

```

# -*- coding: utf-8 -*-
"""
Created on Mon Sep  2 08:32:51 2019

@author: YYX
"""

import numpy as np
import matplotlib.pyplot as plt
import random
#parameter
lamdb = 1 # step size is 1
#random.seed(151)
# (1)

N = 10000 # number of steps
Q = 100000 # number of trials
bins_D = 50
D = 2*np.linspace(10,200,bins_D)
n = np.zeros(bins_D) # number of trials which stays between the plates

# calculate Distance for a given steps
for q in range(Q):
    # calculate Distance for a given steps
    xx = 0
    yy = 0
    x = []
    y = []
    # y array for one polynom
    for i in range(N):
        theta = random.random()*2*np.pi
        xx += lamdb*np.cos(theta)
        yy += lamdb*np.sin(theta)
        x.append(xx)
        y.append(yy)
    for i_d,d in zip(range(len(D)),D):
        if max(abs(np.array(y))) < d/2:
            n[i_d] +=1
            ## show the plot
            if n[i_d]==1 and d in D[np.array([5])]:
                plt.figure(i_d+1)
                for NN in range(N-1):
                    dx = x[NN+1]-x[NN]; dy = y[NN+1]-y[NN]

                    plt.arrow(x[NN], y[NN], dx, dy,length_includes_head=True,
                               head_width=0.04, head_length=0.04)
                plt.plot(np.linspace(-d,d,1000),[d/2]*1000)
                plt.plot(np.linspace(-d,d,1000),[-d/2]*1000)
                plt.axis([-0.7*d,0.7*d, -0.6*d ,0.6*d])
                plt.savefig('%d.pdf'%i_d)

kT = 1
F= -kT*np.log(n)
np.savetxt('F.txt',F)
from scipy import interpolate
FF = interpolate.interp1d(D, F)

f = np.zeros(bins_D-1)
for i in range(bins_D-1):

```

```

    f[i] = -(FF(D[i]+0.1)-FF(D[i]))/0.1
np.savetxt('force.txt',f)

plt.figure(31)
plt.semilogx(D, F)
plt.xlabel('D'); plt.ylabel('F')
plt.savefig('FvsD.pdf')
plt.figure(32)
plt.loglog(D[:-1],f)
plt.xlabel('D'); plt.ylabel('f')
plt.savefig('forcevsD.pdf')

```