exo4-solution

October 16, 2023

[2]:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

"""
Import et traitement des données d'une dynamique moléculaire
"""

# Importation des librairies
import numpy_MD_utils
import numpy as np
import scipy
import scipy.constants as constants

# Definition des fonctions
def calc_EC(masses,vs):
    """
    Computes the kinetic energy based on :
    - masses : (ndarray) 1D array for the masses corresponding to each atome␣
 ↪along -2 axis of data
    - vs (ndarray) 4D array time,molecule,atome,position : speed for eache atom␣
 ↪(in m/s) the last dimension is supposed to contain x, y and z values of the␣
 ↪speed
    returns the kinetic energy as a 3D array time,molecule,atom,kinetic energy␣
 ↪(Joule)
    """
    return  1/2 * masses * np.sum(vs**2,axis=-1)
    """
    avec np.linalg.norm :
    return  1/2 * masses * np.linalg.norm(vs,axis=-1)**2
    'axis = -1' peut être remplacé par 'axis = 3'
    """

def calc_bary(masses,data):
    """
    Compute the barycentre of each molecule from their positions (data) and␣
 ↪masses
```

```
    masses : (ndarray) 1D array for the masses corresponding to each atome␣
 ↪along -2 axis of data
    vs : (ndarray) 4D array time,molecule,atome,position

    returns (ndarray) position of each barycentre as a 3D array␣
 ↪time,molecule,barycentre (x,y,z)
    """
    """
    Attention, souci de broadcasting : il faut multiplier chaque coordonnée par␣
 ↪la masse
    et surtout pas x par la masse de l'oxygène, y par la masse de l'hydrogène␣
 ↪et z par la masse de l'hydrogène
    """
    return np.sum(np.expand_dims(masses,-1)*data,axis=-2)/np.sum(masses)

# Programme principal
if __name__ == "__main__":
    data = numpy_MD_utils.extract_data('testFile.xyz')
    #conversion en mètre
    data = data*1e-10
    #shape du tableau importé
    print('\nQu3. shape')
    print(data.shape)
    print('\nQu4. coordonnées')
    print(data[1,1,1,:].shape)
    print(data[1,1,1,:]*1e10)
    #trajectoire du premier atome d'oxygène
    print('\nQu5. trajectoire')
    print(data[:,0,0,:].shape)
    print(data[:,0,0,:]*1e10)
    #pas de temps, remarquer la notation pour les fs
    dt = 0.5e-15
    #ATTENTION à préciser ET le pas de temps (dt) ET l'axe de calul (le temps)
    vs = np.gradient(data,dt,axis=0,edge_order=2)
    print('\nQu6. vitesse')
    print(vs.shape)
    print(vs[:,0,0,:])

    masses = np.array([16,1,1])*1e-3/constants.N_A
    #calcul de l'énergie cinétique :
    Ec_at = calc_EC(masses,vs)
    print('\nQu7. E_c')
    print(Ec_at.shape)
    print(Ec_at[:,0,0])

    print('\nQu8. bary')
    #Calcul de la position du barycentre de chaque molécule
    barys = calc_bary(masses,data)
    print(barys.shape)
    print('avec np.sum')
    print(barys[:,0,:])
    #avec la fonction average :
    barys = np.average(data,weights = masses, axis = 2)
    print('avec np.average')
    print(barys[:,0,:])

    print('\nQu9. E_c totale')
    Ec_tot = np.sum(Ec_at,axis=(1,2))
    print(Ec_tot.shape)
    print(Ec_tot)
    mean_Ec = np.mean(Ec_tot)
    print('moyenne temporelle de E_c : {:.3e} J, écart-type réduit : {:.3e} J'.format(mean_Ec,np.std(Ec_tot,ddof=1)) )
    Temp = 2/(3*constants.k)*mean_Ec/(data.shape[1]*data.shape[2])
    print('Température : {:.3f} K'.format(Temp))


    print('\nQu10. Coordonnées réduites')
    barys_average = np.mean(barys,axis=0,keepdims=True)
    """
    Ici, par défaut les coordonnées ne sont pas broadcastable car le calcul du barycentre a « enlevé »
    l'axe 2 (O,H,H)
    l'option keepdims = True ci-dessus permet de garder la dimension liée au temps mais il faut
    rajouter une dimension avec expand_dims
    """
    print(np.expand_dims(barys_average,2).shape)
    coordinates_reduced = data - np.expand_dims(barys_average,2)
    """
    alternative :
    barys_average = np.mean(barys,axis=0)
    coordinates_reduced = data - np.expand_dims(barys_average,(0,2))
    """
    print(coordinates_reduced[:,0,0,:]*1e10)
```

```
Qu3. shape
(10, 64, 3, 3)

Qu4. coordonnées
(3,)
[-3.34152225  0.82586296  4.15455712]
```

```
Qu5. trajectoire
(10, 3)
[[ 3.85858738 -4.75331752  5.50919748]
 [ 3.86207789 -4.75647898  5.50987972]
 [ 3.86670402 -4.76021707  5.51020989]
 [ 3.87165408 -4.76422854  5.51167051]
 [ 3.87814674 -4.76788065  5.51245274]
 [ 3.88368732 -4.77156524  5.51394204]
 [ 3.88807613 -4.77544192  5.51568638]
 [ 3.89273159 -4.77678148  5.51657667]
 [ 3.89688626 -4.77724275  5.51936164]
 [ 3.90072985 -4.7777787   5.52115613]]

Qu6. vitesse
(10, 64, 3, 3)
[[ 584.53991 -574.62943  171.65707]
 [ 811.66397 -689.95485  101.24125]
 [ 957.61961 -774.95612  179.07878]
 [1144.27258 -766.35776  224.28546]
 [1203.32347 -733.67012  227.15323]
 [ 992.93892 -756.12735  323.36409]
 [ 904.4278  -521.62392  263.46327]
 [ 881.01316 -180.0825   367.52526]
 [ 799.82575  -99.72119  457.94512]
 [ 737.61013 -114.65865  259.85056]]

Qu7. E_c
(10, 64, 3)
[9.31697967e-21 1.52116788e-20 2.05861879e-20 2.58641304e-20
 2.70715400e-20 2.20814580e-20 1.54030814e-20 1.25362558e-20
 1.14162629e-20 8.29920984e-21]

Qu8. bary
(10, 64, 3)
avec np.sum
[[ 3.83572787e-10 -4.77373429e-10  5.55890074e-10]
 [ 3.83891259e-10 -4.77672251e-10  5.55896508e-10]
 [ 3.84284552e-10 -4.78002300e-10  5.55907874e-10]
 [ 3.84722315e-10 -4.78288804e-10  5.56035633e-10]
 [ 3.85272551e-10 -4.78530403e-10  5.56119206e-10]
 [ 3.85757044e-10 -4.78809546e-10  5.56251688e-10]
 [ 3.86161465e-10 -4.79095799e-10  5.56385251e-10]
 [ 3.86638120e-10 -4.79233971e-10  5.56484026e-10]
 [ 3.87102070e-10 -4.79354019e-10  5.56796340e-10]
 [ 3.87527274e-10 -4.79472438e-10  5.57060592e-10]]
avec np.average
[[ 3.83572787e-10 -4.77373429e-10  5.55890074e-10]
```

```
 [ 3.83891259e-10 -4.77672251e-10  5.55896508e-10]
 [ 3.84284552e-10 -4.78002300e-10  5.55907874e-10]
 [ 3.84722315e-10 -4.78288804e-10  5.56035633e-10]
 [ 3.85272551e-10 -4.78530403e-10  5.56119206e-10]
 [ 3.85757044e-10 -4.78809546e-10  5.56251688e-10]
 [ 3.86161465e-10 -4.79095799e-10  5.56385251e-10]
 [ 3.86638120e-10 -4.79233971e-10  5.56484026e-10]
 [ 3.87102070e-10 -4.79354019e-10  5.56796340e-10]
 [ 3.87527274e-10 -4.79472438e-10  5.57060592e-10]]

Qu9. E_c totale
(10,)
[1.42433866e-18 1.17235456e-18 1.16635993e-18 1.17455483e-18
 1.21474045e-18 1.22383857e-18 1.20993172e-18 1.15322206e-18
 1.12301717e-18 1.41728112e-18]
moyenne temporelle de E_c : 1.228e-18 J, écart-type réduit : 1.061e-19 J
Température : 308.823 K

Qu10. Coordonnées réduites
(1, 64, 1, 3)
[[ 0.00365794  0.03251544 -0.05362972]
 [ 0.00714845  0.02935398 -0.05294747]
 [ 0.01177458  0.02561589 -0.0526173 ]
 [ 0.01672465  0.02160442 -0.05115668]
 [ 0.02321731  0.01795231 -0.05037445]
 [ 0.02875788  0.01426771 -0.04888515]
 [ 0.03314669  0.01039104 -0.04714081]
 [ 0.03780216  0.00905148 -0.04625052]
 [ 0.04195683  0.00859021 -0.04346555]
 [ 0.04580042  0.00805426 -0.04167107]]
```

```
[ ]:
```

```
[ ]:
```