INSPECTOR POCKET

An open source software to detect protein binding sites

Elizaveta Korchevaya and Jordi Martín

MSc in Bioinformatics for Health Sciences - Universitat Pompeu Fabra
Introduction to Python & Structural Bioinformatics joint project

INTRODUCTION

Protein-ligand interactions are responsible for the vast majority, if not all, biological processes, and they occur in specific regions of the tri-dimensional protein structure referred to as pockets, cavities or clefts. In such regions, specific binding sites that interact with ligands are located. Ligands are usually small organic or inorganic molecules that are able to fit inside protein pockets and interact with specific residues to induce changes in the protein structure that can have long term effects on multiple levels.

Ligand binding site identification allows to gain insight into normal biological processes as well as pathological mechanisms of diseases and subsequent drug design. Computational approaches for such studies have the upper hand over experimental approaches in terms of cost and time efficiency. Hence, the development of ligand binding site prediction algorithms is currently on the rise.

Binding site detection methods can be categorized into geometry-based and energy-based approaches, as well as grid-based and grid-free methods. Geometry-based approaches analyze molecular surface shapes to identify cavities, while energy-based approaches assess interactions with probes or fragments to identify favorable pockets. These methods can operate on a Cartesian grid or without one [1].

In geometry-based, grid-based approaches like LIGSITE [2], a Cartesian grid is placed over the protein, scanning each grid point to tally Protein-Solvent-Protein events and identify buried points as pockets. Conversely, grid-free methods like SURFNET [3] place spheres between atom pairs, adjusting their radius to avoid clashes and defining resulting cavities.

In energy-based, grid-based methods like DrugSite [4], carbon probes on a grid interact with protein surroundings, with unfavorable energies discarded and remaining points merged into pockets. Grid-free methods, such as docking-based approaches, involve docking fragments against the protein and assigning pockets based on fragment binding quantities [1].

SELECTED APPROACH

Among many possible approaches to predict ligand binding sites, InspectorPocket is based on spatial geometry of a protein structure, specifically a geometric grid-based approach. This method allows it to directly work with protein structures encoded in the Protein Data Bank (PDB) format that contain all the necessary information for cartesian representation of atoms, residues and their assembly in a full protein. The core idea behind geometric-based ligand binding site prediction is based on localization of hollow cavities, or pockets through different geometric metrics from the available structural information. Even so, a plethora of different methods behind this rather simplistic

approach were described, starting from the breakthrough of POCKET that placed small spheres between two protein structures in the early 1990s [5]. Nowadays, geometric based approaches include complex space decomposition algorithms like the Voronoi tessellation, spatial probes like Alpha spheres and projections of 3D meshes or grids [1].

A particularly interesting method that uses 3D meshes is LIGSITE [2], as pointed out before, one of the first spatial geometry softwares developed.

During the development of InspectorPocket we've decided to design a hybrid approach that contains geometric analysis combined with clustering algorithms based on the spatial projections of the target protein on a grid and secondary structure analysis. InspectorPocket starts by calculating the geometric center of the protein and according to this center, the protein is mapped onto a grid, where each point is a delimited region of space. Therefore, the grid functions to decompose the space occupied by the protein. InspectorPocket takes advantage of linear algebra to perform fast transformations and changes of coordinates.

As previously mentioned, ligand binding sites tend to be located inside the pockets which are detected by an algorithm that analyzes the occupancy of the grid. A grid point is said to be occupied if protein atoms are present in it. The level of "buriedness", which represents cavity depth, is calculated to identify protein cavities. The specific term "buriedness" is used by other geometry-based software like CAVIAR [6] to remark that protein binding sites are hidden towards the inside of the protein. A clustering algorithm then finds neighboring grid points that are also occupied to facilitate the determination of pocket shape as well as binding site identification. Lastly, protein binding sites are identified as regions where pockets or binding sites interact with each other or with the solvent.

The detected pockets, clusters, and interfaces are typically visualized using molecular visualization software such as PyMOL. This allows researchers to inspect and analyze the spatial arrangement of cavities and binding sites within the protein structure.

The methodology employed by InspectorPocket can be delineated as follows:

1. **Grid Initialization**: The protein structure is mapped onto a grid, where each grid point represents a small region of space.

   At its core, the grid step follows a systematic approach to manage spatial data and perform calculations:

   The initialization process establishes the grid's structure by defining its dimensions and spacing. This allows for precise spatial representation and

organization of data. Atoms are then assigned to specific grid cubes based on their coordinates. This step enables efficient spatial indexing and facilitates subsequent calculations involving atoms and their spatial relationships.

The detection of buried cubes is a critical aspect of the approach. Buried cubes represent regions within the molecular structure that are surrounded by occupied grid points, often indicative of potential pockets or voids. Detecting and analyzing these buried regions can provide valuable insights into the molecular structure's functional characteristics.

2. **Pocket Detection**: The algorithm identifies pockets within the protein structure by analyzing the occupancy of grid points. A grid point is considered occupied if it contains atoms from the protein structure.

   It utilizes a spatial grid representation to discretize the 3D space occupied by the molecule. The matrix transformation process involves aligning the molecule's main axis with the coordinate axes (x-, y-, z- axes) to facilitate subsequent grid generation, as employed in LigSite.

   Initially, the molecule is rotated so that its main axis coincides with one of the coordinate axes, typically the z-axis. However, this rotation alone doesn't ensure that the molecule is centered at the origin (0,0,0) of the coordinate system. To address this, a transformation matrix is stored, preserving information about the rotation applied to the molecule. This matrix enables reverting the rotation back to the molecule's original orientation after subsequent calculations, ensuring consistency in the molecular coordinates.

   Additionally, the molecule undergoes a transformation aimed at minimizing the size of the grid needed for further analysis, such as grid-based ligand docking. This transformation involves adjusting the molecule's orientation and position in a manner similar to aligning its main axis with the x-, y-, z- axes. By aligning the molecule in this way, the extent of the grid required for subsequent calculations is minimized, optimizing computational resources and facilitating efficient analysis of ligand binding sites [7].

   Buried cubes within the molecular structure are then identified, representing enclosed regions.

3. **Cluster Identification**: Once pockets are detected, clusters of occupied grid points are identified within these pockets. Each identified cluster represents a distinct cavity or pocket within the protein structure. These clusters are characterized by a set of connected occupied grid points in the 3D mesh. This step

helps in delineating distinct cavities within the protein structure. Pockets are defined by selecting clusters that meet certain criteria indicative of a functional or structural cavity within the protein. Coordinates of the grid points corresponding to the identified clusters are extracted. These coordinates delineate the boundaries of the pockets within the protein structure.

Some thresholds were defined according to other softwares and approaches: minimum number of neighbors, degree of buriedness or minimum cluster size.

4. **Residues involved in each pocket:** After having clearly defined the pocket coordinates, the residues involved in each pockets are retrieved following the next approach. Alpha carbon (CA) atoms in each residue of the protein are searched and, for each CA the distances to each pocket coordinate are calculated. If the distance is within a 4 Angstroms threshold, the residue is considered involved in the pocket. The decision of the threshold can be justified by different reasons: based on typical pocket dimensions, amino acid side chain lengths (ensures that all residues are considered), functional significance, and empirical validation in structural biology studies

OUTPUT

Detected Pockets in PDB Format:
Pockets are identified and saved individually in PDB format files labeled as `PDBcode_pocketX.pdb`, where 'X' represents the pocket number. Each pocket is defined in the residue name as POK. Moreover, the residue sequence number corresponds to the number of the pocket, which is useful to separately identify each binding site in PyMOL.

Single Document Containing All Pockets
All detected pockets are consolidated into a single document named `PDBcode_all_pockets.pdb`, streamlining access and management of the pocket data.

PyMOL Script for Visualization:
A PyMOL script named `visualize_pymol.pml` is provided, facilitating direct visualization of the PDB structure along with its detected pockets represented as surfaces in PyMOL, a molecular visualization software.

Chimera Script for Visualization:
A Chimera script titled `visualize_chimera.cmd` is included, enabling direct visualization of the PDB structure along with its detected pockets in Chimera, another molecular visualization tool.

Pocket Report:

A text file named `pocket_report.txt` is provided, containing a comprehensive list of residues involved in each detected pocket.

REFERENCES

[1] TeachOpenCADD Project. (n.d.). T014_binding_site_detection. Retrieved from https://projects.volkamerlab.org/teachopencadd/talktorials/T014_binding_site_detection.html.

[2] Hendlich, M., Rippmann, F., & Barnickel, G. (1997). LIGSITE: automatic and efficient detection of potential small molecule-binding sites in proteins. Journal of molecular graphics & modelling, 15(6), 359–389. https://doi.org/10.1016/s1093-3263(98)00002-3

[3] Laskowski R. A. (1995). SURFNET: a program for visualizing molecular surfaces, cavities, and intermolecular interactions. Journal of molecular graphics, 13(5), 323–308. https://doi.org/10.1016/0263-7855(95)00073-9

[4] An, J., Totrov, M., & Abagyan, R. (2004). Comprehensive identification of "druggable" protein ligand binding sites. Genome informatics. International Conference on Genome Informatics, 15(2), 31–41.

[5] Zhao, J., Cao, Y., & Zhang, L. (2020). Exploring the computational methods for protein-ligand binding site prediction. Computational and structural biotechnology journal, 18, 417–426. https://doi.org/10.1016/j.csbj.2020.02.008

[6] Marchand, J. R., Pirard, B., Ertl, P., & Sirockin, F. (2021). CAVIAR: a method for automatic cavity detection, description and decomposition into subcavities. Journal of computer-aided molecular design, 35(6), 737–750. https://doi.org/10.1007/s10822-021-00390-w

[7] Masuya, M., & Doi, J. (1995). Detection and geometric modeling of molecular surfaces and cavities using digital mathematical morphological operations. Journal of molecular graphics, 13(6), 331–336. https://doi.org/10.1016/0263-7855(95)00071-2