_n + \Delta t\,(1-\gamma)\, \ddot{q}_n + \Delta t\, \gamma\, \ddot{q}_{n+1}$$

et $\ddot{q}_{n+1} = -w_0^2\, q_{n+1}$

$\Rightarrow$

$$\begin{bmatrix} 1+\beta \Delta t^2 w_0^2 & 0 \\ \gamma \Delta t\, w_0^2 & 1 \end{bmatrix} \begin{Bmatrix} q_{n+1} \\ \dot{q}_{n+1} \end{Bmatrix} = \begin{bmatrix} 1-\Delta t^2 (0.5-\beta) w_0^2 & \Delta t \\ -(1-\gamma)\Delta t\, w_0^2 & 1 \end{bmatrix} \begin{Bmatrix} q_n \\ \dot{q}_n \end{Bmatrix}$$

$$[B] \begin{Bmatrix} q_{n+1} \\ \dot{q}_{n+1} \end{Bmatrix} = [C] \begin{Bmatrix} q_n \\ \dot{q}_n \end{Bmatrix}$$

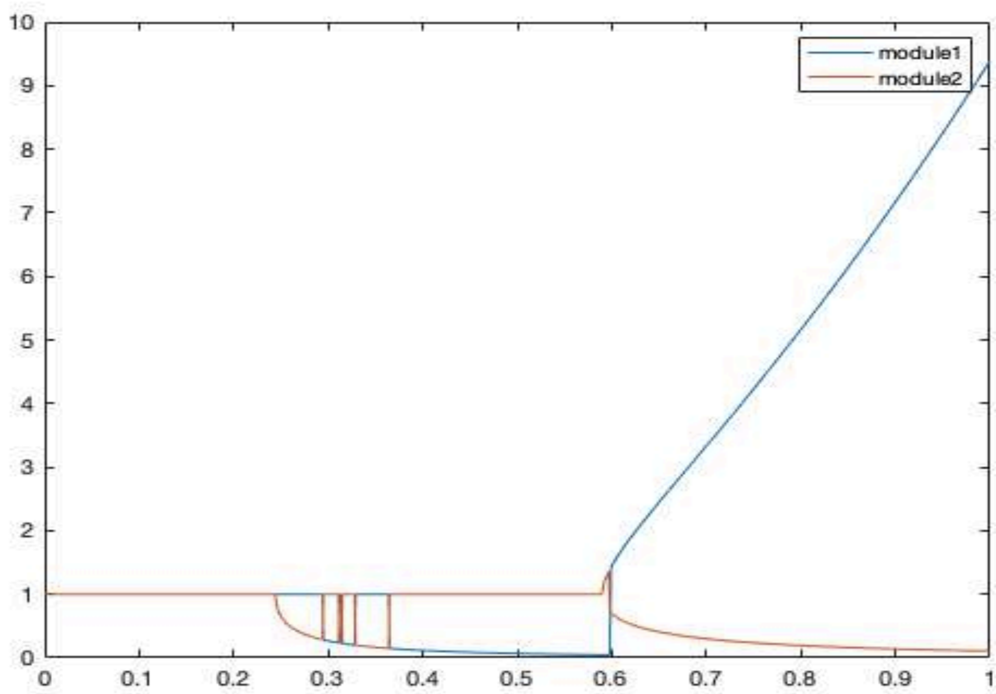
la matrice d'amplification $A = [B]^{-1} \times [C]$

$$A = \begin{bmatrix} 1 - \dfrac{w_0^2 \Delta t^2}{2(1+\beta w_0^2 \Delta t^2)} & \dfrac{\Delta t}{1+\beta w_0^2 \Delta t^2} \\ -w_0^2 \Delta t \left[-\dfrac{\gamma w_0^2 \Delta t^2}{2(1+\beta w_0^2 \Delta t^2)} + 1 \right] & 1 - \dfrac{\gamma w_0^2 \Delta t^2}{1+\beta w_0^2 \Delta t^2} \end{bmatrix}$$

2.2

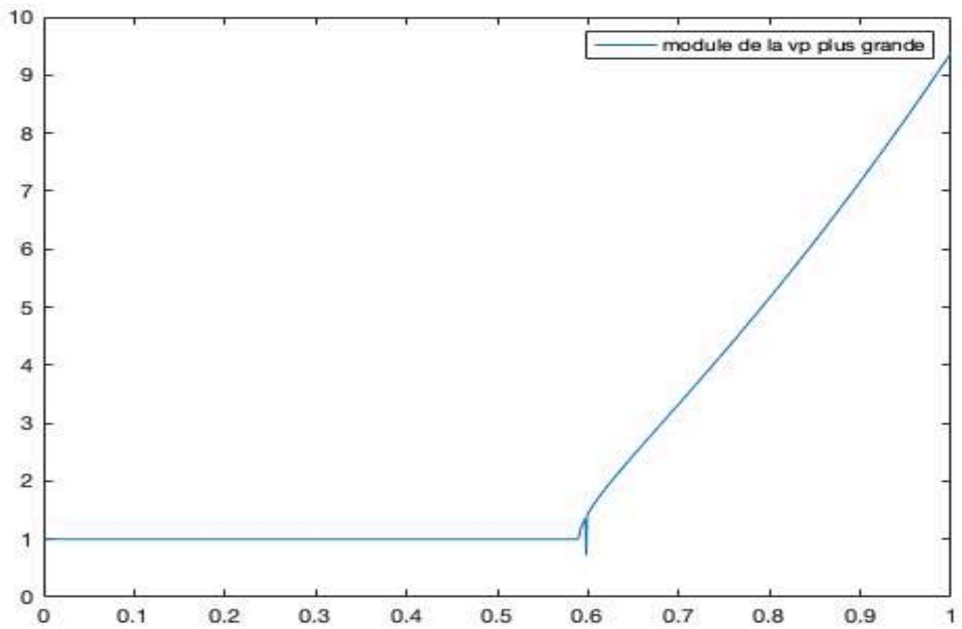
```
clf;
dt2 = 0:0.001:1;
for j = 1:length(dt2)
    B2{1,j} = [[1,0;0,1]+beta1*dt2(j)^2*omega02,0*[1,0;0,1];gamma1*dt2(j)*omega02,[1,0;0,1]];
    C2{1,j} = [[1,0;0,1]-(0.5-beta1)*dt2(j)^2*omega02,dt2(j)*[1,0;0,1];-(1-gamma1)*dt2(j)*omega02,[1,0;0,1]];
    A2{1,j} = inv(B2{1,j})*C2{1,j};
    %calculer les valeurs propres
    vp2(:,j) = eig(A2{1,j});
    re2(:,j) = [real(vp2(2,j));real(vp2(4,j))];
    im2(:,j) = [imag(vp2(2,j));imag(vp2(4,j))];

    mo2(:,j) = [abs(vp2(2,j));abs(vp2(4,j))];
end
plot(dt2,mo2(1,:));
hold on;
plot(dt2,mo2(2,:));
legend('module1','module2');
%legend('module de la vp plus grande');
```

On présente les deux modules dans un même graphique :



Et puis on choisit le plus grande valeur propre :

```
clf;
dt2 = 0:0.001:1;
for j = 1:length(dt2)
    B2(1,j) = [[1,0;0,1]+beta1*dt2(j)^2*omega02,0*[1,0;0,1];gamma1*dt2(j)*omega02,[1,0;0,1]];
    C2(1,j) = [[1,0;0,1]-(0.5-beta1)*dt2(j)^2*omega02,dt2(j)*[1,0;0,1];-(1-gamma1)*dt2(j)*omega02,[1,0;0,1]];
    A2(1,j) = inv(B2(1,j))*C2(1,j);
    %calculer les valeurs propres
    vp2(:,j) = eig(A2(1,j));
    re2(:,j) = [real(vp2(2,j));real(vp2(4,j))];
    im2(:,j) = [imag(vp2(2,j));imag(vp2(4,j))];
    %choisir la plus grande valeur propre
    mo2(j) = max(abs(vp2(2,j)),abs(vp2(4,j)));
end
plot(dt2,mo2);
legend('module de la vp plus grande');
```



2.3

$$ma^2 \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \ddot{q}_0^{\,2} + mga \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} q_0 = 0$$

2.4

En présence d’une force dépendant du temps, on a :

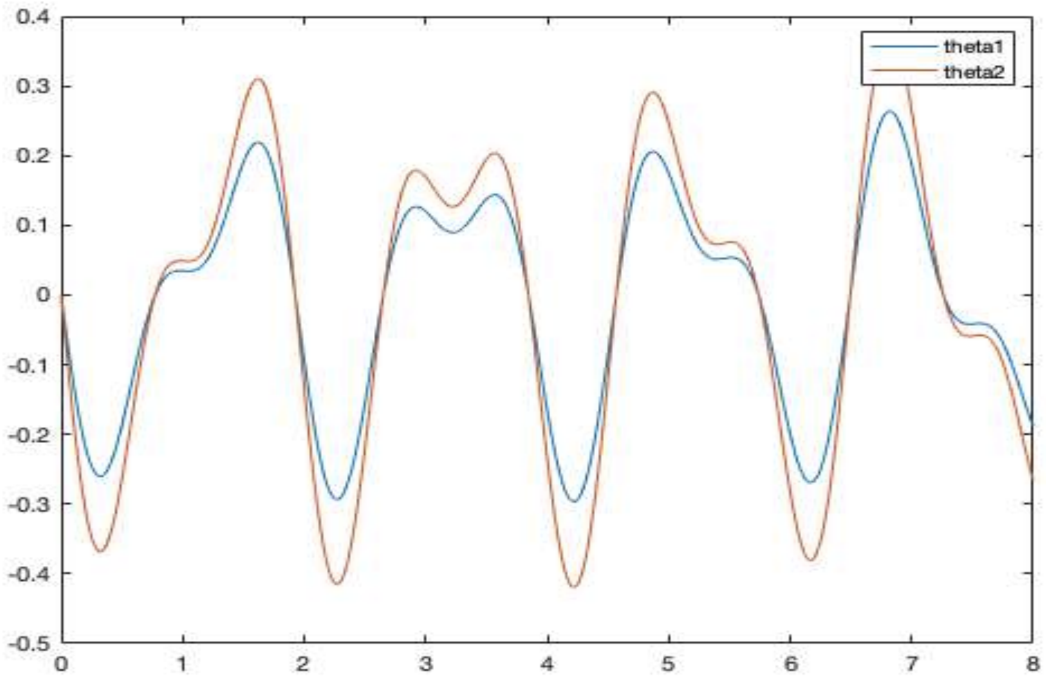
$$q_{n+1} = q_n + \Delta t\, \dot{q}_n + \Delta t^2 (0.5 - \beta)\, \ddot{q}_n + \Delta t^2 \beta\, \ddot{q}_{n+1}$$

$$\dot{q}_{n+1} = \dot{q}_n + \Delta t\, (1 - \gamma)\, \ddot{q}_n + \Delta t\, \gamma\, \ddot{q}_{n+1}$$

$$ma^2 \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \ddot{q}_n + mga \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} q_n = F_0 \sin \omega t \left| \begin{matrix} a \\ \frac{a}{\sqrt{2}} \end{matrix} \right.$$

2.5

```
%D2q+omega0*omega0*q = f*sin(w*t)
omega02 = inv([2,1;1,1])*m*g*a*[2,0;0,1]/(m*a^2);
f = F0*inv([2,1;1,1])*[a;a/sqrt(2)]/(m*a^2);

dt = 0.02;
t = 0:dt:T0;
B = [[1,0;0,1]+beta2*dt^2*omega02,0*[1,0;0,1];gamma2*dt*omega02,[1,0;0,1]];
C = [[1,0;0,1]-(0.5-beta2)*dt^2*omega02,dt*[1,0;0,1];-(1-gamma2)*dt*omega02,[1,0;0,1]];
A = inv(B)*C;
U(:,1) = [q0;Dq0];
for i = 1:length(t)-1
    F = [dt^2*(0.5-beta2)*f*sin(w*i*dt);dt*(1-gamma2)*f*sin(w*i*dt)];
    U(:,i+1) = A*U(:,i)+inv(B)*F;
end
plot(t,U(1,:));
hold on;
plot(t,U(2,:));
legend('theta1','theta2');
```



2.6

Pour la valeur de t egale à 0s :

```
%quand t = 0s
q = U(1:2,1)
Dq = U(3:4,1)
D2q = f*sin(w*25*dt)-omega02*q
```

```
q =

     0
     0


Dq =

   -1.3152
   -1.8600


D2q =

   1.0e-14 *

    0.0717
    0.1015
```

Pour la valeur de t egale à dt :

```
%quand t = dt
q = U(1:2,2)
Dq = U(3:4,2)
D2q = f*sin(w*25*dt)-omega02*q
```

```
q =

   -0.0262
   -0.0371


Dq =

   -1.3048
   -1.8453


D2q =

    0.3011
    0.4259
```

Pour la valeur de t egale à 2*dt :

```
%quand t = 2*dt
q = U(1:2,3)
Dq = U(3:4,3)
D2q = f*sin(w*25*dt)-omega02*q
```

```
q =

   -0.0521
   -0.0736


Dq =

   -1.2813
   -1.8120


D2q =

    0.5983
    0.8462
```

Pour la valeur de t egale à 0.5s :

```
%quand t = 0.5s
q = U(1:2,26)
Dq = U(3:4,26)
D2q = f*sin(w*25*dt)-omega02*q
```

```
q =

   -0.1872
   -0.2647


Dq =

    0.7152
    1.0115


D2q =

    2.1515
    3.0427
```

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

1. Newmark explicite

1.1

$$q_{j+1} = q_j + \Delta t\,\dot{q}_j + \Delta t^2 (0.5-\beta)\,\ddot{q}_j + \beta\,\Delta t^2\,\ddot{q}_{j+1}$$

$$\dot{q}_{j+1} = \dot{q}_j + \Delta t(1-\gamma)\,\ddot{q}_j + \gamma\,\Delta t\,\ddot{q}_{j+1}$$

$$\ddot{q}_j + w_0^2\, q_j\,(1 + a q_j^2) = 0$$

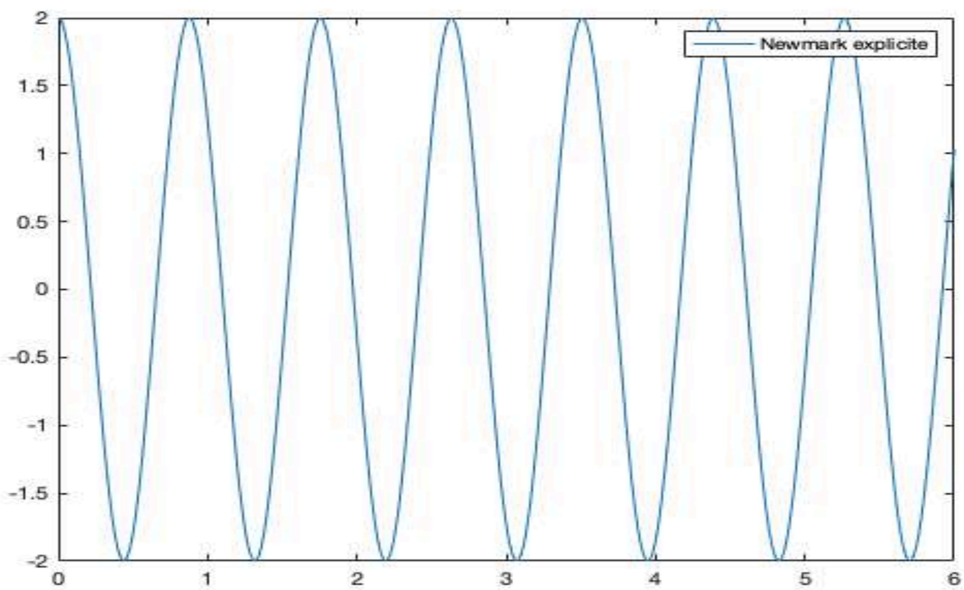
1.2

```
w0 = 2*pi;
a = 0.1;
q0 = 2;
Dq0 = 0;
gamma1 = 0.5;
beta1 = 0;
q(1) = q0;
Dq(1) = Dq0;
T0 = 6;
```

1.2

```
dt = 0.02;
t = 0:dt:T0;
for i = 1:length(t)-1
    D2q(i) = -w0^2*q(i)*(1+a*q(i)*q(i));
    q(i+1) = q(i)+dt*Dq(i)+dt^2*(0.5-beta1)*D2q(i);
    Dq(i+1) = Dq(i)+dt*(1-gamma1)*D2q(i)+gamma1*dt*(-w0^2*q(i+1)*(1+a*q(i+1)*q(i+1)));
end
plot(t,q);
legend('Newmark explicite');
```

Le résultat est comme ci-dessous :



1.3

Quand t = 0s:

```
%quand t = 0s
q1 = q(1)
Dq1 = Dq(1)
D2q1 = D2q(1)
```

```
q1 =

     2


Dq1 =

     0


D2q1 =

 -110.5396
```

Quand t = dt:

```
%quand t = dt
q2 = q(2)
Dq2 = Dq(2)
D2q2 = D2q(2)
```

```
q2 =

    1.9779


Dq2 =

   -2.1917


D2q2 =

 -108.6310
```

Quand t = 2*dt:

```
%quand t = 2*dt
q3 = q(3)
Dq3 = Dq(3)
D2q3 = D2q(3)
```

```
q3 =

    1.9123


Dq3 =

   -4.3091


D2q3 =

 -103.1048
```

Quand t = T0 :

```
%quand t = T0
qT0 = q(length(t))
DqT0 = Dq(length(t))
D2qT0 = D2q(length(t))
```

```
qT0 =

    1.0329


DqT0 =

   12.0118


D2qT0 =

  -45.1299
```