Université
Paris Cité

Project n°2:

Calculating the solvent-accessible surface area of a protein

By Bel Alexis

Master 2 biologic-informatique

Year 2022/2023

MASTER 2 Biologie
Informatique

1. Introduction:

With the use of structural analysis, we can obtain the position of each atom in a 3D space. This positions can be use to make models of the protein. But it can also be use to calculate others properties of the protein. One of the properties is the accessible surface area (ASA). This ASA is a surface in angstroms**2 of the protein which is exposed to the solvent (water). With this ASA, we can make better prediction of the secondary structure of the protein[2].

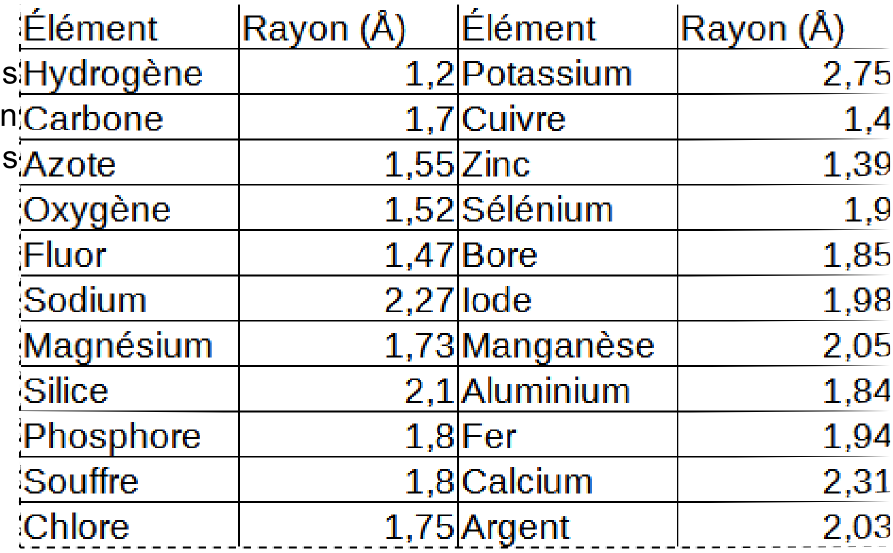
2. Material and method

a. Protein and variables:

For this project, we used 4 protein 1bja, 1rmd1 1zmh and 7mme which are available on the website protein data bank. We used also 92 points by atoms and a radius for the probe of 1,4 A [4].

b. Radius atom:

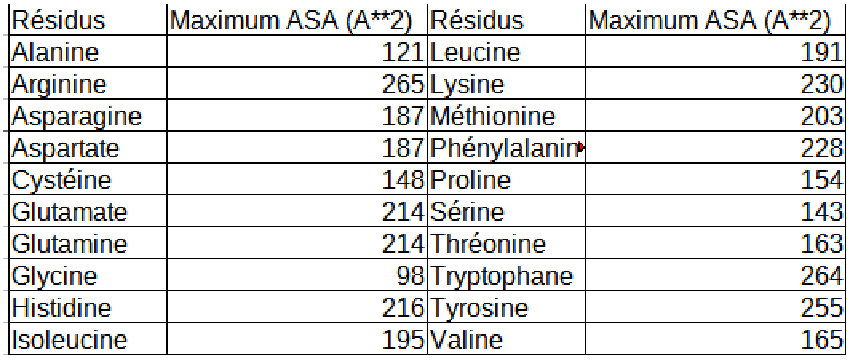
We used the radius of Van der Walls from the publications of Bondi [1]. In the table 2, you can see the radius used.



Élément	Rayon (Å)	Élément	Rayon (Å)
Hydrogène	1,2	Potassium	2,75
Carbone	1,7	Cuivre	1,4
Azote	1,55	Zinc	1,39
Oxygène	1,52	Sélénium	1,9
Fluor	1,47	Bore	1,85
Sodium	2,27	Iode	1,98
Magnésium	1,73	Manganèse	2,05
Silice	2,1	Aluminium	1,84
Phosphore	1,8	Fer	1,94
Souffre	1,8	Calcium	2,31
Chlore	1,75	Argent	2,03

Table 1 : Radius of Van der Walls used in the project

c. Max ASA protein:



Résidus	Maximum ASA (A**2)	Résidus	Maximum ASA (A**2)
Alanine	121	Leucine	191
Arginine	265	Lysine	230
Asparagine	187	Méthionine	203
Aspartate	187	Phénylalanin	228
Cystéine	148	Proline	154
Glutamate	214	Sérine	143
Glutamine	214	Thréonine	163
Glycine	98	Tryptophane	264
Histidine	216	Tyrosine	255
Isoleucine	195	Valine	165

Table 2 : Maximal accessible surface area for each residue[5]

To calculate the relative accessible surface area of the residues, we used the empirical values of the publications from Tien and al.

The formula is RASA = ASA / Max ASA

d. Algorithm de Saff:

The Algorithm of Saff was used to models the position of each points on the atom surface. We obtain the spherical coordinates of each points. By using this coordinates θ and ϕ, we can place the points on the surface by following a spiral like you see in the figure 1. With:

$0 \leq \theta \leq pi$ et $0 \leq \phi \leq 2*pi$

$2 \leq k \leq N-1$ et $\phi(1) = \phi(N) = 0$

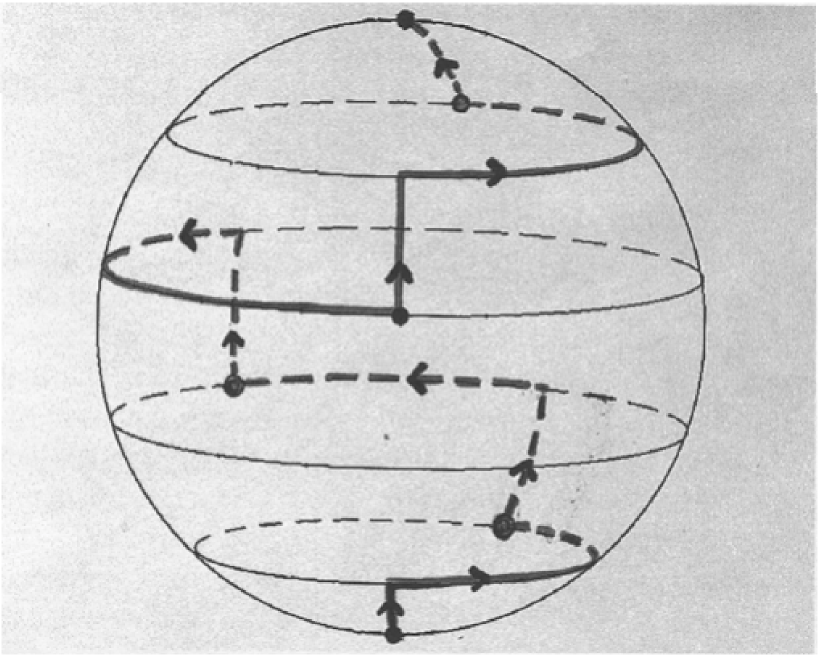


Figure 1 : Construct of points in spiral. With N = 6 [3]

e. Algorithm de Shrake rupley :

The Algorithm de Shrake rupley will calculate the exposed ASA of each atom. To work, it needs 3 data: the coordinates (x, y, z) of the points of an atom A, the position of the center of an atom B and the radius of the atom B + the radius of the probe. By comparing the distance between the distance of the points of the surface of atom A and the center of the atom B and the radius of atom B + the radius of the probe, we can determine if a point is buried or not.

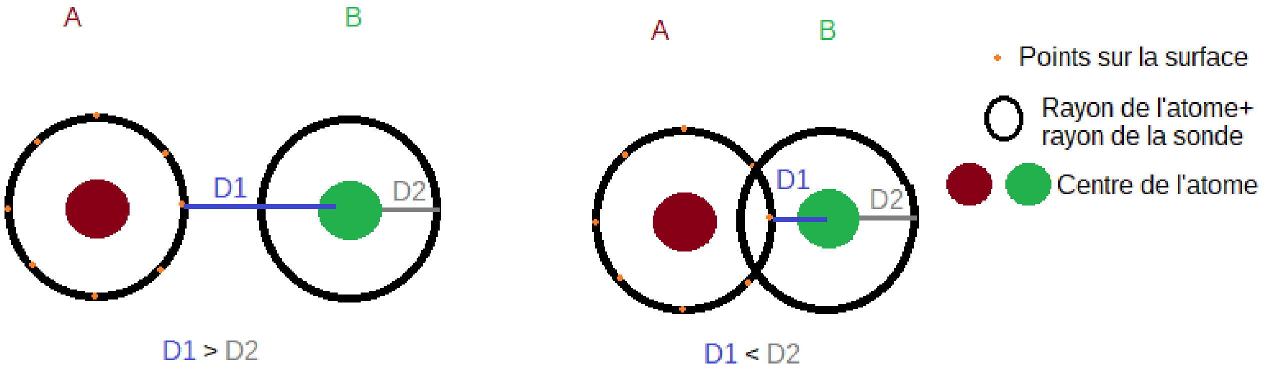


Figure 2: Sketch showing the differences between a buried and a free points

You can see in figure 2 when the distance between one of the points of A and the center of B is greater with the radius of B + probe, then the points is free. But in the reverse case the point is buried.

By comparing each points of each atoms, we can obtain the accessible surface area of each residue and of the protein.

3. Result:

a. ASA

We then compare the results of our script with the result obtain by dssp . We can see on the figure 2, the result in red of dssp and in blue of our script.

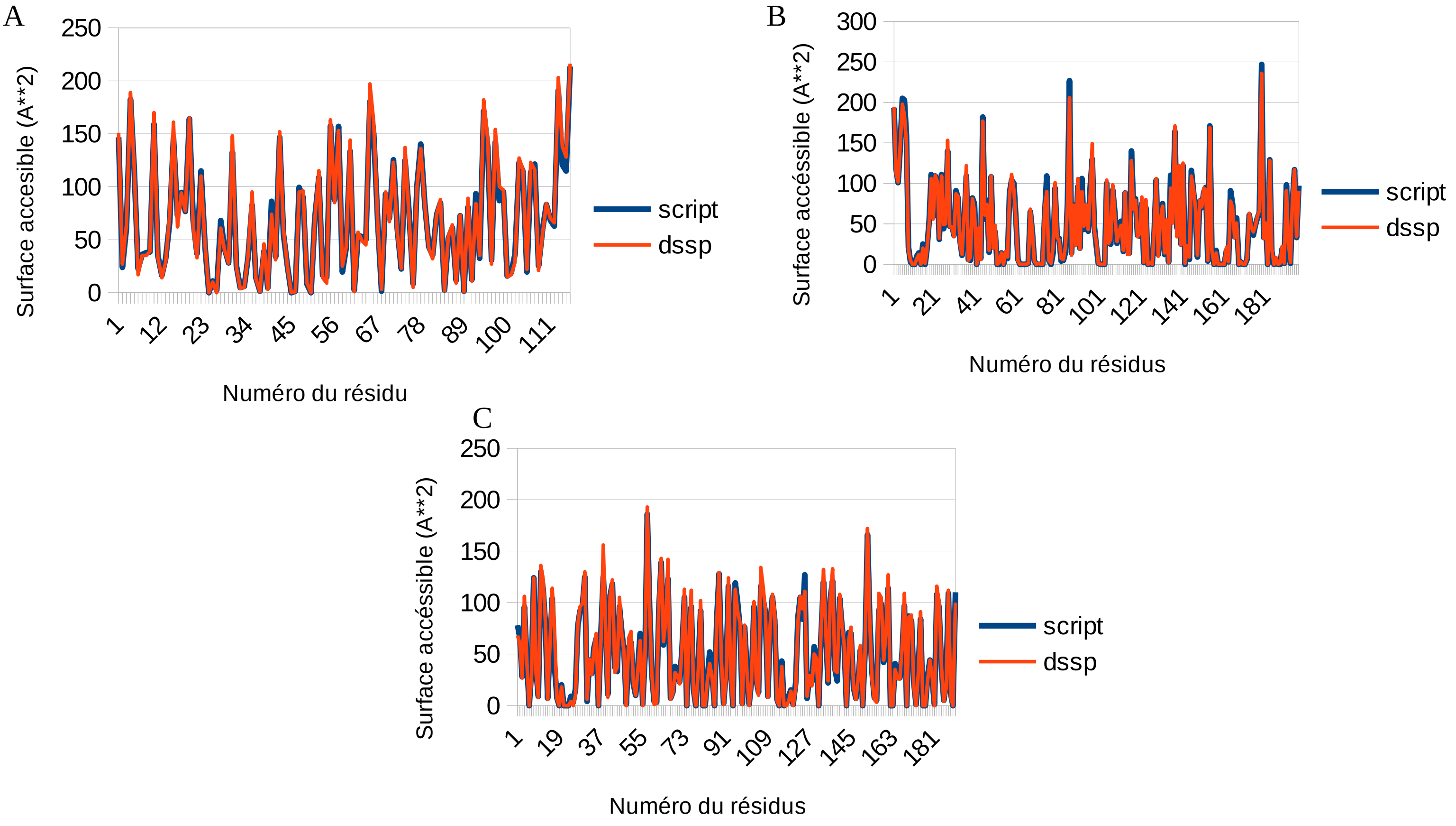
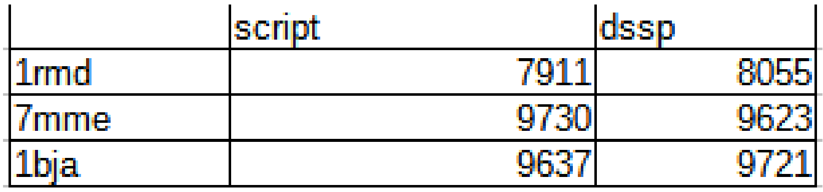


Figure 2: Curve of the ASA of each residues for our 3 tested proteins A: 1rmd; B: 7mme; C: 1bja;

We can see on figure 2 the differences between the results of our script and of dssp. The differences are small and was calculated. For 1rmd, the mean of the differences was 3,5%, for 7mme it was 3,3% and for 1bja was at 6%. For our 3 test the mean total of variation was at 4,2%. By taking 5% as limit on our sample of 3 proteins, we can affirm that the results are the same.

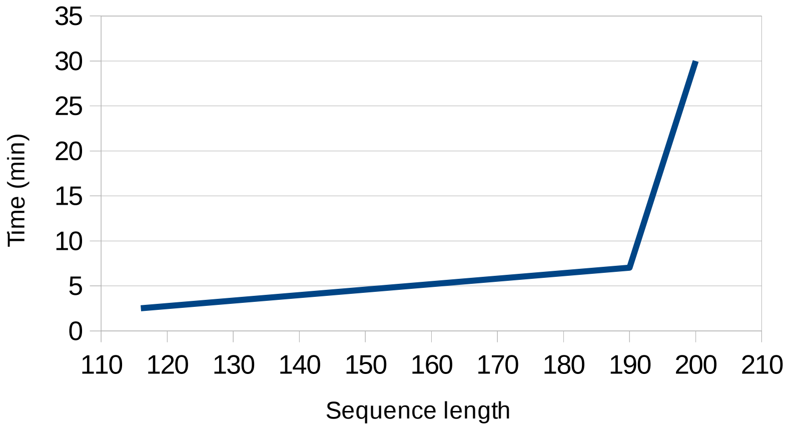


	script	dssp
1rmd	7911	8055
7mme	9730	9623
1bja	9637	9721

Also we found that the ASA of the protein is also close between the two with a mean of variation of 0,5%

Table 3: Result of ASA total of the protein obtain by the script and dssp

b. Time of execution:



The figure 3 is the time of execution for our 3 protein on a computer with a AMD Ryzen 5 2600 6 core and 32 GO of RAM. We can see that the higher the sequence length is, the more time the script need to process.

Figure 3: Time of executing the script on the 3 proteins

4. Discussion:

We manage to calculate the accessible surface area of each residue and the pourcentage of the protein exposed to the solvent but there is a major problem with the script which is the time of executing with only 200 amino acids it takes 30 minutes compare to dssp which is instantaneous. We can also note that there is a large differences of time between the protein with 190 aa and the one with 200 aa. It comes from the type of protein the one with 190 is a homo-dimers so the script will ignore some of the atom and will accelerate. The one with 200 is a monomers so each atom is unique.

5. Bibliography

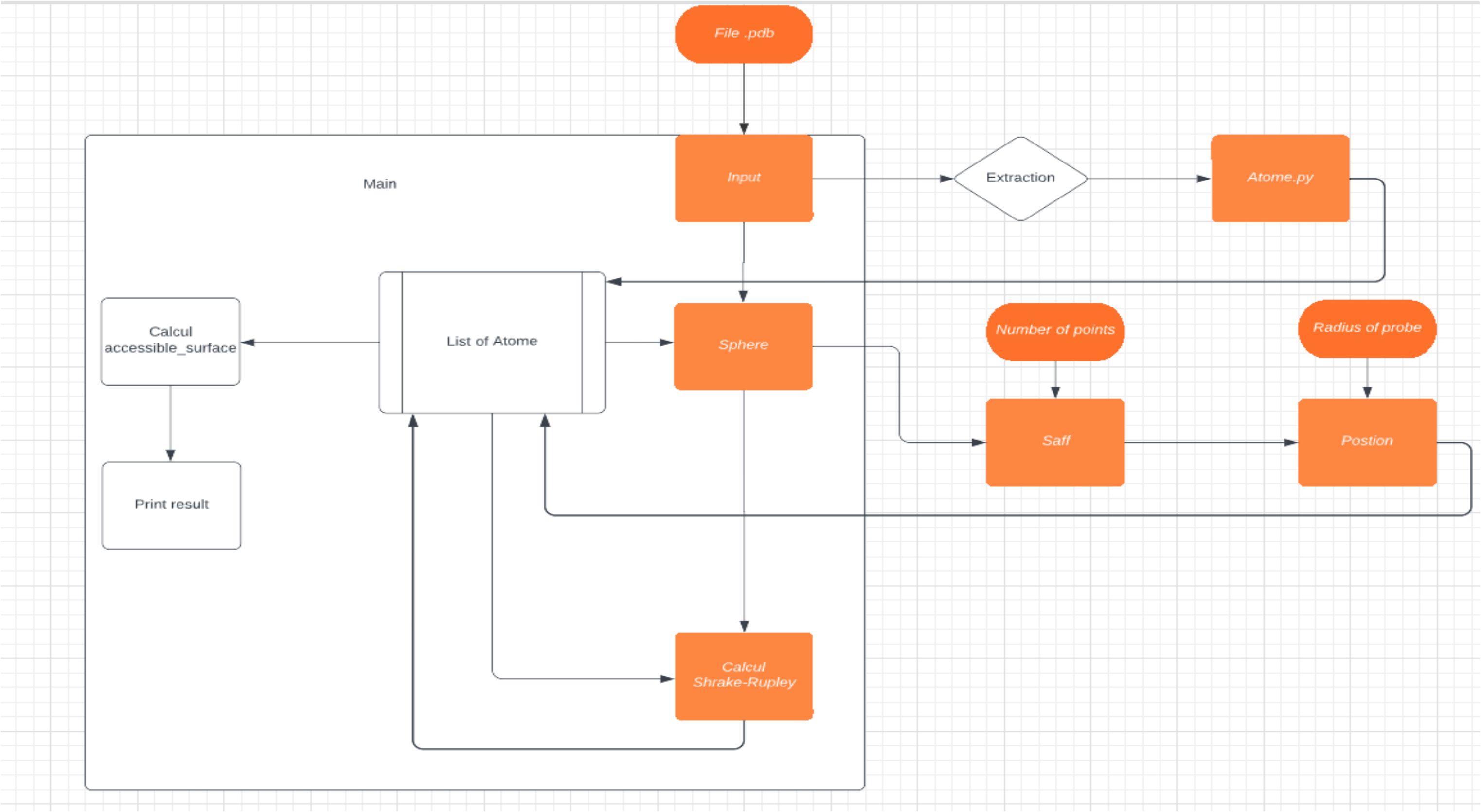
1.A.Bondi,«Van der Waals Volumes and Radii»,*J. Phys. Chem.*,vol.68,no3,1964,p.441–51,DOI10.1021/j100785a001

2.Momen-Roknabadi, A; Sadeghi, M; Pezeshk, H; Marashi, SA (2008)."Impact of residue accessible surface area on the prediction of protein secondary structures".*BMC Bioinformatics*.**9**: 357.doi:10.1186/1471-2105-9-357.PMC2553345.*PMID* 18759992.

3.Saff, E.B., Kuijlaars, A.B.J. Distributing many points on a sphere.*The Mathematical Intelligencer* 19, 5–11 (1997). https://doi.org/10.1007/BF03024331

4. Shrake, A; Rupley, JA. (1973). "Environment and exposure to solvent of protein atoms. Lysozyme and insulin". J Mol Biol 79 (2): 351–71. doi:10.1016/0022-2836(73)90011-9.

5.Tien, M. Z.; Meyer, A. G.; Sydykova, D. K.; Spielman, S. J.; Wilke, C. O. (2013)."Maximum allowed solvent accessibilites of residues in proteins".*PLOS ONE*.8(11):e80635.arXiv:1211.4251.Bibcode:2013PLoSO...880635T.doi:10.1371/journal.pone.0080635.PMC3836772.PMID24278298.



ANNEXE

Annex 1: Pipeline of the script

First, our script will recuperate the name file, the number of points by atom and the radius of the probe. Then in main, it will create an empty file named Data. Input will extract all the necessary information from the file and put it in many object Atom. These object Atom will be place in Data. Sphere will then calculate the spherical coordinate of each point on a atom and then for each object Atom in Data will calculate the coordinate (x,y,z) and put it in the object. In Calcul, the position between each atom will be calculate, if the distance is lesser than 10 A, it compare the distance between the point and the center of the second atom, and the radius of the second atom + the radius of the probe. If it is inferior then the point is buried.

When all the atom are compared, we then create a dictionary in accessible_surface to contain the accessible surface, of each residue, of residue hydrophobic, hydrophilic, the sum of all residue, the max ASA of all residue and then show the results.

- For the algorithm of Saff, we only need to use it once and then transform the spherical coordinate in 3D coordinate:

Here is the pseudo code of the algorithm of Saff

```
0 ≤ θ ≤ pi et 0 ≤ ϕ ≤ 2*pi

2 ≤ k ≤ N-1 et ϕ(1) =  ϕ(N) = 0

For k in N:

        h = -1 + (2*(k- 1)/(nbr-1))

        θ[k] = arccos(h)

        ϕ[k] =   ϕ[k-1] +(3,6/sqrt(N) * (1/sqrt(1-(h**2))) mod 2*pi)
```

Here the transformation:

x = (radius+probe) * sin(θ)*cos(ϕ) + x_of_the_center_of_the_atom

y = (radius+probe) * sin(θ)*sin(ϕ) + y_of_the_center_of_the_atom

z = (radius+probe) * cos(θ) + z_of_the_center_of_the_atom

- Here the pseudo-code for Shrake-rupley:

```
for atom_A in data:
        Select the atom_B in data close to atom_A:
        Compare the distance of the point of Atom A and the center of Atom_B with the
        radius of Atom_B + radius of the probe
        if it is superior then the point is free
        if it is inferior then the point is buried
```

- Here the formula used to calculate the accessible surface with the point:

surface = ((4*pi) / N)* (radius+probe**2) * number of free points

- To calculate the percentage of the protein exposed to the solvent, we used:

% = (accessible surface area of all residue / Maximal accessible surface area of all residue)

-Usage:

```
python3 main.py -p exemple/1bja.pdb
```