

```
import numpy as np
from PIL import Image
from pylab import *
import matplotlib.pyplot as plt

def tableau(i,j):
    return [[4.44*10**(-13) for a in range(j)] for k in range(i)],[[100 for a in
range(j)] for k in range(i)]
# 4.44*10**(-13) représente le temps de parcours de la case

def dimtab2D(t):
    return len(t), len(t[0])

def bulle(t1,t2): #fonction qui crée une bulle dans la cuve , modifie 2 tableaux un
pour le trajet, l'autre pour l'affichage
    i,j=dimtab2D(t1)
    r=randint(1,50) # rayon de la bulle
    (a,b)=(randint(0,i),randint(0,j)) # place de la bulle
    for p in range(-r,r+1):
        for w in range(-r,r+1):
            if floor(sqrt(p**2+w**2))<=r and -1<a-p<i and -1<b-w<j: #on regarde qu'on
est bien à une distance r du centre
                t1[a-p][b-w]=4.92*10**(-13) # on modifie la valeur du temps de parcours
                t2[a-p][b-w]=200

t1,t2=tableau(3000,5000)
for i in range(1000):
    bulle(t1,t2)

def somme(t,i):
    # somme la ième ligne de t
    s = 0
    for p in t[i]:
        s=s+p
    return s

def moyenne(L):
    s=0
    for i in L:
        s=s+i
    return(s/len(L))

def simulation(t):
    L=[]
    A=[]
    for i in range(len(t)):
        A=A+[i]
    A=np.array(A)
    A=len(t)-A
    print(A) # on définit le nombre de lignes sur lesquelles on va travailler
    for i in range(len(t)):
        L=L+[somme(t,i)/100] # on incorpore dans L la durée de parcours de chaque ligne
    print(moyenne(L))
    return (np.array(L)-moyenne(L))/1000,A

X,Y=simulation(t1)
plt.close()
plt.plot(X,Y)
plt.xlim(-5e-15,5e-15)
```

```
plt.ylabel('ligne de la cuve')
plt.xlabel('écart entre les temps de parcours')
plt.show()

a=np.array(t2)
image = Image.fromarray(a)
image.show()
```