Rakotomavo
Miara
M2BI

Paris Diderot Paris 7
2019 - 2020

Sujet: **Environnement & surface accessible au solvant d'une protéine**

I – INTRODUCTION :

Les protéines sont des macromolécules biologiques présentes dans toutes les cellules vivantes. Elles sont formées d'une ou plusieurs chaînes polypeptidiques. Chacune de ces chaînes est constituée de l'enchaînement de résidus d'acides aminés liés entre eux par des liaisons peptidiques.

1.1 Environnement d'une protéine :

Chaque Protéine possède son environnement qui lui est propre dans une cellule. Il est important d'analyser la surface moléculaire d'une protéine ainsi que l'environnement des groupes réactifs car ces derniers sont très fortement corrélés avec les propriétés chimiques de la protéine en question.
Plusieurs interactions non covalentes ont lieu entre ces divers groupes réactifs, mais aussi entre ces groupes et les molécules du solvant.

1.2 Repliement d'une protéine :

Le niveau de base de la structure des protéines, appelé «structure primaire», est la séquence linéaire des acides aminés. Toutefois, une protéine ne garde jamais une forme strictement linéaire. L'énergie contenu dans les liaisons hydrogène, les ponts disulfures, l'attraction entre les charges positives et négatives, et les radicaux hydrophobes ou hydrophiles , imposent à la protéine une structure secondaire en hélice alpha ou en feuillet bêta. Les molécules deviennent ensuite encore plus compactes en adoptant une nouvelle structure: La structure Tertiaire ou encore la phase de repliement des protéines (Protein Folding).

1.3 Rôle du Solvant :

Le solvant naturel (dans la cellule) des protéines est l'eau , qui joue un rôle capital dans le repliement et la stabilisation des états conformationnels intermédiaires. Le mécanisme de repliement s'accompagne d'une importante augmentation de l'entropie de l'eau, qui compense la diminution de celle de la chaîne polypeptidique. Dans l'état natif d'une protéine , ce sont exclusivement les acides aminés situés en surface de la protéine qui sont au contact du solvant. Analyser cette surface de la protéine exposé au solvant nous permettrait d'acquérir plus d'information au niveau du mode de repliement de la protéine.

1.4 Objectif du projet :

L'objectif de ce projet est donc de créer un programme permettant de calculer la surface accessible au solvant (absolue et relative) à partir des coordonnées d'une protéine issue d'un fichier pdb.

II – MATÉRIELS ET MÉTHODES :

2.1 Matériels :

2.1.1 Python

Cette application a été codé en python3 Python Core Team (2019) et lancé sous un environnement Linux. Python est un langage de programmation interprété et multi-plateformes.

2.1.2 Librairies non-standard pour python

Pour obtenir les fonctionnalités additionnelles, nous avons utilisé un ensemble de bibliothèques externes spécialisées sur l'analyse de données à grande échelle.
Pandas :
Pandas est une bibliothèque écrite pour le langage de programmation Python permettant la manipulation et l'analyse des données. Elle propose en particulier des structures de données et des opérations de manipulation de tableaux numériques et de séries temporelles, qui seront nécessaire pour lire nos données sous forme de séries de données.
NumPy :
Numpy est une extension du langage de programmation Python, destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux.

2.1.3 Jeu de donnée

Le programme nécessite un seul fichier d'entrée :
Potein 3D coordinate data file : un fichier pdb contenant les coordonnées en 3D des atomes de tous les résidus de la protéine. Dans ce projet , le programme a été lancé avec 2 fichiers pdb téléchargés sur le site http://www.rcsb.org :
- 3i40.pdb : (human insulin crystallized from a solution containing polysialic acid).
- CD59_2J8B.pdb : (glycoprotein that protects host cells from lysis by inhibiting the terminal pathway of complement),

2.1.4 Github

Tous les codes de ce logiciel ont été déposés sur le répertoire github dont le lien est :
https://github.com/Miara1502/Protein_Exposure.git

2.2 Méthodes :

Pour commencer , le programme va lire le fichier .pdb et extraire toutes les coordonnées 3D de chaque atome pour ensuite les insérer dans un Pandas DataFrame. Pour chaque atome , le programme va ensuite générer un nuage de point sous forme de sphère grâce à la fonction «**golden_sphere»,** qui sera ensuite translatée en fonction des coordonnées de l'atome. La fonction « **distance_all_atom»** permettra ensuite de calculer la distance entre chaque point contenu dans la sphère et tous les atomes de la protéine. Ensuite, la surface de l'atome exposée au solvant sera donnée par les fonction « **Exposition »** et « **Surface** ». **(cf figure 1 ci dessous)**

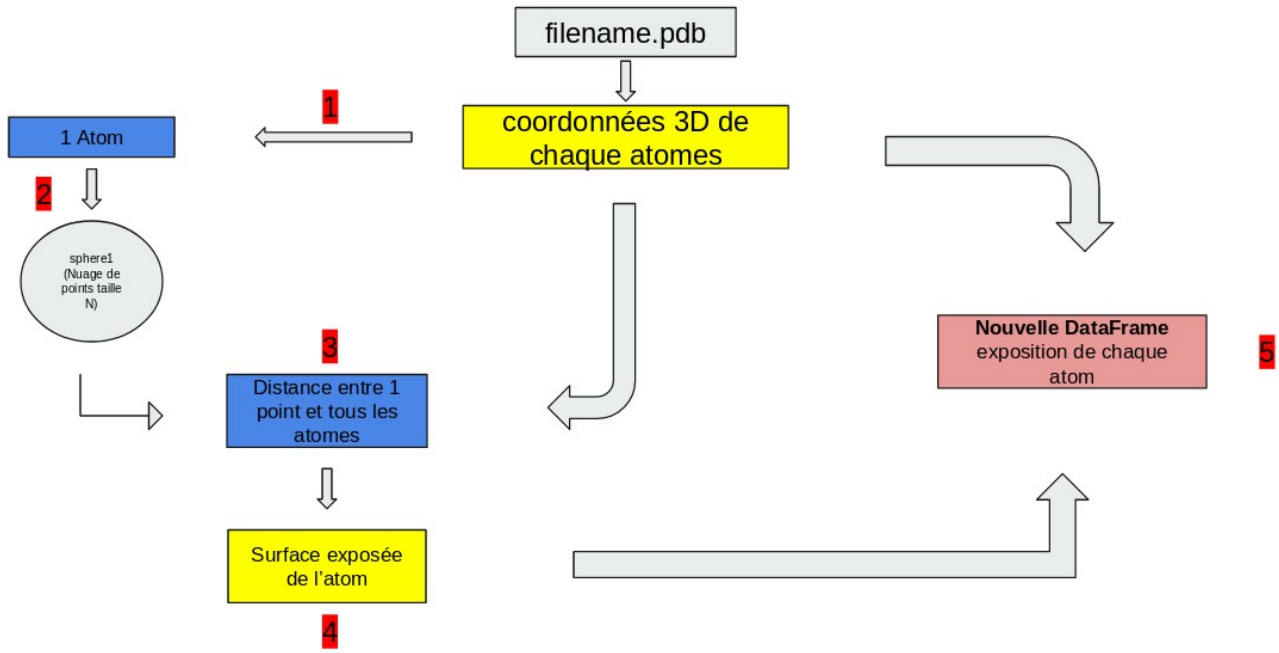


figure 1 : Description du programme

2.2.1 / 2.2.2 Extraction et sélection d'atome

Le programme va extraire les coordonnées des atomes du fichier pdb et les mettre dans un data frame grâce à la fonction **exctraction_coord()** qui va prendre comme argument un fichier.pdb. Ensuite, un atome va être sélectionner pour translater une sphère généré par la fonction **golden_shere()** et **translocation() .**

2.2.3 Calcul de distance entre une sphère et tous les atomes

Grâce à la fonction **distance_all_atom()** , qui regroupe les fonctions **calcule_distance()** et **distance_all_point()** , le programme va créer une liste de distance entre une sphère, qui correspond à un atome choisi, et tous les autres atomes de la protéine issu de la pdb.

2.2.4 Calcul de la surface exposée de l'atome

La fonction **Exposition()** va calculer d'abord la surface de la sphère (atome), puis elle va calculer le ratio des distances générées qui sont supérieur au diamètre du solvant + le rayon de l'atome pour finalement faire le produit de ce ratio et de la surface pour déterminer la surface exposée au solvant de la sphère ou de l'atome.

2.2.5 Ajout des surfaces / Surfaces exposées au solvant dans la Data Frame de base

La fonction **Exposition_all()** va ensuite répéter le même processus pour chaque atome dans le Data Frame issu du fichier .pdb, et ensuite créer 2 nouvelles colonnes: la surface de l'atome et la surface exposée au solvant.

III – RÉSULTATS :

3.1 Résultats expérimentaux :

Les résultats suivants ont été obtenu en utilisant le fichier 3i40.pdb, contenant les coordonnées des atomes de l'insuline chez l'homme. Après avoir lancé le programme avec les lignes de commande suivante (même protocole mais avec des valeurs N différentes) :

$time python3 src/main.py -p data/3i40.pdb -n 10 -d 6 > results/res_3i40_10point.txt
$time python3 src/main.py -p data/3i40.pdb -n 100 -d 6 > results/res_3i40_100point.txt

avec : n = 10 et 100 : nuages de point pour la sphère
d = 6 dans les deux cas : limite de distance

On obtient les résultats suivants :

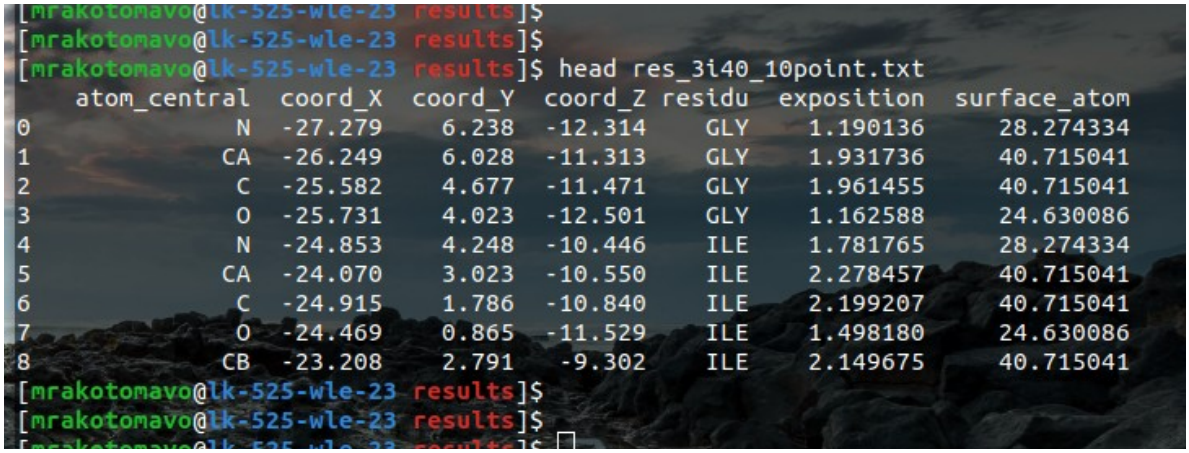
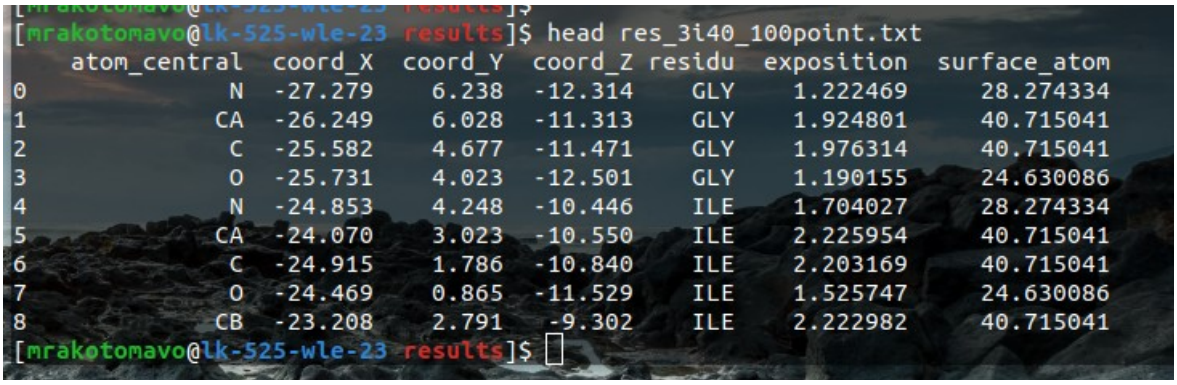


figure 2 : Résultat avec le fichier 3i40.pdb et avec 10 nuages de poins



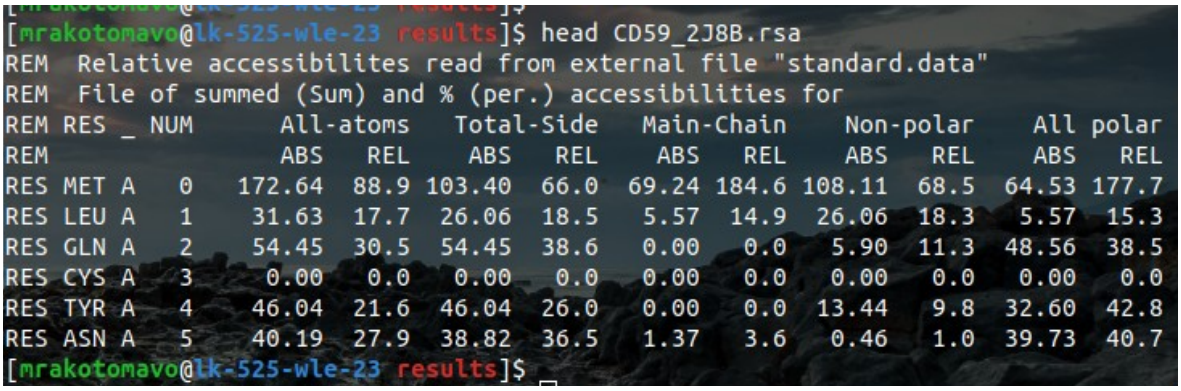
```
[mrakotomavo@lk-525-wle-23 results]$ head res_3i40_100point.txt
    atom_central  coord_X  coord_Y  coord_Z residu  exposition  surface_atom
0              N  -27.279    6.238  -12.314    GLY    1.222469     28.274334
1             CA  -26.249    6.028  -11.313    GLY    1.924801     40.715041
2              C  -25.582    4.677  -11.471    GLY    1.976314     40.715041
3              O  -25.731    4.023  -12.501    GLY    1.190155     24.630086
4              N  -24.853    4.248  -10.446    ILE    1.704027     28.274334
5             CA  -24.070    3.023  -10.550    ILE    2.225954     40.715041
6              C  -24.915    1.786  -10.840    ILE    2.203169     40.715041
7              O  -24.469    0.865  -11.529    ILE    1.525747     24.630086
8             CB  -23.208    2.791   -9.302    ILE    2.222982     40.715041
[mrakotomavo@lk-525-wle-23 results]$
```

figure 3 : Résultat avec le fichier 3i40.pdb et avec 10 nuages de poins

3.1.1 Comparaison des deux résultats obtenus

D'après les figures 2 et 3, le programme génère un Pandas Data Frame de dimensions (n : nombre d'atome , colonnes = 7) . On peut donc visualiser la surface exposée au solvant de chaque atome dans la colonne['Exposition'] . La suite du programme serait de calculer la somme des surfaces exposées des atomes appartenant à un même résidu puis on aurait pu comparer avec les résultats générés par NACESS. Malheureusement, je n'ai pas eu le temps de l'implémenter.
Mais ce qu'on pourrait rajouter grâce à ses résultats c'est qu'on augmentant les nuages de points (10 à 100 par exemple), on obtiendra des résultats plus significatifs.

3.1.2 Résultats NACESS



```
[mrakotomavo@lk-525-wle-23 results]$ head CD59_2J8B.rsa
REM  Relative accessibilites read from external file "standard.data"
REM  File of summed (Sum) and % (per.) accessibilities for
REM RES _ NUM      All-atoms   Total-Side   Main-Chain    Non-polar    All polar
REM                 ABS   REL    ABS   REL    ABS   REL    ABS   REL    ABS   REL
RES MET A   0   172.64  88.9 103.40  66.0  69.24 184.6 108.11  68.5  64.53 177.7
RES LEU A   1    31.63  17.7  26.06  18.5   5.57  14.9  26.06  18.3   5.57  15.3
RES GLN A   2    54.45  30.5  54.45  38.6   0.00   0.0   5.90  11.3  48.56  38.5
RES CYS A   3     0.00   0.0   0.00   0.0   0.00   0.0   0.00   0.0   0.00   0.0
RES TYR A   4    46.04  21.6  46.04  26.0   0.00   0.0  13.44   9.8  32.60  42.8
RES ASN A   5    40.19  27.9  38.82  36.5   1.37   3.6   0.46   1.0  39.73  40.7
[mrakotomavo@lk-525-wle-23 results]$
```

figure 4 : Résultat avec le fichier CD59_2J8B.rsa

IV – Conclusion & Discussion :

4.1 Problèmes du code :

4.1.1 Problème d'optimisation

Pour conclure, le programme semble marcher mais présente pas mal de problèmes notamment au niveau du temps de calcul. En effet , du à un problème d'optimisation du code, il prend beaucoup de temps pour générer des résultats. Par exemple , pour une protéine qui contient 413 atomes , si on fixe n = 10 , c'est à dire que chaque sphère possède 10 points , le programme durera aux alentours de 6 à 7 mn. Par contre si on genère 100 points par sphère pour le même fichier pdb, le programme peut prendre jusqu'à 17 mn pour générer des résultats.

4.1.2 résultat très approximatif :

Le programme calcule toutes les distances entre chaque point appartenant à une sphère et tous les autres atomes, or à partir d'une certaine distance , elle sera insignifiante puisque elle sera toujours supérieure au (diam(eau)x2 + rayon(atom)). D'où le problème de temps.

4.2 Solutions :

Pour éviter de calculer toutes les distances, la solution serait de générer un cube autour de chaque sphère, qui permettra ainsi de limiter les calculs de distances au niveau des atomes voisins seulement.

Pour pouvoir comparer les résultats avec Nacess , il faudra faire la somme des surfaces exposé pour chaque atome contenu dans un résidu. Par exemple, pour un résidu Glycine , il faudra calculer la somme des surfaces exposées au solvant pour chaque atome qui se trouve dans le résidu Glycine. Ainsi, on pourrait créer un dictionnaire ayant comme clé , le nom des résidu de la protéine et comme valeur, une liste de valeur de surface exposée au solvant. Finalement, il suffira de faire la somme pour chaque liste et comparer les résultats avec le fichier générer par NACESS.

V – REFERENCES :

1) Saff-Kuijlaars , (1997) Distributing Many PointsOn A Spher.pdf
2) A.Shrake and J. A. Rupley, J. Mol. Biol., 79 , 351 (1973) . Environment and exposure to solvent of proteins Atoms—Lysozyme and Insulin
3)https://www.mathcurve.com/courbes2d.gb/logarithmic/http://biochimej.univ-angers.fr/Page2/COURS/7RelStructFonction/4Repliement/1Introduction.htmspiraledor.shtml
4) http://chemphys.u-strasbg.fr/mpb/teach/proteines/proteines.html
5) http://iramis.cea.fr/ComScience/Phases/phases_23/p23article1.html
6) http://biochimej.univ-angers.fr/Page2/COURS/7RelStructFonction/4Repliement/1Introduction.htm