Short project report : Protein Interaction Calculator

Diane DABIR MOGHADDAM

September 18, 2019

1 Introduction

Intra-protein interactions have a strong role in the secondary and tertiary structure of a protein. They constrain helices and sheets, bond different monomeres together, and constrain which parts of the protein chain will be exposed and thus available for inter-protein interactions (with an enzyme or a target, for example). The number of bonds will also affect how robust against environment changes (for example, a higher number of hydrogen bonds will heighten the temperature necessary to denaturate the whole protein). Assessing these interactions in a simple manner is therefore a crucial part of organic chemistry. PIC (Protein Interaction Calculator) is an open-source program using python3.7. The program determines interactions within a protein, based on the informations contained in a PDB file of this protein. Implemented interactions are: disulphide bonds, hydrophobic interactions, ionic interactions and hydrogen bonds. More interactions will be added in the near future, such as aromatic-aromatic, aromatic-sulphur and cation-π interactions.

1.1 Disulphide bonds

Disulphide bonds are the result of a covalent bond between the sulphurs of two cystein amino-acids in a protein. They are detected by the program when the bond length is below 2.2Å.

1.2 Hydrophobic interactions

Hydrophobic interactions happen between hydrophobic residues : Valin, Alanin, Leucin, Isoleucin, Methionine, Phenylalanin,Tryptophane, Prolin, Tyrosin. The distance has to be below 5Å.

1.3 Ionic interactions

Ionic interactions happen between charged residues. The program finds couple between positive (Asparagin, Glutamin) and negative (Lysin, Arginin, Histidin)

residues which distance is inferior to 6Å.

1.4 Hydrogen bonds

Hydrogen bonds are found in 3 different scenarios : between atoms of the main chain (thus constraining the secondary structures, helix and sheets), between atoms of the amino-acids' residues, and between the main chain and residues.

1.5 Main chain/main chain

As of today, only hydrogen bonds between residues of the main chain can be found. The distance between the acceptor and the donor of hydrogen must be inferior to 3.5Å.

2 Material and Methods

2.1 Description of the algorithm

The script command.py requires a PDB file as input, containing at the very least the full list of atoms in the protein, as well as their coordinates. The script acts as follows :

- After input from the user, it parses the file and stocks each atom (and its characteristics) in a list.
- Depending on the searches the user specified, it will run a parse for each individual search, eliminating atoms or amino-acids that do not have a role in the specified interaction.
- The program then compares in this new list the distances between atoms. If the couple respects the interaction's criteria and if the distance is in the range of what is expected for this type of interaction, the couple of atoms and the distance between them is stocked as a list in a list.

The results are then displayed in a CSV file. Each section is announced by a title, then each bond appears as a line precising the position, the type (as the 3-letter code) and the chain of both residues involved in the interaction. The distance of the interaction can be fetched for the hydrogen bonds.

3 Results

Multiple molecules were tested with the algorithm : 2eti and 6pa8 were used as test samples. 2eti is a small protein, enabling fast parsing and debugging, while 6pa8 is a homomeric macromolecule containing 4 chains, which presents at least one interaction of each type.

4 Conclusion

PIC is a promising tool providing a first glance at the interactions at the intraprotein level.

5 Annexes

5.1 How to use

Quick start :

```
python3 run.py 6pa8.pdb --hydrophobic --disulfide --ionic
```

For an optimal use of this program, please follow these steps :

- Download the whole directory of the project, do not change the file tree
- Install MiniConda (follow this documentation : https://docs.conda.io/en/latest/miniconda.html)
- Import the environment from the `pic_env.yml` file
- Activate the `pic_env` environment
- In a terminal, in the main directory, type

```
python3 run.py <YOURPDB.pdb> --argument
```

The available arguments are :

- –disulfide, for disulfide bonds
- –hydrophobic, for hydrophobic interactions
- –ionic, for ionic interactions
- –hydrogen, for hydrogen bonds
- -h, –help for the help.

The result file can be found as a CSV file in the results directory.

5.2 Algorithm

When a file is submitted to the program, whatever the arguments, the PDB file is checked. If it ends with ".pdb", the program continues.

Whatever the arguments, the file is parsed, and every line beginning with "ATOM" is stocked in a list, along with the characteristics of the atom.

If an argument has been selected, a second parse is done on the list, and only the relevant residues or atoms are kept in a new list.

In this list, all atoms in general will have to be compared as pairs. The strategy used here is to have a first iterator which will roam the entire list, except for the last item in the list. A second iterator will roam at the same time, constrained from the next value after the first iterator to the last item of the list.

This strategy enables to test every pair possible, without having duplicate pairs, which is unwanted for all interactions, except for hydrogen bonds, where

the place of donor and acceptor is important. In the case of a hydrogen bond research, all pairs are tested commutatively. For each pair, the distance is calculated, and if it is inferior to the threshold, the pair is added to a list which contains the two atoms and the distance, which is then added to a list of interactions. This list is parsed, and for each line, a line is added to the CSV file.

As many of these steps are similar between different researches, the program could be optimized with more time available.

5.3 Future implementations

As this project confronted the author to their first real experience with OOP, the subject was not finished in its entirety, and priority was placed in creating a functional, documented and clean base code on which the other functions could be added on a later stage. Missing functions are :

- finding aromatic-aromatic interactions,
- finding hydrogen bonds between the side chains and the main chain,
- finding hydrogen bonds between side chains,
- finding aromatic-sulphur interactions,
- finding cation-π interactions,
- applying each interaction research to different monomeres of a protein,
- letting the user place the threshold for each interaction.

The `calcul_centre_masse` function would have been used to calculate the position of the centroid of an aromatic residue.