Research article   **Open Access**

PGR: A Novel Graph Repository of Protein 3D-Structures

Wajdi Dhifli and Abdoulaye Baniré Diallo*

Computer Science Department, University of Quebec at Montreal, Downtown station, Montreal, Canada

Abstract

Graph theory and graph mining constitute rich fields of computational techniques to study the structures, topologies and properties of graphs. These techniques constitute a good asset in bioinformatics if there exist efficient methods for transforming biological data into graphs. In this paper, we present Protein Graph Repository (PGR), a novel database of protein 3D-structures transformed into graphs allowing the use of the large repertoire of graph theory techniques in protein mining. This repository contains graph representations of all currently known protein 3D-structures described in the Protein Data Bank (PDB). PGR also provides an efficient online converter of protein 3Dstructures into graphs, biological and graph-based description, pre-computed protein graph attributes and statistics, visualization of each protein graph, as well as graph-based protein similarity search tool. Such repository presents an enrichment of existing online databases that will help bridging the gap between graph mining and protein structure analysis. PGR data and features are unique and not included in any other protein database. The repository is available at http://wjdi.bioinfo.uqam.ca/

Introduction

The advances in computational and biological techniques of protein studies have provided enormous online databases. However, the complexity of protein structure requires adequate bioinformatics methods to mine these databases.

The principles of graph theory have been adopted to investigate organic molecules [1] and proteins [2-4]. The tertiary structure captures homology between proteins that are distantly related in evolution. With the availability of more protein 3D-structures due to techniques such as X-ray crystallography, increasing efforts have been devoted to directly deal with them. A crucial step in the computational study of protein structures is to look for a convenient representation of their spatial conformations. A possible representation of protein 3D-structure can be a graph of interconnected amino acids. Figure 1 shows an example of a signaling protein and its corresponding graph. The graph representation preserves the overall structure and its components. Such representation can be considered as an alternative to existent representations, such as the PDB format [5]. It allows fully exploiting the potential of data mining and graph theory algorithms to perform complex studies such as the discovery of important substructures in protein families which can be performed through frequent subgraph mining, pattern recognition, and functional motif discovery. Figure 2 shows two subgraphs corresponding to two recurrent substructures in a dataset of 38 proteins from the immunoglobin C1-set domains family, and their corresponding mapping on the original 3D-structure of the HFE (human) hemochromatosis protein. Such substructures are relevant for protein classification, protein function prediction, protein folding, *etc*. For instance, we have previously explored the potential of graph representation in the classification of four protein 3D-structure datasets, including G-proteins, immunoglobin C1-set domains, C-type lectin domains, and protein kinases catalytic subunit [4]. Frequent subgraphs were mined and used as features for the classification of each dataset. The experimental results showed that this graph-based approach outperformed the most competitive bioinformatics approaches including structural alignment-based classification (using Dali [6]) and Blast-based classification [7].

In this paper, we present *Protein Graph Repository* (PGR), an online repository of graphs representing all protein 3D-structures of the *Protein Data Bank* (PDB) [5]. PGR provides bioinformatics tools that facilitate the integration of graph theory techniques in the core of protein 3D-structure studies [4,8,9].

PGR Contents

Graph transformation of protein 3D structure

To transform protein 3D structures into graphs, it is straightforward to consider the relevant chemical interactions of the proteins. Chemical interactions are the electrostatic forces that hold atoms and residues together, stabilizing proteins and forming molecules that give them their 3 dimensional shape [3,8,10,11]. These interactions are mainly:

Covalent bonds between two atoms sharing a pair of valence electrons.

Ionic bonds between oppositely charged components.

Hydrogen bonds between two partially negative charged atoms sharing partially positive charged hydrogen.

Hydrophobic interactions where hydrophobic amino acids in the protein closely link their side chains together.

Van der Waals forces that represent transient and weak electrical attraction of one atom for another when electrons are fluctuating.
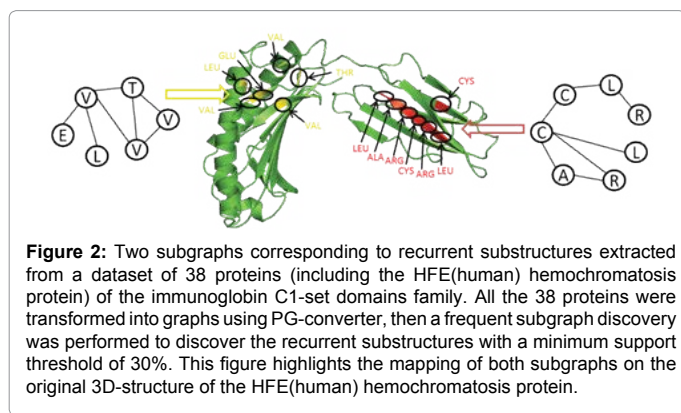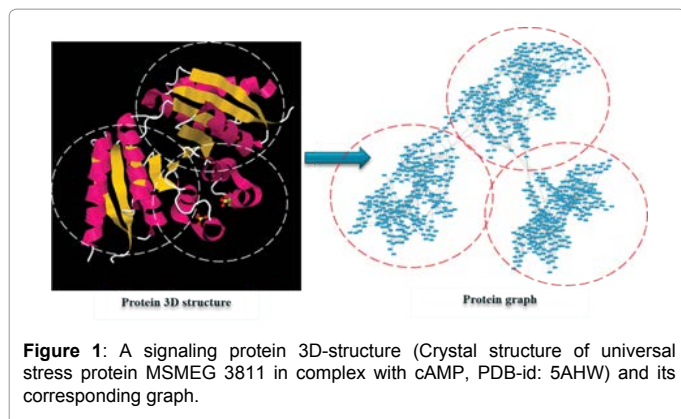
Existing transformation approaches of protein 3D-structure into graph, similarly consider amino acids as graph nodes, but they

---

Figure 1: A signaling protein 3D-structure (Crystal structure of universal stress protein MSMEG 3811 in complex with cAMP, PDB-id: 5AHW) and its corresponding graph.

Figure 2: Two subgraphs corresponding to recurrent substructures extracted from a dataset of 38 proteins (including the HFE(human) hemochromatosis protein) of the immunoglobin C1-set domains family. All the 38 proteins were transformed into graphs using PG-converter, then a frequent subgraph discovery was performed to discover the recurrent substructures with a minimum support threshold of 30%. This figure highlights the mapping of both subgraphs on the original 3D-structure of the HFE(human) hemochromatosis protein.

differ in considering the edges in attempt to reflect the truly existing interactions. In the following, we present the main approaches in the literature that are used for building protein graphs in PGR. Let G be a graph, u and v two nodes of G ($u, v \in G$), Δ a function that computes the distance between pairs of nodes $\Delta(u, v)$, and δ a distance threshold.

Main atom: Abstracts each amino acid in only one main atom, M_A [3, 10]. Two nodes representing two amino acids u and v are linked by an edge $e(u, v)$ if the Euclidean distance between their two main atoms $\Delta(M_A(u), M_A(v))$ is below a distance threshold δ. The main atom used in the literature is C_α with usually $\delta \geq 7\text{Å}$ on the argument that C_α atoms define the overall shape of the protein conformation [3].

All atoms: Considers the distances between all pairs of atoms $\Delta(A_A(u), A_A(v))$, where $A_A(u)$ represents all atoms of u ($A_A(u) = \forall atom \in u$) [11]. Two nodes representing two amino acids u and v are linked by an edge $e(u, v)$ if the Euclidean distance between any pair of atoms from both amino acids $\Delta(A_A(u), A_A(v))$ is below a distance threshold δ. Although this increases the complexity of graph building, it allows detecting connections that were omitted using Main Atom [11].

Both Main Atom and All Atoms suffer drawbacks. Since, the Main Atom technique abstracts amino acids into one main atom, it may omit possible edges between other atoms in the amino acids that are closer than their main atoms. Moreover, in the case of considering centroids of the amino acids as the main atoms, it may also suffer from two problems. In the case of two big amino acids, if their centroids are farther than the given distance threshold, they will be considered with no links while a real connection could be established between other close atoms (other than the centroids). In the case of small amino acids, if the distance between their centroids is smaller than the given distance threshold then they will be considered as connected while they

can be disconnected in reality. The all Atoms technique overcomes both limitations by theoretically considering the distance between all the atoms in the amino acids, this highly increases the runtime and complexity of the technique. However, the authors proposed some heuristics to alleviate the complexity. For instance, they consider only the distance between the side chains' centroids to decide whether their amino acids are connected or not, without regards to their chemical properties. This reduces the runtime but it may engender false edges.
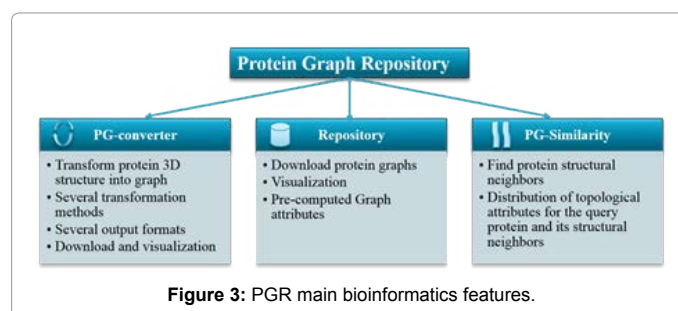
Main features of PGR

PGR can be divided in three main entries associated to the Protein conversion to graph, a comprehensive and complete databank of known protein structures and graph mining tools (Figure 3).

Graph repository: Here, we transformed all protein 3D-structures (of the PDB) into protein graphs (in PGR) using the described methods. The repository is enriched by a selection tool allowing the filtering and targeting of a specific population of proteins. Each protein graph can be displayed solely in a light-weight and interactive visualization interface using the best available visualization libraries including D3.js [12] and Cytoscape [13]. A set of the most important attributes for protein graph mining have been pre-computed including density, diameter, link impurity, etc. These attributes are presented with their Z-score according to all protein graph attribute distributions. So far, the repository contains 188 252 graphs corresponding to 94 126 protein 3D-structures from the PDB. The 188 252 protein graphs are composed of 94 126 graphs created using Main Atom method with C_α and 94 126 graphs created using All Atoms method. PGR data will be regularly updated according to the PDB.

PG-converter: PGR also provides an online converter that allows to upload and transform protein 3D-structures into protein-graphs. Available transformation methods include All Atoms, Main Atom based on $C\alpha$, $C\beta$, amino acid centroid, side chain centroid, amino acid ray, amino acid ray and side chain orientation, or side chain ray.

PG-similarity: We provide a search tool based on the pairwise similarity of structural protein attributes. Such tool could constitute an asset for several biological tasks such as protein classification and function prediction. The pairwise similarity between two protein graphs is measured by the distance between their corresponding vector representation based on the structural and topological attributes. We selected a set of attributes from the literature that are efficient in describing connected graphs [14,15]. Let $G = (V, E)$ be a graph representation of protein 3D structure. G is given as a collection of nodes V and a collection of edges E. We denote by $|V|$ the number of nodes (also called the graph order) and by $|E|$ the number of edges (also called the graph size). In the following, we list and define the set of used structural and topological attributes:

Average degree: The degree of a node u, denoted $deg(u)$, represents

Figure 3: PGR main bioinformatics features.

the number of nodes adjacent to u. The average degree of a graph G is the average value of the degrees of all nodes in G. Formally: $\deg(G) = \frac{1}{n}\sum_{i=1}^{n}\deg(u_i)$ where $deg(u_i)$ is the degree of the node u_i and n is the number of nodes in G.

Density: The density of a graph $G = (V, E)$ measures how many edges are in E compared to the maximum possible number of edges between the nodes in V. Formally: $c(u) = \frac{2e_u}{k_u(k_u-1)}$

Average clustering coefficient: The clustering coefficient of a node u, denoted by $c(u)$, measures how complete the neighborhood of u is, i.e., $c(u) = \frac{2e_u}{k_u(k_u-1)}$ where k_u is the number of neighbors of u and e_u is the number of connected pairs of neighbors. If all the neighbor nodes of u are connected, then the neighborhood of u is complete and we have a clustering coefficient of 1. If no nodes in the neighborhood of u are connected, then the clustering coefficient is 0. The average clustering coefficient of an entire graph G having n nodes, is given as the average value over all the nodes in G. Formally: $C(G) = \frac{1}{n}\sum_{i=1}^{n}c(u_i)$

Average effective eccentricity: For a node u, the effective eccentricity represents the maximum length of the shortest paths between u and every other node v in G, i.e., $e(u) = max\{d(u, v) : v \in V\}$. If u is isolated then $e(u) = 0$. The average effective eccentricity is defined as $C_e(u) = \frac{n-1}{\sum_{v \in |V/u|} d(u,v)}$, where n is the number of nodes of G.

Effective diameter: The effective diameter represents the maximum value of effective eccentricity over all nodes in the graph G, i.e., $diam(G) = max\{e(u)|u \in V\}$ where $e(u)$ represents the effective eccentricity of u as defined above.

Effective radius: The effective radius represents the minimum value of effective eccentricity over all nodes in the graph G, i.e., $rad(G) = min\{e(u)|u \in V\}$ where $e(u)$ is the effective eccentricity of u.

Closeness centrality: The closeness centrality measures how fast information spreads from a given node to other reachable nodes in the graph. For a node u, it represents the reciprocal of the average shortest path length between u and every other reachable node in the graph, i.e., $C_c(u) = \frac{n-1}{\sum_{v \in |V/u|} d(u,v)}$ where $d(u, v)$ is the length of the shortest path between the nodes u and v. For a graph G, we consider the average value of closeness centrality of all the nodes, i.e. $C_c(G) = \frac{1}{n}\sum_{i=1}^{n}u_i$

Percentage of central nodes: Here, we compute the ratio of the number of central nodes from the number of nodes in the graph. A node u is considered as central point if the value of its eccentricity is equal to the effective radius of the graph, i.e., $e(u) = rad(G)$.

Percentage of end points: It represents the ratio of the number of end points from the total number of nodes of the graph. A node u is considered as end point if $deg(u) = 1$.

Neighborhood impurity: The impurity degree of a node u belonging to a graph G, having a label $L(u)$ and a neighborhood (adjacent nodes) $N(u)$, is defined as $ImpurityDeg(u) = |L(v): v \in N(u), L(u) \neq L(v)|$. The neighborhood impurity of a graph G represents the average impurity degree over all nodes with positive impurity.

Link impurity: An edge $\{u, v\}$ is considered to be impure if $L(u) \neq L(v)$. The link impurity of a graph G with k edges is defined as: $\frac{|\{u,v\} \in E : L(u) \neq L(v)|}{k}$

Label entropy: It measures the uncertainty of labels. The label entropy of a graph G having k labels is measured as

$E(G) = -\sum p(l_i)\log p(l_i)$, where l_i is the ith label.

Compared to the conventional distance matrix representation [6] *PG-similarity* measures the global structural and topological similarity between protein structures on a macro side, whereas distance matrix based similarity operates on a micro side and looks into every single detail in compared structures. Even though both similarity methods should be highly correlated and not diverge (as similar structures have similar topological descriptions), each method has its positive and negative sides. Distance matrix based methods has the advantage of detecting exact superposition and local matching sites, however, they are combinatorial and thus computationally greedy. *PG-similarity* method is based on a vector embedding of graphs of protein structures based on a set of structural and topological attributes. This makes it unable to return local matches, however, such strategy makes it able to capture structural similarity in a very fast way. Moreover, some attributes, like clustering coefficient and neighborhood impurity, makes *PG-similarity* able to reveal hidden similarities that are undetected using existing methods.

Conclusion

With the growth of protein 3D-structures in online databases, the transformation of protein 3D-structures into graphs of interconnected amino acids and the application of graph mining concepts constitute a relevant feature for the development of rapid and efficient computational techniques. In this paper, we introduced Protein Graph Repository (PGR), a novel database of protein 3D-structures transformed into graphs allowing the use of the large repertoire of graph theory techniques in protein mining. PGR provides three main features that are unique and not included in any other online protein database, namely the *graph repository*, *PG-converter* and *PG-similarity*. The *graph repository* contains graph representations of all currently known protein 3D-structures described in the PDB. Each graph representation is enriched by graph based description and light weight interactive visualization. *PG-converter* provides an efficient online converter of protein 3D structures into graphs. *PG-similarity* offers a pairwise similarity search tool based on a set of informative structural and topological attributes. Such tool could constitute an asset for several biological tasks such as protein classification and function prediction. PGR is an independent repository, but it can also act as a complementary resource to existing ones such as the PDB. PGR is apt for extension and additional services and functionalities will be added in the next coming versions.

References

1. Borgelt C, Berthold MR (2002) Mining molecular fragments: Finding relevant substructures of molecules. IEEE International Conference on Data Mining (ICDM), Japan.

2. Milik M, Szalma S (2003) Common structural cliques: A tool for protein structure and function analysis. Protein Engineering 16: 543–552.

3. Huan J, Bandyopadhyay D, Wang W, Snoeyink J, Prins J, et al. (2005) Comparing graph representations of protein structure for mining family specific residue-based packing motifs. Journal of Computational Biology 12: 657–671.

4. Dhifli W, Saidi R, Mephu Nguifo E (2014) Smoothing 3D protein structure motifs through graph mining and amino-acids similarities. Journal of Computational Biology 21: 162–172.

5. Berman HM, Westbrook JD, Feng Z, Gilliland G, Bhat TN, et al. (2000) The protein data bank. Nucleic Acids Research 28: 235–242.

6. Holm L, Rosenström P (2010) Dali server: conservation mapping in 3d. Nucleic Acids Research 38: 545–549.

7. Altschul S, Gish W, Miller W, Myers E, Lipman D (1990) Basic local alignment search tool. Journal of Molecular Biology 215: 403–410.

8. Stout M, Bacardit J, Hirst JD, Smith RE, Krasnogor N (2008) Prediction of topological contacts in proteins using learning classifier systems. Soft Computing 13: 245–258.

9. Malod-Dognin N, Przulj N (2014) Gr-align: fast and flexible alignment of protein 3d structures using graphlet degree similarity. Bioinformatics 30: 1259–1265.

10. Lovell SC, Davis IW, Arendall WB, Word JM, Prisant MG, et al. (2003) Structure validation by Cα geometry: φ, ψ and Cβ deviation. Proteins: Structure, Function, and Genetics 50: 437–450.

11. Saidi R, Maddouri M, Mephu Nguifo E (2009) Comparing graph-based representations of protein for mining purposes. In: Proceedings of the ACM KDD Workshop on Statistical and Relational Learning in Bioinformatics: 35–38.

12. Bostock M, Ogievetsky V, Heer J (2011) D3: Data-driven documents. IEEE Trans Visualization & Comp Graphics.

13. Smoot ME, Ono K, Ruscheinski J, Wang PL, Ideker T (2011) Cytoscape 2.8: new features for data integration and network visualization. Bioinformatics 27: 431–432.

14. Li G, Semerci M, Yener B, Zaki MJ (2012) Effective graph classification based on topological and label attributes. Statistical Analysis and Data Mining 5: 265–283.

15. Leskovec J, Kleinberg J, Faloutsos C (2005) Graphs over time: densification laws, shrinking diameters and possible explanations. ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD): 177–187.