CHAPTER V

MATCHING PROTEIN STRUCTURES [‡]

[‡]This part of the work has been published in International Journal of Combinatorial Optimization Problems and Informatics [13]

Chapter 5

MATCHING PROTEIN STRUCTURES

5.1 Introduction

In bio-informatics, structural comparison of proteins is useful in several domains. The number of known protein structures is increasing rapidly due to advances in the technology required for the determination of protein structures. Thus, there is an increasing need for efficient techniques to compare and classify both new and existing proteins according to their structural properties. Geometric hashing is a widely used method for structural comparisions and pattern recognition [95]. Over the last decade, a number of techniques for structurally comparing proteins have been developed how-

ever none, have proved adequate across a range of applications. A relatively new technique, Contact Map Overlap (CMO), first proposed in [52], is to identify alignments between protein contact maps with the goal of maximizing the number of consistent alignments. However the algorithm for finding cliques are frequently used in bioinformatics, where these algorithms have been applied to compare three-dimensional molecular structures [107]. i.e., searching for the maximum clique is often bottle-neck computational step in these applications. As shown in [112], protein structural alignment problem can be directly translated to a maximum clique problem (MCP) which calls for finding the maximum sized subgraph of pairwise adjacent vertices in a given graph. Three dimensional objects that are to be compared with the clique algorithms must be represented as graphs of vertices and edges. Vertices are points with labels in three-dimensional space.

5.2 Theory

Protein graph: A protein consists of a chain of residues (amino acids). When a protein folds into its tertiary (lowest energy) structure, residues that are not directly adjacent in the chain may be physically close in the space. Components of the protein are identified as the alpha carbon atoms (C_α)

of each residue (amino acid). contact map overlap (CMO) represents this three dimensional structure of protein into a matrix of all pairwise distances between the components of the protein. It is further simplified into a 0-1 contact map by encoding each pairwise distance as one if the pairwise distance is less than some threshold. i.e., an edge between two vertices is drawn if the difference between the distance of vertices $<$ resolution. In this study resolution being 2.0 A° and the pairwise distance is 4.0 - 30.0 A°.

Product graph: Two protein graphs are used to construct a product graph. A maximum clique in this graph corresponds to the maximum substructure that is common to both graphs [103]. A vertex in a product graph is a pair of vertices (m, n), where the first member, vertex m, belongs to the first protein graph, and the second member, vertex n, belongs to the second protein graph. An edge between two vertices (m, n) and (p, q) is drawn if the difference between distances $d(m, p)$ and $d(n, q)$ is less than the resolution. The example of a product graph based on the defintion of edge product graph in the section 2.1 shown in the Figure 5.1 and the examples of two protein graphs, together with their product graph, are shown in Figure 5.2. In this figure a product graph is constructed from two protein graphs and the maximum clique in this product

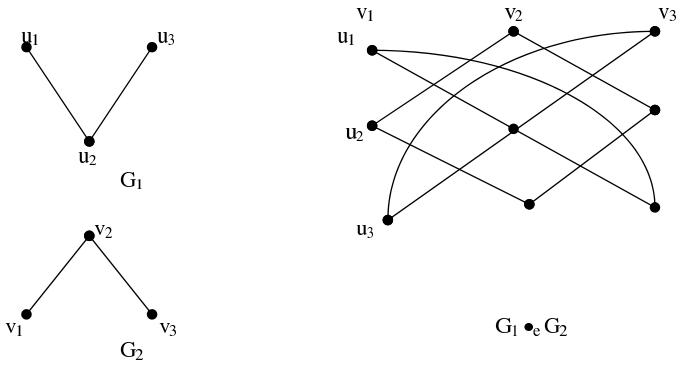

Figure 5.1: Edge product of G_1 and G_2

graph $\{(AC, AC), (DO, DO), (PI, PI), (PI, PI), (AL, AL)\}$ corresponds to the maximum common substructure in the two original graphs

5.3 Experimental Results and Analysis

To evaluate the ability of the VSA in detecting the similarities in protein structures, the benchmark instances have been taken from the protein data bank [17]. The surface residues have extracted around the binding site of each of the proteins. The proteins in this benchmark range in size from 55 to 200 residues. The Superposition of the binding sites residues of the proteins with PDB codes 1TPO and 2PRK are shown in Figure 5.3. Protein 1TPO coloured black and protein 2PRK is grey. Black and grey points correspond to aligned vertices of the two protein graphs.

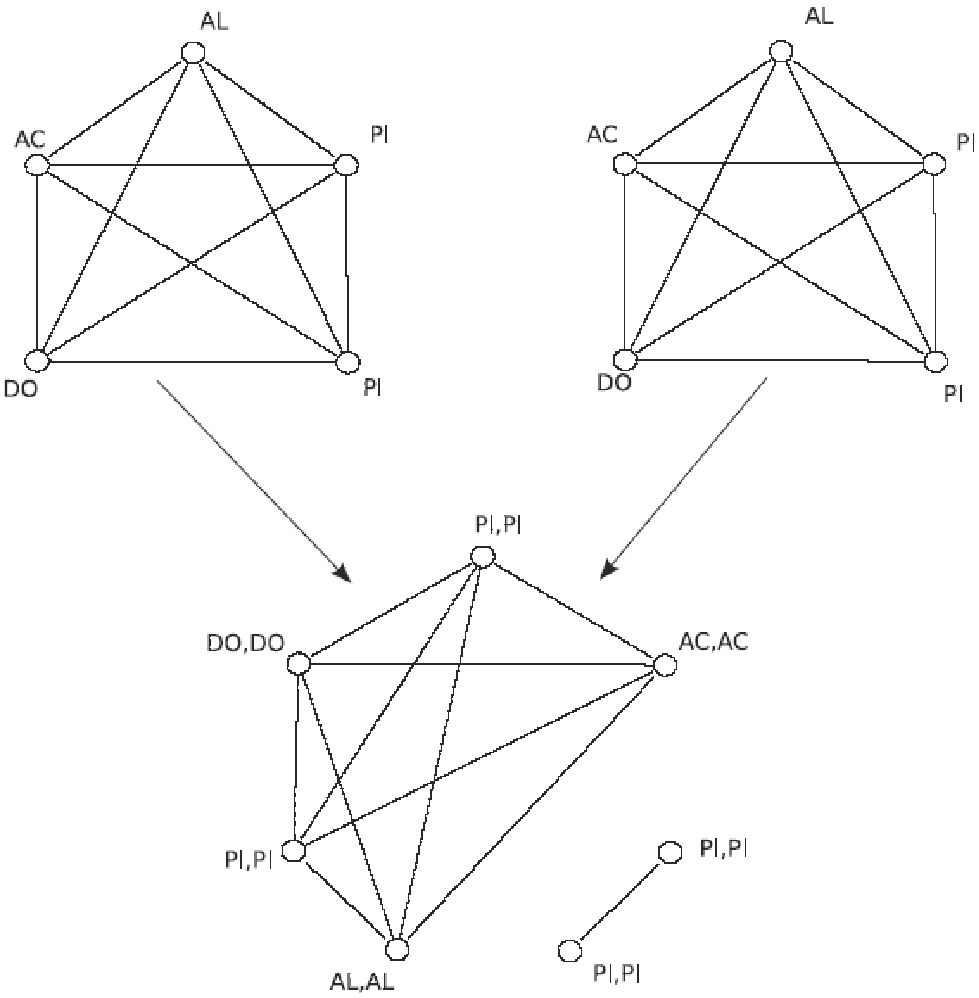

Figure 5.2: Product graph from two protein graphs

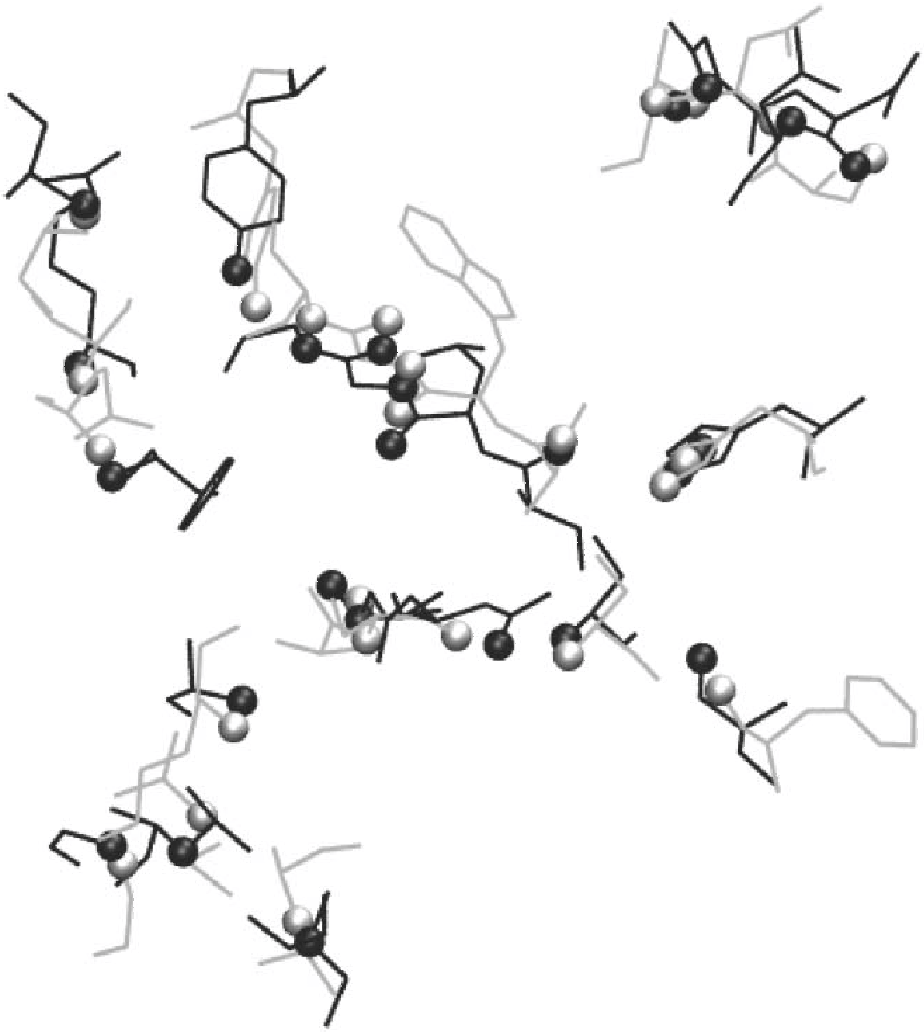

Figure 5.3: The Superposition of the binding sites of the proteins with PDB codes 1TPO and 2PRK

The Universality Similarity Measure (USM) software [74] was used to generate the contact maps for these proteins. From the contact maps for the proteins to be compared, the two-dimensional grid G was generated and the correspondence graph created by adding an edge when the two alignments represented by pairs of vertices are a feasible solution to the CMO problem. Then a product graph constructed from the two protein graphs, which was input to the proposed algorithm, correspondence graphs have up to approximately 8000 vertices. All experiments were carried out an Intel Pentium Core2 Duo 1.6 GHz CPU and 1 GB of RAM. The performance of VSA on PDB benchmark instances averaged over 100 trials is shown in Table 5.1, for each instance the corresponding clique sizes obtained in [100] taken as the optimum solution for these instances. In this table 'Max. Clique' is the known maximum clique size. 'CPU(s)' is the VSA run-time in CPU seconds for each instances averaged over all successful trials. 'SCPU(s)' is the CPU time reported in [100], scaled by 2.31 to allow some basis for comparison with the reference computer used in this study. From the table, it is clear that the VSA achieved 100% success rate for most of the test instances with considerably less processor time than that required in [100].

Table 5.1: VSA performance on PDB benchmark instances averaged over 100 trials

Problem Instance	G Vertices	Max. Clique	Success Rate	VSA CPU(s)	SCPU(s)
1A8O-1F22	2728	25	100	13.56	30.52
1AVY-1BCT	6278	51	100	56.20	630.30
1B6W-1BW5	4131	34	100	26.98	84.09
1BAW-2B3I	7200	53	96	12.87	28.36
1BCT-1BW5	4386	47	100	65.56	1117.91
1BCT-1F22	3784	25	100	3.23	20.38
1BCT-1ILP	4988	30	100	14.89	23.62
1BPI-2KNT	1848	32	100	23.89	55.24
1C7V-1C7W	2401	34	100	67.34	83.62
1C9O-1KDF	2805	21	100	37.54	52.63
1DF5-1F22	3960	27	100	28.64	46.27
1KDI-1BAW	7920	53	100	123.62	610.23
1KDI-1PLA	6424	53	100	89.23	369.05
1KDI-2B3I	7040	47	100	63.58	83.52
1KDI-2PCY	7216	58	95	86.28	300.03
1NMF-2NEW	2728	21	100	21.67	57.65
1NMG-1WDC	4698	17	100	32.48	84.34
1PFN-1SVF	5992	30	100	38.23	40.62
1PLA-1BAW	6570	55	100	97.54	117.95
1PLA-2B3I	5840	47	100	83.27	218.30
1PLA-2PCY	5986	57	98	79.65	368.79
1TPO-2PRK	2435	24	100	12.40	80.16
1VII-1CPH	903	15	100	34.83	138.98
1VNB-1BHB	6120	28	100	12.78	33.62
2KNT-1KNT	1980	41	100	10.65	24.76
2NEW-3MEF	2552	16	100	12.76	17.94
2PCY-1BAW	7380	66	94	75.32	103.34
2PCY-2B3I	6560	52	100	34.67	58.76
3EBX-1ERA	2205	19	100	11.98	65.43
3EBX-6EBX	2331	25	100	16.54	25.64
5PTI-1BPI	1596	35	100	8.32	23.65
5PTI-1KNT	1710	31	100	5.26	17.34
5PTI-2KNT	1672	32	100	4.43	16.45
6EBX-1ERA	1295	22	100	6.34	21.78

5.4 Features

In this chapter we have described how the proposed VSA with some modification efficiently performs in computing common substructure of proteins. With this procedure it is possible to reduce the time to find the maximum commmon substructure of two proteins. The proposed VSA is considerably faster than a widely used algorithm. Its use is likely in protein structural comparison and protein classifications.