AMRITA SCHOOL OF ENGINEERING, AMRITAPURI

AMRITA VISHWA VIDYAPEETHAM

AMRITAPURI CAMPUS

Complete Enumeration of Compact Structural Motifs in Proteins

1.	Prasanth R	(AM.EN.U4CSE10142)
2.	Swamy T	(AM.EN.U4CSE10060)
3.	Kaushik S	(AM.EN.U4CSE10056)
4.	Sushanth M	(AM.EN.U4CSE10129)

Under the guidance of
Dr. Bhadrachalam Chitturi

1. Introduction

Identifying the copies of a particular structure in a given protein structure is of fundamental significance in structural biology. This helps one establish relations among pairs of proteins. The substructures of interest are called motifs. The search of structural motifs that specify the spatial arrangement of polypeptide segments i.e. secondary structure elements or simply SSEs is preferred over other methods such as structural superposition in comparing protein structures. 3D protein structures can be modelled as graphs whose maximum degree is bounded by a constant [4]. Structural motifs can also be modelled as graphs and a significant percentage of them are trees [4]. In this project G denotes the protein graph and Q denotes the query i.e. the motif where Q is restricted to be a tree. That is, we restrict ourselves to identifying the copies of a tree in G.

2. The Problem

The existing systems have limitations. For example, ProSMoS [1, 2] finds all the copies of a motif in a protein; however, the search process mimics the order of SSEs specified in the motif. For example, given a motif X with k SSEs $(1,2,\ldots,k)$ and a protein P, SSE indices $(i_1,i_2,\ldots,i_k)$ in P are possible candidates for X only if $i_j < i_{j+1}$ $(1\leq j < k)$ [4]. Similarly, Aung and Li [3] extract sequence independent common motifs from an input that consists of a set of proteins but they limit the queries (motifs) to be cliques.

Several pairs of proteins are known to exhibit similar properties based on a shared motif. Even though the shared motif is identical in structure in both the proteins, the sequence of its SSEs is not identical in both proteins. Thus, there is a need to identify the presence of a motif in a protein where the numbering of the SSEs in the motif can be ignored. However, the existing methods do not address the general version of this problem; [3] is applicable to only cliques. In [4] a more general method is proposed based on identifying the copies of a given tree in a given graph that can be extended to identify the copies of sparse graphs. The aim of this project is to implement an algorithm that identifies all copies of a motif, a tree, in a protein independent of the SSE sequence.

3. Proposed Solution

The article "Complete Enumeration of Compact Structural Motifs in Proteins" by Bhadrachalam Chitturi, Doina Bein & Nick V. Grishin gives an efficient divide-and-conquer algorithm that finds all copies of the query graph Q (a tree) in the graph G by partitioning Q using a minimum dominating set. Here, the order of the SSEs in Q is ignored. A dominating set is a set of nodes S such that every node in the graph G is a neighbour of at least one element of S. The Minimum Dominating Set problem is to find minimum such S for a given graph. This can be reduced to subgraph isomorphism problem. The subgraph isomorphism problem is a computational task in which two graphs G and H are given as input, and one must determine whether G contains a subgraph that is isomorphic to H.

Bhat and Cosmadakis [6] proved that finding an embedding of a given tree in a grid, whose nodes have a maximum degree of four, is also NPcomplete. Thus, identifying a copy of a given tree Q in G is NP-complete. The method suggested in [4] can be extended to identify the copies of sparse graphs that are supergraphs of the corresponding spanning trees. That is, a query graph can be a connected graph which contains a spanning tree of the nodes. However, one can have only a constant number of additional edges compared to the corresponding spanning tree [4]. So, this a more general method compared to the existing methods.

The proposed system of [4] consists of Tasks (i)-(iv) as shown in Fig. 1. Our main contribution is the implementation of Task (iii). Task (i) constructs graphs $\{G_i\}$ for all proteins with the help of PALSSE

[5]. PALSSE is a software system that generates a corresponding graph for a given protein. This step creates a database offline. In Task (ii), given a motif as a matrix, the corresponding tree *Q* is obtained such that cardinality of *S* is minimum, where *S* is a minimum dominating set of *Q*. Note that in [4] the query graph (motif) need not be a tree; thus, certain edges might be removed. In Task (iii), $\{G_i\}$ is efficiently searched for the copies of *Q*. In Task (IV), the copies of *Q* thus found are processed to identify the copies of Q^o the original query.

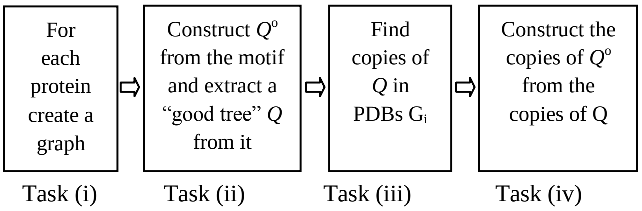
For each protein create a graph
Construct Q^o from the motif and extract a "good tree" *Q* from it
Find copies of *Q* in PDBs G_i
Construct the copies of Q^o from the copies of Q
Task (i)
Task (ii)
Task (iii)
Task (iv)

Figure 1. System Overview

The intended sequence of the activities along with the time line is stated below.

• Identify the centre of the tree, decompose the tree into Dominions and identify the copies of the dominions in G	2 weeks
• Implement tree isomorphism algorithm	2 weeks
• Enumerate the copies of Q in G and eliminate duplicates	7 weeks
• Other unexpected issues	4 weeks
• Write project report	2 weeks

4. Team Organization

We have a team of 4 members and an additional member as our project guide. The assignment of tasks among the team members is maintained equally.

Finding the centre of the given tree T	Swamy
Dividing T in to different dominions	Sushanth
Tree isomorphism algorithm research	Prasanth, Kaushik, Swamy and Sushanth
Implementing tree isomorphism	Kaushik, Prasanth and Sushanth
Finding copies of given T in G	Prasanth, Kaushik and Swamy
Eliminating duplicates	Prasanth and Kaushik
Writing project report	Sushanth, Swamy

5. Resource Requirements

Laptop computer Speed (minimum)	: Intel I3 (2.67 GHz)
Memory	: 4 GB RAM
Programming language	: C++ (C++ 4.8.1)
Editor (IDE)	: Netbeans (7.4)
Compiler	: C++11

Bibliography

[1] Shi, S., Zhong, Y., Majumdar, I., Krishna, S.S., and Grishin, N.V. Searching for three-dimensional secondary structural patterns in proteins with ProSMoS. *Bioinformatics* 23, 11 (2007) 1331-1338.
[2] Ullman, J.R. An algorithm for sub-graph isomorphism. *J. ACM* 23, 1 (1976) 31-42.
[3] Aung, Z., and Li, J. Mining super-secondary structure motifs from 3D protein structures: a Sequence order independent approach. *Genome Informatics,* 19 (2007) 15-26, PMID: 18546501.
[4] Bhadrachalam. C, Doina Bein & Nick V. Grishin. Complete Enumeration of Compact Structural Motifs in Proteins.
[5] Majumdar, I., Krishna, S.S., and Grishin, N.V. PALSSE: a program to delineate linear secondary structural elements from protein structures, BMC Bioinformatics 6 (2005) 202.
[6] Bhat, S. N., Cosmadakis, S. S: The Complexity of Minimizing Wire Lengths in VLSI Layouts. Inf. Process. Lett. 25(4): 263-267 (1987).