

Prénom français : Léa

Nom et prénom chinois : Wu Tong

Numéro d'étudiante :SY1924137

Remarque :Ce document contient tous les devoirs de ce cours

Oscillateur conservatif à un degré de liberté

- Le schéma Runge Kutta

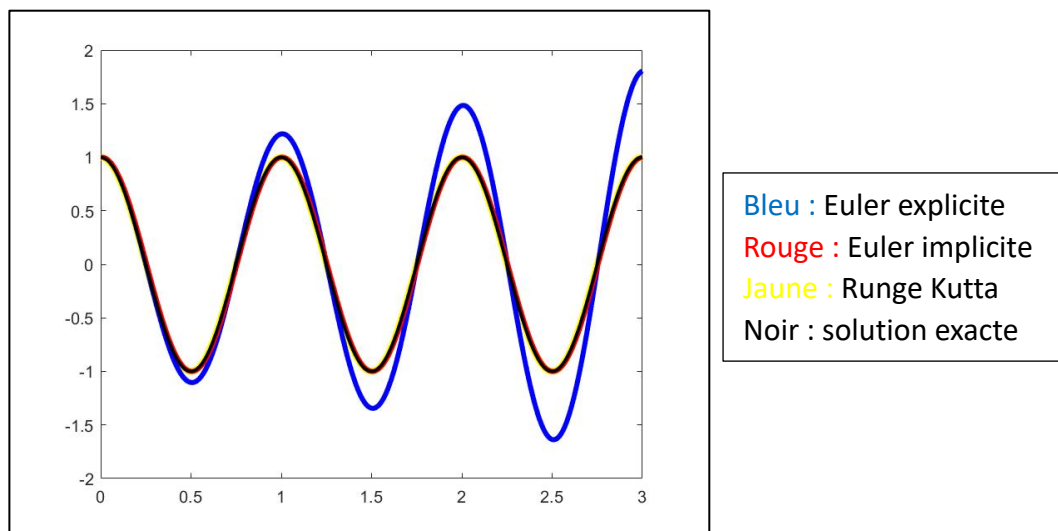
Code en Matlab :

```
dt = 0.04;
t = (0:dt:3)';
np = size(t,1);
q0 = 1;
dq0 = 0;
q = zeros(np,1);
dq = zeros(np,1);
q(1) = q0;
dq(1) = dq0;
qj=[q0;dq0];
for i = 2:np
    tc = t(i-1);
    xc=qj;
    k1=cal_f(xc, tc);
    xc=qj+k1*dt/2;
    k2=cal_f(xc, tc+dt/2);
    xc=qj+k2*dt/2;
    k3=cal_f(xc, tc+dt/2);
    xc=qj+k3*dt;
    k4=cal_f(xc, tc+dt);
    dq=(k1+2*k2+2*k3+k4)/6;
    qj=qj+dq*dt;
    q(i)=qj(1)
    dq(i)=qj(2);
end
plot(t, q, 'b-', 'Linewidth', 3)
```

```
function[dU] = cal_f(U, tc)
    w = 2 * pi;
    dU = zeros(2, 1);
    dU(1) = U(2);
    dU(2) = - w*w*U(1);
end
```

Résultat obtenu :

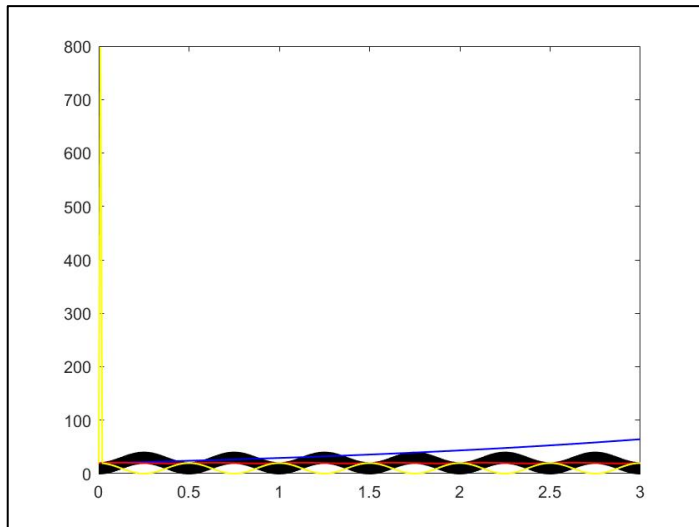
(1) Les valeurs de la solution $q(t)$ obtenu avec le schéma de *Runge Kutta*,
Euler explicite, *Euler implicite* et *solution exacte*



C'est les résultats $q(t)$ obtenus par des différents schémas quand on prend $\Delta t = 0.01s$

Les résultats de schémas Euler implicite, Runge Kutta et solution exacte sont presque coïncidents. Mais le résultat de schéma Euler explicite est divergent.

(2) Les valeurs de la quantité E^* obtenu avec le schéma de *Runge Kutta*,
Euler explicite, *Euler implicite* et *solution exacte*



Bleu : Euler explicite
 Rouge : Euler implicite
 Jaune : Runge Kutta
 Noir : solution exacte

C'est la quantité E^* obtenue par des différents schémas quand on prend $\Delta t = 0.01s$

- Le schéma Euler explicite

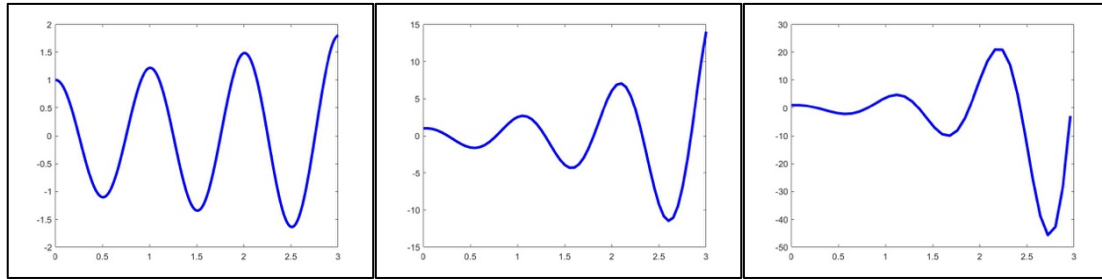
Code en Matlab :

```

dt = 0.05;
T = 3;
q0 = 1;
dq0 = 0;
w0 = 2*pi;
t = (0:dt:T)';
np = size(t,1);
q = zeros(np,1);
dq = zeros(np,1);
ddq = zeros(np,1);
q(1) = q0;
dq(1) = dq0;
ddq(1) = - w0 * w0 * q(1);
for i = 2:np
    q(i) = q(i-1)+ dt * dq(i-1);
    dq(i) = dq(i-1)+ dt * ddq(i-1);
    ddq(i) = - w0 * w0 * q(i);
end
plot(t,q,'b-', 'Linewidth', 3)

```

Résultat obtenu :



$\Delta t = 0.01s$

$\Delta t = 0.05s$

$\Delta t = 0.08s$

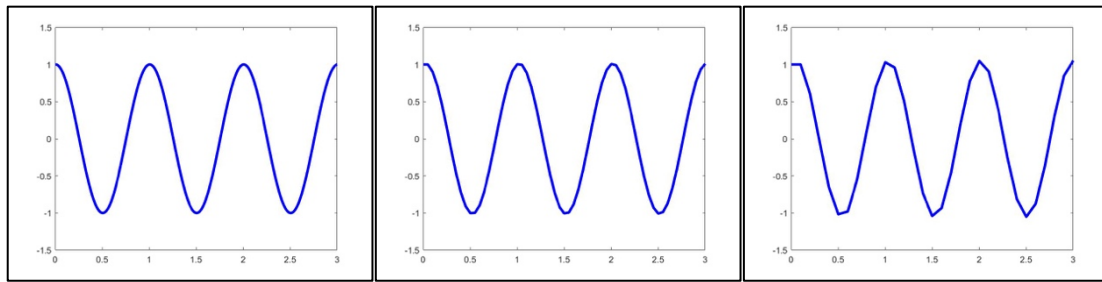
Et quand on change le pas de temps Δt , on peut voir que plus le pas de temps Δt est petit, plus la divergence est lente.

● Le schéma implicite

Code en Matlab :

```
dt = 0.05;
T = 3;
q0 = 1;
dq0 = 0;
w0 = 2*pi;
t = (0:dt:T)';
np = size(t,1);
q = zeros(np,1);
dq = zeros(np,1);
ddq = zeros(np,1);
q(1) = q0;
dq(1) = dq0;
ddq(1) = -w0 * w0 * q(1);
for i = 2:np
    q(i) = q(i-1) + dt * (dq(i-1) - dt * q(i) * w0 * w0);
    ddq(i) = -w0 * w0 * q(i);
    dq(i) = dq(i-1) + dt * ddq(i);
end
plot(t, q, 'b-', 'Linewidth', 3)
```

Résultat obtenu :



$\Delta t = 0.01s$

$\Delta t = 0.05s$

$\Delta t = 0.1s$

Et quand on change le pas de temps Δt , on peut voir que plus le pas de temps Δt est petit, plus l'atténuation des oscillations est faible.

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

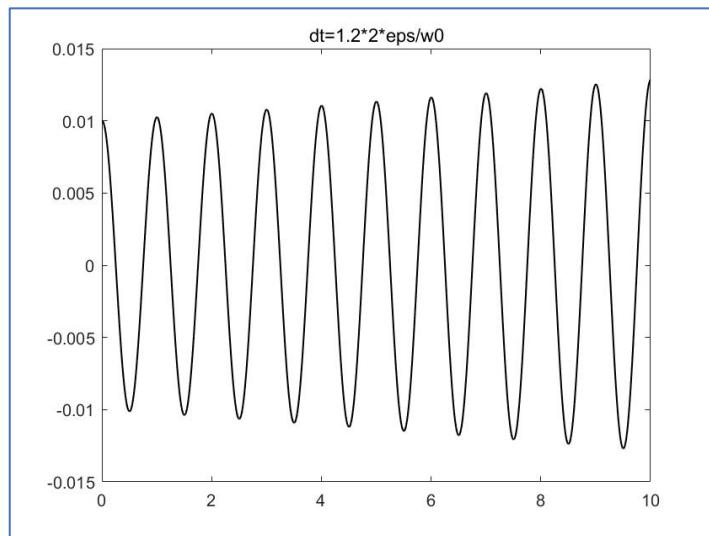
1.1) Résolution avec un schéma d'EULER explicite

$$1.1.a) \frac{2\varepsilon}{\omega_0} < \Delta t$$

Code :

```
w0 = 2*pi;
eps = 0.02;
dt = 1.2 * 2 * eps / w0 ;
T = 10;
q0 = 0.01;
dq0 = 0;
t = (0:dt:T)';
np = size(t,1);
q = zeros(np,1);
dq = zeros(np,1);
ddq = zeros(np,1);
energ = zeros(np,1);
q(1) = q0;
dq(1) = dq0;
ddq(1) = - w0 * w0 * q(1);
for i = 2:np
    q(i) = q(i-1)+ dt * dq(i-1);
    dq(i) = dq(i-1)+ dt * ddq(i-1);
    ddq(i) = - w0 * w0 * q(i) - 2 * eps * w0 * dq(i);
end
l1 = plot(t,q , 'k-', 'Linewidth',1);
title(' dt=1.2*2*eps/w0' )
hold on
```

Résultat :

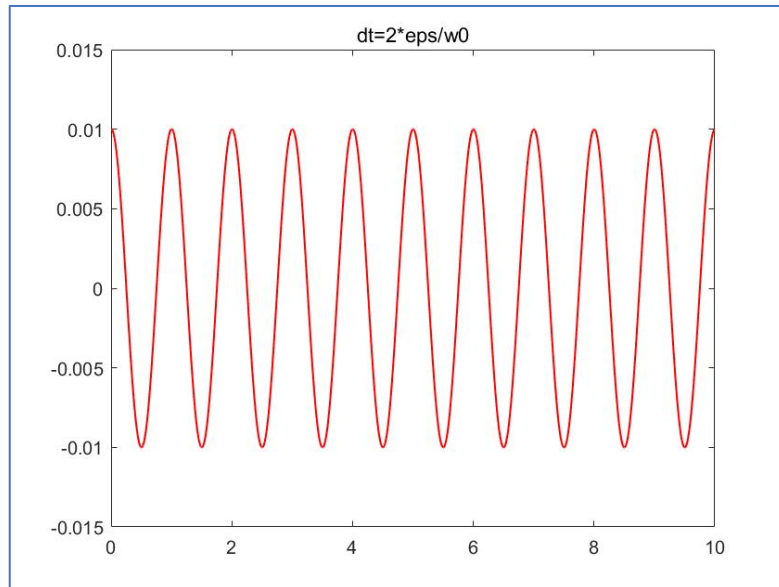


$$1.1.b) \Delta t = \frac{2\varepsilon}{\omega_0}$$

Code :

```
w0 = 2*pi;
eps = 0.02;
dt = 2 * eps / w0 ;
T = 10;
q0 = 0.01;
dq0 = 0;
t = (0:dt:T)';
np = size(t,1);
q = zeros(np,1);
dq = zeros(np,1);
ddq = zeros(np,1);
energ = zeros(np,1);
q(1) = q0;
dq(1) = dq0;
ddq(1) = - w0 * w0 * q(1);
for i = 2:np
    q(i) = q(i-1)+ dt * dq(i-1);
    dq(i) = dq(i-1)+ dt * ddq(i-1);
    ddq(i) = - w0 * w0 * q(i) - 2 * eps * w0 * dq(i);
end
l2 = plot(t, q , 'r-', 'Linewidth', 1);
title(' dt=2*eps/w0')
hold on
```

Résultat :

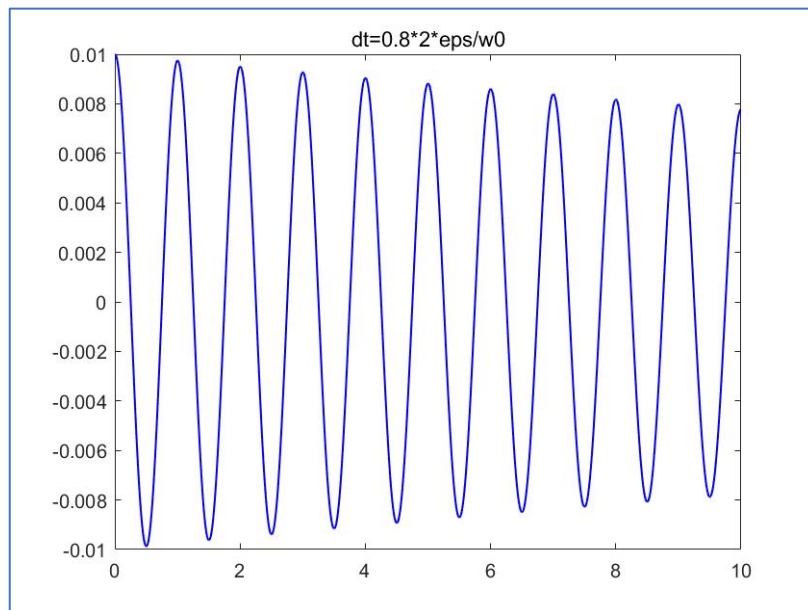


$$1.1.c) \Delta t = 0.8 \times \frac{2\varepsilon}{\omega_0}$$

Code :

```
w0 = 2*pi;
eps = 0.02;
dt = 0.8 * 2 * eps / w0 ;
T = 10;
q0 = 0.01;
dq0 = 0;
t = (0:dt:T)';
np = size(t,1);
q = zeros(np,1);
dq = zeros(np,1);
ddq = zeros(np,1);
energ = zeros(np,1);
q(1) = q0;
dq(1) = dq0;
ddq(1) = - w0 * w0 * q(1);
for i = 2:np
    q(i) = q(i-1)+ dt * dq(i-1);
    dq(i) = dq(i-1)+ dt * ddq(i-1);
    ddq(i) = - w0 * w0 * q(i) - 2 * eps * w0 * dq(i);
end
l1 = plot(t,q , 'b-', 'Linewidth',1);
title(' dt=0.8*2*eps/w0')
hold on
```


Résultat :



1.2) Résolution avec un schéma d'EULER implicite

Le pas de temps critique est $\frac{2\varepsilon}{\omega_0}$

1.3) Résolution avec un schéma de RUNGE KUTTA

1.3.a)

h=0.04

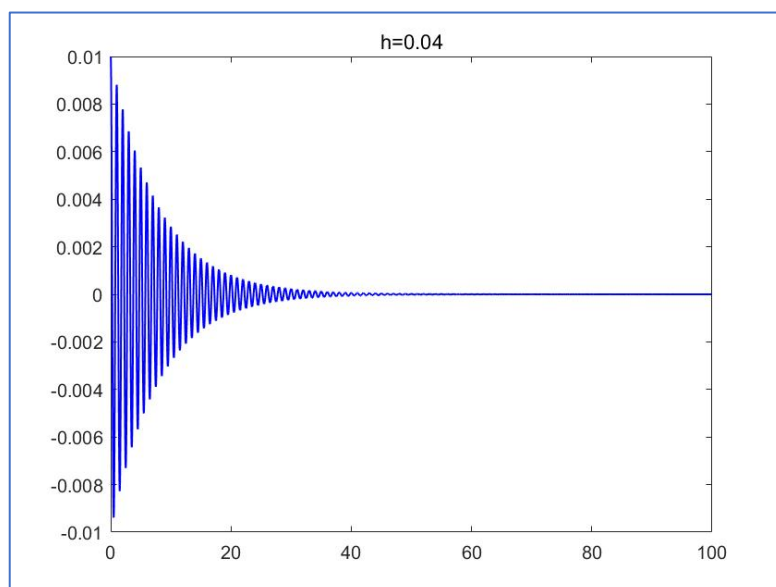
Code :

```

h = 0.04; w = 2 * pi;
dt = h * 2 * sqrt(2)/w ;
t = (0:dt:100)';
np = size(t,1);
q0 = 0.01; dq0 = 0;
q = zeros(np,1);
dq = zeros(np,1);
q(1) = q0;
dq(1) = dq0;
qj=[q0;dq0];
for i = 2:np
    tc = t(i-1);
    xc=qj;
    k1=cal_f(xc,tc);
    xc=qj+k1*dt/2;
    k2=cal_f(xc,tc+dt/2);
    xc=qj+k2*dt/2;
    k3=cal_f(xc,tc+dt/2);
    xc=qj+k3*dt;
    k4=cal_f(xc,tc+dt);
    dq=(k1+2*k2+2*k3+k4)/6;
    qj=qj+dq*dt;
    q(i)=qj(1);
    dq(i)=qj(2);
end
plot(t,q,'b-','Linewidth',1)
title('h=0.04')

```

Résultat :

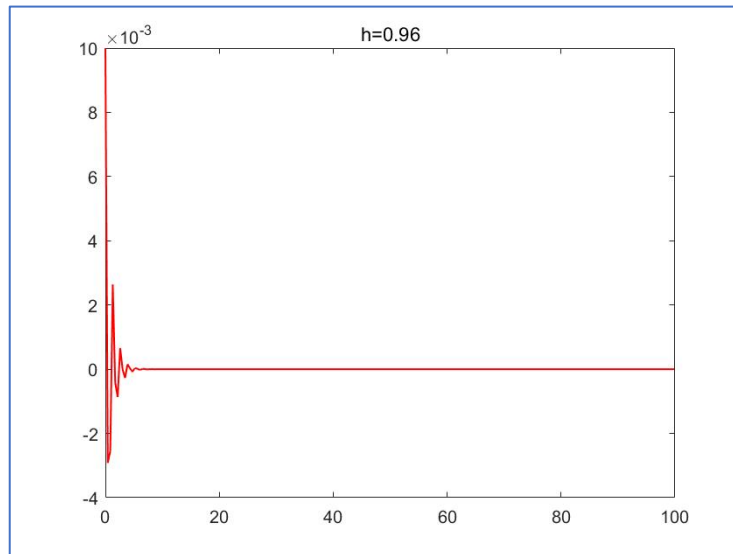


h=0.96

Code :

```
h = 0.96;w = 2 * pi;
dt = h * 2 * sqrt(2)/w ;
t = (0:dt:100)';
np = size(t,1);
q0 = 0.01; dq0 = 0;
q = zeros(np,1);
dq = zeros(np,1);
q(1) = q0;
dq(1) = dq0;
qj=[q0;dq0];
for i = 2:np
    tc = t(i-1);
    xc=qj;
    k1=cal_f(xc, tc);
    xc=qj+k1*dt/2;
    k2=cal_f(xc, tc+dt/2);
    xc=qj+k2*dt/2;
    k3=cal_f(xc, tc+dt/2);
    xc=qj+k3*dt;
    k4=cal_f(xc, tc+dt);
    dq=(k1+2*k2+2*k3+k4)/6;
    qj=qj+dq*dt;
    q(i)=qj(1);
    dq(i)=qj(2);
end
plot(t,q , 'r-', 'Linewidth',1)
title('h=0.96')
```

Résultat :



h=1.04

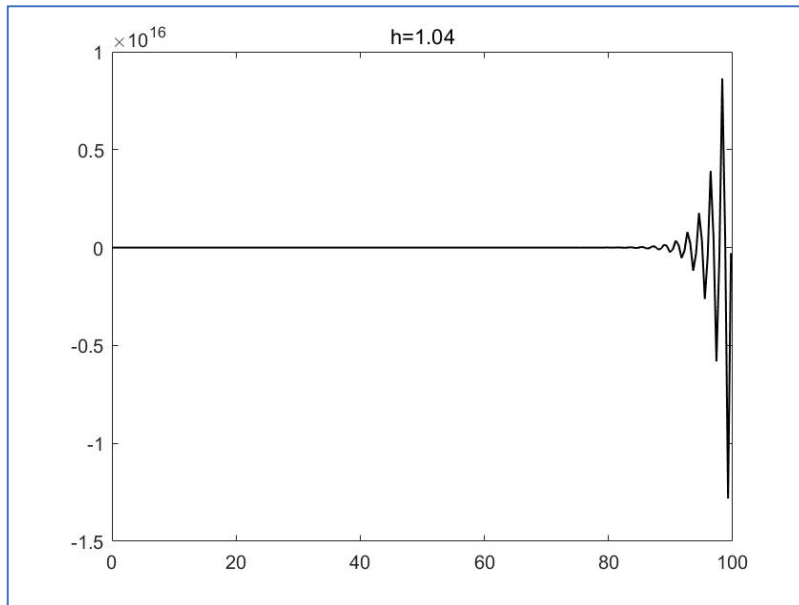
Code :

```

h = 1.04; w = 2 * pi;
dt = h * 2 * sqrt(2)/w ;
t = (0:dt:100)';
np = size(t,1);
q0 = 0.01; dq0 = 0;
q = zeros(np, 1);
dq = zeros(np, 1);
q(1) = q0;
dq(1) = dq0;
qj=[q0;dq0];
for i = 2:np
    tc = t(i-1);
    xc=qj;
    k1=cal_f(xc, tc);
    xc=qj+k1*dt/2;
    k2=cal_f(xc, tc+dt/2);
    xc=qj+k2*dt/2;
    k3=cal_f(xc, tc+dt/2);
    xc=qj+k3*dt;
    k4=cal_f(xc, tc+dt);
    dq=(k1+2*k2+2*k3+k4)/6;
    qj=qj+dq*dt;
    q(i)=qj(1);
    dq(i)=qj(2);
end
plot(t, q , 'k-', 'Linewidth', 1)
title('h=1.04')

```

Résultat :



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

1. Résolution avec un schéma de NEWMARK explicite

1.1) La matrice d'amplification est

$$A = \begin{pmatrix} 1 - \frac{\omega_0^2 \Delta t^2}{2(1 + \beta \omega_0^2 \Delta t^2)} & \frac{\Delta t}{1 + \beta \omega_0^2 \Delta t^2} \\ -\omega_0^2 \Delta t \left[1 - \frac{\gamma \omega_0^2 \Delta t^2}{2(1 + \beta \omega_0^2 \Delta t^2)} \right] & 1 - \frac{\gamma \omega_0^2 \Delta t^2}{1 + \beta \omega_0^2 \Delta t^2} \end{pmatrix}$$

Comme $\beta = 0, \gamma = 0.5, \omega_0 = 2 * \pi,$

on a
$$A = \begin{pmatrix} 1 - 2\pi^2 \Delta t^2 & \Delta t \\ -4\pi^2 \Delta t [1 - \pi^2 \Delta t^2] & 1 - 2\pi^2 \Delta t^2 \end{pmatrix}$$

1.2) Le pas de temps critique est $\Delta t = \frac{1}{\pi}$

1.3) La relation entre q_0, \dot{q}_0 et \ddot{q}_0 est

$$\ddot{q}_0 = - \begin{pmatrix} 1 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}^{-1} * \begin{pmatrix} 19.62 & 0 \\ 0 & 9.81 \end{pmatrix} * q_0$$

1.4) Les trois relations sont

$$q_{j+1} = q_j + \Delta t * \dot{q}_j + 0.5 * \Delta t^2 * \ddot{q}_j$$

$$\ddot{q}_{j+1} = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}^{-1} * (\sin(2\pi t) * \begin{pmatrix} 10 \\ 5\sqrt{2} \end{pmatrix} - \begin{pmatrix} 19.62 & 0 \\ 0 & 9.81 \end{pmatrix} * q_{j+1})$$

$$\dot{q}_{j+1} = \dot{q}_j + 0.5 * \Delta t * (\ddot{q}_j + \ddot{q}_{j+1})$$

1.5) Programmation

Code :

```

dt = 0.01;
T = 8;
q0 = [0, 0];
dq0 = [-1.31519275, -1.85996342];
t = (0:dt:T)';
np = size(t, 1);
q = zeros(np, 2);
dq = zeros(np, 2);
w = 2 * pi;
q(1, :) = q0;
dq(1, :) = dq0;
ddq(1, :) = (-inv([1, 0.5; 0.5, 0.5])*[19.62, 0; 0, 9.81] * (q(1, :)))';
for i = 2:np
    q(i, :) = q(i-1, :) + dt * dq(i-1, :) + dt * dt * 0.5 * ddq(i-1, :);
    ddq(i, :) = (inv([1, 0.5; 0.5, 0.5])*(-[19.62, 0; 0, 9.81] * (q(i, :)) + [10; 5*sqrt(2)] * sin(2 * pi * i * dt)))';
    dq(i, :) = dq(i-1, :) + 0.5 * dt * (ddq(i-1, :) + ddq(i, :));
end
plot(t, q(:, 1), 'b-', 'Linewidth', 1)
hold on
plot(t, q(:, 2), 'k-', 'Linewidth', 1)
legend('theta1', 'theta2')

```

1.6) Des valeurs sont dans le tableau au-dessous :

t \	q(t)	q̇(t)	q̈(t)
0	$\begin{pmatrix} 0 \\ 0 \end{pmatrix}$	$\begin{pmatrix} -1.3152 \\ -1.8600 \end{pmatrix}$	$\begin{pmatrix} 0 \\ 0 \end{pmatrix}$
Δt	$\begin{pmatrix} -0.0132 \\ -0.0186 \end{pmatrix}$	$\begin{pmatrix} -1.3108 \\ -1.8537 \end{pmatrix}$	$\begin{pmatrix} 0.8853 \\ 1.2521 \end{pmatrix}$
$2\Delta t$	$\begin{pmatrix} -0.0262 \\ -0.0371 \end{pmatrix}$	$\begin{pmatrix} -1.2993 \\ -1.8376 \end{pmatrix}$	$\begin{pmatrix} 1.3990 \\ 1.9784 \end{pmatrix}$
0.5	$\begin{pmatrix} -0.2013 \\ -0.2847 \end{pmatrix}$	$\begin{pmatrix} 0.0460 \\ 0.0651 \end{pmatrix}$	$\begin{pmatrix} 8.1603 \\ 11.5404 \end{pmatrix}$

2. Résolution avec un schéma de NEWMARK implicite

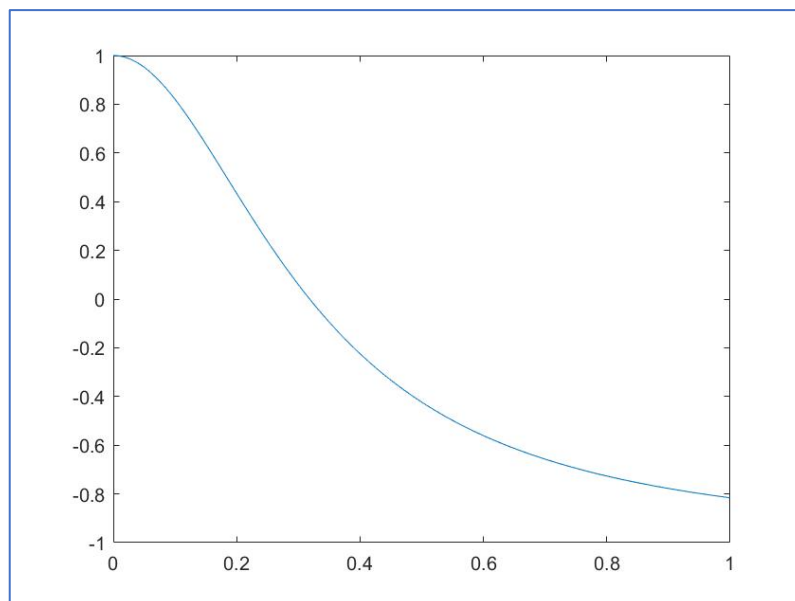
2.1) La matrice d'amplification est

$$A = \begin{pmatrix} 1 - \frac{\omega_0^2 \Delta t^2}{2(1 + \beta \omega_0^2 \Delta t^2)} & \frac{\Delta t}{1 + \beta \omega_0^2 \Delta t^2} \\ -\omega_0^2 \Delta t \left[1 - \frac{\gamma \omega_0^2 \Delta t^2}{2(1 + \beta \omega_0^2 \Delta t^2)} \right] & 1 - \frac{\gamma \omega_0^2 \Delta t^2}{1 + \beta \omega_0^2 \Delta t^2} \end{pmatrix}$$

Comme $\beta = 0.25$, $\gamma = 0.5$, $\omega_0 = 2 * \pi$,

$$A = \begin{pmatrix} 1 - \frac{2\pi^2 \Delta t^2}{1 + \pi^2 \Delta t^2} & \frac{\Delta t}{1 + \pi^2 \Delta t^2} \\ -4\pi^2 \Delta t \left[1 - \frac{\pi^2 \Delta t^2}{1 + \pi^2 \Delta t^2} \right] & 1 - \frac{2\pi^2 \Delta t^2}{1 + \pi^2 \Delta t^2} \end{pmatrix}$$

2.2) L'allure de l'évolution de la plus grande valeur propre en fonction du pas de temps est



2.3) La relation entre q_0 , \dot{q}_0 et \ddot{q}_0 est

$$\ddot{q}_0 = - \begin{pmatrix} 1 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}^{-1} * \begin{pmatrix} 19.62 & 0 \\ 0 & 9.81 \end{pmatrix} * q_0$$

2.4)

$$q_{j+1} = q_j + \Delta t * \dot{q}_j + 0.25 * \Delta t^2 * (\ddot{q}_j + \ddot{q}_{j+1})$$

$$\dot{q}_{j+1} = \dot{q}_j + 0.5 * \Delta t * (\ddot{q}_j + \ddot{q}_{j+1})$$

$$\ddot{q}_{j+1} = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}^{-1} * (\sin(2\pi t) * \begin{pmatrix} 10 \\ 5\sqrt{2} \end{pmatrix} - \begin{pmatrix} 19.62 & 0 \\ 0 & 9.81 \end{pmatrix} * q_{j+1})$$

2.5)

Code :

```

dt = 0.01;
T = 8;
q0 = [0, 0];
dq0 = [-1.31519275, -1.85996342];
t = (0:dt:T)';
np = size(t, 1);
q = zeros(np, 2);
dq = zeros(np, 2);
ddq = zeros(np, 2);
w = 2 * pi;
q(1, :) = q0;
dq(1, :) = dq0;
ddq(1, :) = (inv([1, 0.5; 0.5, 0.5]) * (-[19.62, 0; 0, 9.81] * (q(1, :))'))';
ddq(1, :)
for i = 2:np
    q(i, :) = q(i-1, :) + dt * dq(i-1, :) + dt * dt * 0.25 * (ddq(i-1, :) + ddq(i, :));
    dq(i, :) = dq(i-1, :) + 0.5 * dt * (ddq(i-1, :) + ddq(i, :));
    ddq(i, :) = (inv([1, 0.5; 0.5, 0.5]) * (-[19.62, 0; 0, 9.81] * (q(i, :)) + [10; 5*sqrt(2)] * sin(2 * pi * i * dt)))';
end
plot(t, q(:, 1), 'b-', 'Linewidth', 1)
hold on
plot(t, q(:, 2), 'k-', 'Linewidth', 1)
legend('thetal', 'theta2')

```

2.6) Des valeurs sont dans le tableau au-dessous :

t	q(t)	$\dot{q}(t)$	$\ddot{q}(t)$
0	$\begin{pmatrix} 0 \\ 0 \end{pmatrix}$	$\begin{pmatrix} -1.3152 \\ -1.8600 \end{pmatrix}$	$\begin{pmatrix} 0 \\ 0 \end{pmatrix}$
Δt	$\begin{pmatrix} -0.0132 \\ -0.0186 \end{pmatrix}$	$\begin{pmatrix} -1.3152 \\ -1.8600 \end{pmatrix}$	$\begin{pmatrix} 0.8853 \\ 1.2521 \end{pmatrix}$
$2\Delta t$	$\begin{pmatrix} -0.0263 \\ -0.0372 \end{pmatrix}$	$\begin{pmatrix} -1.3108 \\ -1.8537 \end{pmatrix}$	$\begin{pmatrix} 1.3997 \\ 1.9795 \end{pmatrix}$
0.5	$\begin{pmatrix} -0.2671 \\ -0.3778 \end{pmatrix}$	$\begin{pmatrix} -0.6309 \\ -0.8923 \end{pmatrix}$	$\begin{pmatrix} 8.9163 \\ 12.6095 \end{pmatrix}$

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

1. Résolution avec un schéma de NEWMARK explicite

1.1) Les relations sont :

$$q_{j+1} = q_j + \Delta t * \dot{q}_j + 0.5 * \Delta t^2 * \ddot{q}_j$$

$$\dot{q}_{j+1} = \dot{q}_j + 0.5 * \Delta t * (\ddot{q}_j + \ddot{q}_{j+1})$$

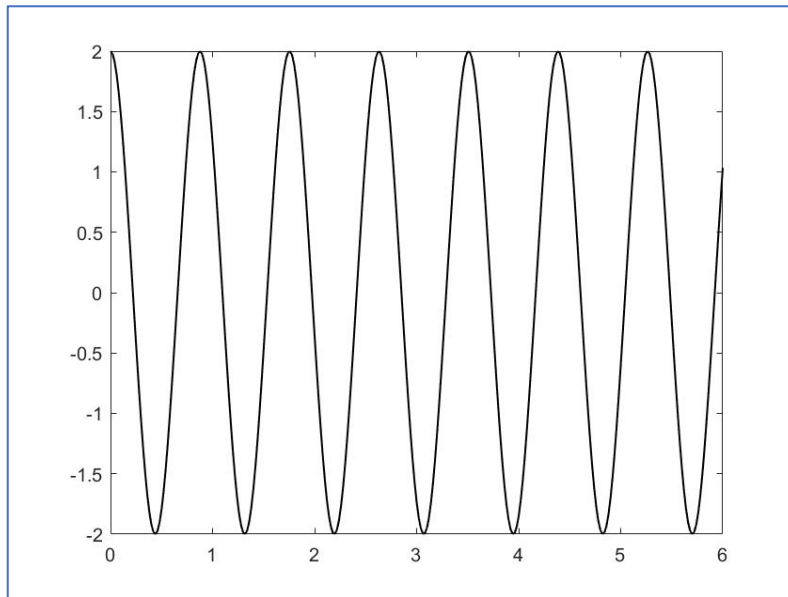
$$\ddot{q}_{j+1} = -\omega_0^2 * q_{j+1} * (1 + a * q_{j+1}^2)$$

1.2) Programmation

Code :

```
dt = 0.02;
T = 6;
q0 = 2;
dq0 = 0;
t = (0:dt:T)';
np = size(t,1);
q = zeros(np,1);
dq = zeros(np,1);
%energ = zeros(np,1);
w = 2 * pi;
q(1) = q0;
dq(1) = dq0;
ddq0 = - w * w * q(1) * (1 + 0.1 * q(1) * q(1));
for i = 2:np
    q(i) = q(i-1) + dt * dq(i-1) + dt * dt * 0.5 * ddq0;
    ddq = - w * w * q(i) * (1 + 0.1 * q(i) * q(i));
    dq(i) = dq(i-1) + 0.5 * dt * (ddq0 + ddq);
    ddq0 = ddq;
end
%energ = 0.5 * (dq.* dq) - w * cos(q);
plot(t,q,'k-','Linewidth',1)
hold on
```

Résultat :



1.3) $q(0) = 2$

$$q(\Delta t) = q(0.02) = 1.9779$$

$$q(2\Delta t) = q(0.04) = 1.9123$$

$$q(T) = q(6) = 1.0329$$

2 Résolution avec un schéma de NEWMARK explicite

2.1) On doit chercher à minimiser la correction

2.2) L'expression analytique de la correction :

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

2.3) Programmation

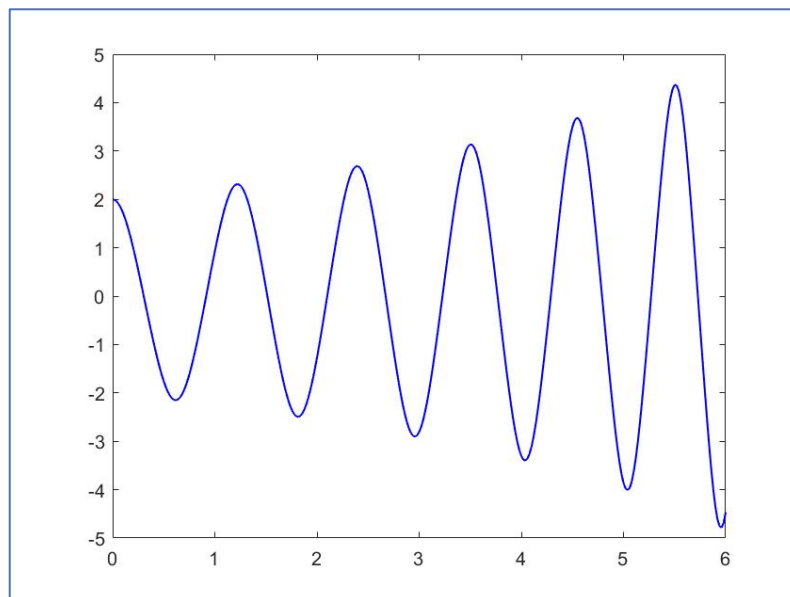
Code :

```

dt = 0.02;
T = 6;
q0 = 2;
dq0 = 0;
t = (0:dt:T)';
np = size(t,1);
q = zeros(np,1);
dq = zeros(np,1);
ddq = zeros(np,1);
%energ = zeros(np,1);
w = 2 * pi;
q(1) = q0;
dq(1) = dq0;
ddq(1) = - w * w * q(1) * (1+0.1*q(1)*q(1));
for i = 2:np
    q(i) = q(i-1) + dt * dq(i-1) + dt * dt * 0.25 * (ddq(i-1) + ddq(i));
    dq(i) = dq(i-1) + 0.5 * dt * (ddq(i-1) + ddq(i));
    ddq(i) = - w * w * q(i) * (1 + 0.1 * q(i) * q(i));
end
%energ = 0.5 * (dq.* dq) - w * cos(q);
plot(t,q,'b|', 'Linewidth',1)

```

Résultat :



2.4) $q(0) = 2$

$$q(\Delta t) = q(0.02) = 1.9889$$

$$q(2\Delta t) = q(0.04) = 1.9559$$

$$q(T) = q(6) = -4.4688$$

3 Énergie mécanique

3.1) L'énergie mécanique pour cet oscillateur non linéaire est

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

3.2) Programmation

Newmark Explicite

Code :

```
dt = 0.02;
T = 6;
q0 = 2;
dq0 = 0;
t = (0:dt:T)';
np = size(t,1);
q = zeros(np,1);
dq = zeros(np,1);
energ = zeros(np,1);
w = 2 * pi;
q(1) = q0;
dq(1) = dq0;
ddq0 = - w * w * q(1) * (1 + 0.1 * q(1) * q(1));
for i = 2:np
    q(i) = q(i-1) + dt * dq(i-1) + dt * dt * 0.5 * ddq0;
    ddq = - w * w * q(i) * (1 + 0.1 * q(i) * q(i));
    dq(i) = dq(i-1) + 0.5 * dt * (ddq0 + ddq);
    ddq0 = ddq;
end
energ = 0.5 * (dq.* dq) - 0.5 * w * w * q .* q;
plot(t, energ, 'b-', 'Linewidth', 1)
```

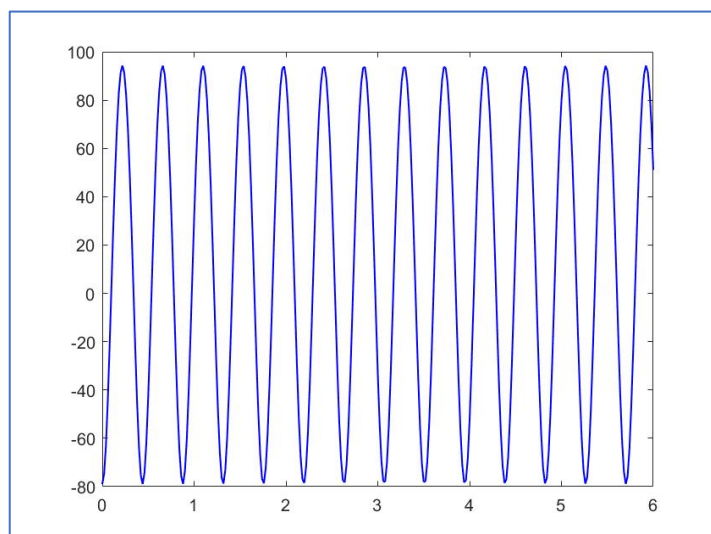
Newmark Implicite

Code :

```
dt = 0.02;  
T = 6;  
q0 = 2;  
dq0 = 0;  
t = (0:dt:T)';  
np = size(t,1);  
q = zeros(np,1);  
dq = zeros(np,1);  
ddq = zeros(np,1);  
energ = zeros(np,1);  
w = 2 * pi;  
q(1) = q0;  
dq(1) = dq0;  
ddq(1) = - w * w * q(1) * (1+0.1*q(1)*q(1));  
for i = 2:np  
    q(i) = q(i-1) + dt * dq(i-1) + dt * dt * 0.25 * (ddq(i-1) + ddq(i));  
    dq(i) = dq(i-1) + 0.5 * dt * (ddq(i-1) + ddq(i));  
    ddq(i) = - w * w * q(i) * (1 + 0.1 * q(i) * q(i));  
end  
energ = 0.5 * (dq.* dq) - 0.5 * w * w * q .* q;  
plot(t, energ, 'b-', 'Linewidth', 1)
```

3.3) Résultat quand $\Delta t = 0.02s$

Résultat de Newmark Explicite



Résultat de Newmark Implicite

