TD-pendule_simple2

April 4, 2022

[1]:
```
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import odeint
```

1 Pendule simple amorti

[61]:
```
l,g,f,m=1.8,9.8,0.1,0.5
w0=np.sqrt(g/l)
T0=2*np.pi/w0
def derivee(inconnues,t):
    theta=inconnues[0]
    dtheta=inconnues[1]
    ddtheta=-f*l/m*np.abs(dtheta)*dtheta-g/l*np.sin(theta)
    return [dtheta,ddtheta]
ci=[40*np.pi/180,0]
tab_t=np.linspace(0,10*T0,1000)
sol=odeint(derivee,ci,tab_t)
tab_theta=sol[:,0]
tab_dtheta=sol[:,1]

plt.figure(figsize=(8,6))
plt.plot(tab_t,tab_theta)
plt.xlabel("t(s)")
plt.ylabel("theta(°)")
plt.show()
plt.figure(figsize=(8,6))
plt.plot(tab_theta,tab_dtheta)
plt.xlabel("theta(°)")
plt.ylabel("thetapoint(°/s)")
plt.show()
```

[63]:
```
ci=[0,1000*np.pi/180]
tab_t=np.linspace(0,10*T0,1000)
sol=odeint(derivee,ci,tab_t)
tab_theta2=sol[:,0]
tab_dtheta2=sol[:,1]
```

```
plt.figure(figsize=(8,6))
plt.plot(tab_t,tab_theta2)
plt.xlabel("t(s)")
plt.ylabel("theta(°)")
plt.show()
plt.figure(figsize=(8,6))
plt.plot(tab_theta,tab_dtheta)
plt.plot(tab_theta2,tab_dtheta2)
plt.xlabel("theta(°)")
plt.ylabel("thetapoint(°/s)")
plt.show()
```

2 Pendule simple secoué

[44]:
```
Z0=5e-2
f=0.1
def derivee(inconnues,t):
    theta=inconnues[0]
    dtheta=inconnues[1]
    ddtheta=-f*l/m*np.abs(dtheta)*dtheta-g/l*np.sin(theta)+Z0*w**2/l*np.
 ↪sin(theta)*np.cos(w*t)
    return [dtheta,ddtheta]
ci=[1*np.pi/180,0]
tab_t=np.linspace(0,50*T0,1000)
w=w0
sol=odeint(derivee,ci,tab_t)
tab_theta_w_w0=sol[:,0]
w=2*w0
sol=odeint(derivee,ci,tab_t)
tab_theta_w_2w0=sol[:,0]
w=3*w0
sol=odeint(derivee,ci,tab_t)
tab_theta_w_3w0=sol[:,0]

plt.figure(figsize=(8,6))
plt.plot(tab_t,tab_theta_w_w0,label="w=w0")
plt.plot(tab_t,tab_theta_w_2w0,label="w=2 w0")
plt.plot(tab_t,tab_theta_w_3w0,label="w=3 w0")
plt.xlabel("t(s)")
plt.ylabel("theta(°)")
plt.legend()
plt.show()
```

[50]:
```
def amplitude(w):
    Z0=5e-2
```

```
    f=0.1
    def derivee(inconnues,t):
        theta=inconnues[0]
        dtheta=inconnues[1]
        ddtheta=-f*l/m*np.abs(dtheta)*dtheta-g/l*np.sin(theta)+Z0*w**2/l*np.
 ↪sin(theta)*np.cos(w*t)
        return [dtheta,ddtheta]
    ci=[1*np.pi/180,0]
    tab_t=np.linspace(0,50*T0,1000)
    sol=odeint(derivee,ci,tab_t)
    tab_theta=sol[:,0]
    return max(tab_theta)
tab_w=np.linspace(1.8,2.2,50)
tab_ampl=np.zeros_like(tab_w)
for i in range(len(tab_w)):
    tab_ampl[i]=amplitude(w0*tab_w[i])

plt.figure(figsize=(8,6))
plt.plot(tab_w,tab_ampl)
plt.xlabel("w/w0")
plt.ylabel("theta_max(°)")
plt.axhline(max(tab_ampl)/np.sqrt(2),ls="--")
plt.show()
```

Largeur de BP: $2.040 - 1.963 = 0,077$ donc $Q = 26$.

Si maintenant on secoue horizontalement, la résonance est au voisinage de ω_0:

[56]:
```
def amplitudeh(w):
    Z0=5e-2
    f=0.1
    def derivee(inconnues,t):
        theta=inconnues[0]
        dtheta=inconnues[1]
        ddtheta=-f*l/m*np.abs(dtheta)*dtheta-g/l*np.sin(theta)+Z0*w**2/l*np.
 ↪cos(theta)*np.cos(w*t)
        return [dtheta,ddtheta]
    ci=[1*np.pi/180,0]
    tab_t=np.linspace(0,50*T0,1000)
    sol=odeint(derivee,ci,tab_t)
    tab_theta=sol[:,0]
    return max(tab_theta)
tab_w=np.linspace(0.8,1.2,50)
tab_amplh=np.zeros_like(tab_w)
for i in range(len(tab_w)):
    tab_amplh[i]=amplitudeh(w0*tab_w[i])

plt.figure(figsize=(8,6))
```

```
plt.plot(tab_w,tab_amplh)
plt.xlabel("w/w0")
plt.ylabel("theta_max(°)")
plt.axhline(max(tab_amplh)/np.sqrt(2),ls="--")
plt.show()
```

[ ]:

```

```