

```

# -*- coding: utf-8 -*-
"""
Created on Fri Sep  4 10:14:40 2020

@author: Fabrice
"""

##résolution de l'équadiff x''=-omega0**2*x-2*lambda*x'  oscillateur amorti
import numpy as np
from scipy.integrate import odeint
import matplotlib.pyplot as plt

#définition zone graphique
fig=plt.figure(figsize=(10,8))
ax1, ax2 = fig.add_subplot(121) , fig.add_subplot(122)

##Oscillateur harmonique amorti (ressort horizontal avec masse oscillante dans
##un milieu avec frottement)
#grandeurs physiques m(masse), alpha (frottement), k (constante ressort)
m = 0.5
alpha = 1
k=1000
lambda=alpha/m
omega0=np.sqrt(k/m)

#conditions initiales en t=0 pour la fonction odeint position0 et vitesse0
position0, vitesse0 = 1 , 0
initial=[position0, vitesse0]

#base de temps
t = np.linspace(0, 2, 1001)

#paramètres dz/dt = [vitesse , accélération] pour la fonction odeint
def OHA(y, t, lambda, omega0):
    position , vitesse = y

    #fichier à 2D: vitesse et vitesse'=-wo**2-2*lambda*vitesse
    dydt=[vitesse , -(omega0**2)*position-2*lambda*vitesse]

    return dydt

#liste des coefficients lambda pour régime aperiodique, pseudo-périodique et
#critique
liste=[100, 45 , 5]

#boucle qui dessine les graphiques 1). position(t) et 2). vitesse(position)
for lambda in liste:

    #intégration
    sol = odeint(OHA , initial, t , args=(lambda,omega0))

    #position = sol[:,0] vitesse = sol[:,1]
    ax1.plot(t , sol[:,0], marker='o', markersize = 1,
             label='position x'+ ' lambda= '+str(lambda)+' omega= '+str(omega0))
    ax2.plot(sol[:,0] , sol[:,1], marker='o', markersize = 1 ,
             label='lambda= '+str(lambda)+' omega= '+str(omega0))

#paramètres des graphiques
ax1.grid(which='both')
ax1.set_xlabel('temps (s)')
ax1.set_ylabel('position (m)')
ax1.set_title('Oscillateur harmonique amorti, position')

ax2.grid(which='both')
ax2.set_xlabel('position (m)')
ax2.set_ylabel('vitesse (m/s)')
ax2.set_title('Oscillateur harmonique amorti, diagramme de phase')

#dessin des graphiques
fig.tight_layout()
plt.legend()
plt.show()

```