BIOINFORMATION
Discovery at the interface of physical and biological sciences

open access www.bioinformation.net Software

Volume 9(10)

*Spec*P: A tool for spectral partitioning of protein contact graph

Saritha Namboodiri* & Kripadas K

Department of Computer Science, V.T.Bhattathiripad College, University of Calicut, Thenhipalam, Kerala 673635 India; Saritha Namboodiri – Email: saritha16.namboodiri@gmail.com; *Corresponding author

Received May 15, 2013; Accepted May 17, 2013; Published June 08, 2013

Abstract:
***Spec*P** is an open-source Python module that performs Spectral Partitioning on Protein Contact Graphs. Protein Contact Graphs are graph theory based representation of the protein structure, where each amino acid forms a 'vertex' and spatial contact of any two amino acids is an 'edge' between them. Spectral partitioning is carried out in ***Spec*P** based on the second smallest spectral value (eigen value) of the Protein Contact Graph. The eigen vector corresponding to the second smallest spectral value are partitioned into two clusters based on the sign of the corresponding vector entry. Spectral Partitioning algorithm is repeatedly carried out until the desired numbers of partitions are obtained. ***Spec*P** visualizes the spectrally partitioned clusters of protein structure along with the Protein Contact Map and Protein Contact Graph which can be saved for later use. It also possesses an interactive mode whereby the user has the ability to zoom, pan, resize and save these raster images in various image formats (.eps, .jpg, .png) manually. ***Spec*P** is a stand-alone extensible tool useful for structural analysis of proteins.

Background:
Spectral partitioning is a graph partition algorithm which partitions data represented in the form of a graph G = (V,E), with V vertices and E edges, into smaller components with specific properties **[1].** *Spectral partitioning* has gained momentum in recent times due to its simplicity and better performance. They have been successfully applied in protein science **[2].**

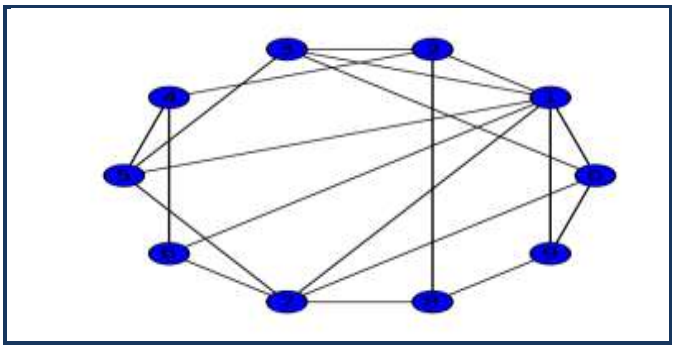

Figure 1: Protein Contact Network (first 10 nodes) of the protein (PDB id 4q21).

Proteins are linear, ordered chain of amino acids that fold by virtue of chemical forces to form a 3D structure. Coarse grain models of proteins using graph theory are spawned to gain insight into the structures of proteins. Proteins are depicted as graph with the amino acids as nodes and the positional information of the C_α atoms that form the backbone of protein structure, as edge connectivity or 'contact'. This graph-theory based network forms the *Protein Contact Graph* **[3].** The Protein Contact Graph of the first 10 nodes for the protein (PDB id 4q21) is as shown in **(Figure 1)**. Protein Contact Maps are a reduced representation of Protein Contact Graph **[4],** providing a quick way of visually inspecting structural features. A *contact map or adjacency matrix* is a square matrix *M* where $M_{i,j}$=1 if the distance between Cα atoms of residues *i* and *j* is below cut-off threshold atomic distance, or Mi,j=0 otherwise. The cut-off atomic threshold distance is > 8 Å for long-range contact network, between 4Å and 8Å (inclusive) for medium-range contact network and < 4 Å for short-range contact networks.

The Spectral Partitioning algorithm is applied on the Protein Contact Map obtained from Protein Contact Graph. The algorithm considers the eigen vectors (Fiedler vector) of the second smallest eigen value that yields a lower bound on the

optimal cost of ratio-cut partition and bisects the graph into two disjoint sets based on the sign of the corresponding vector entry **[5].** The algorithm is repeated until desired numbers of partitions are obtained. ***Spec***P can generate *Spectral Partitions*, *Protein Contact Graph and Protein Contact Map* that can be visualized and the saved for later use.

Methodology:
Computing the *adjacency matrix* is the first step in *Spectral partitioning*. The adjacency matrix for the first 10 amino acids of the protein with PDB id 4q21 is given in **(Figure 2).**

To compute the Laplacian matrix from the adjacency matrix, the *degree matrix* must be obtained. *Degree matrix* is a diagonal matrix that holds the degree of each vertex of a graph. All the elements of a degree matrix are 0 except for the diagonal elements. The degree of each vertex is computed by summing up each row (that correspond to a vertex) of the adjacency matrix and placing it in the diagonal element of that row. The *degree matrix* of the above adjacency matrix is given in **(Figure 2).**

Spectral partitioning which takes the Laplacian matrix, is worked out as L=D-A, where D is diagonal degree matrix and A is the adjacency matrix.

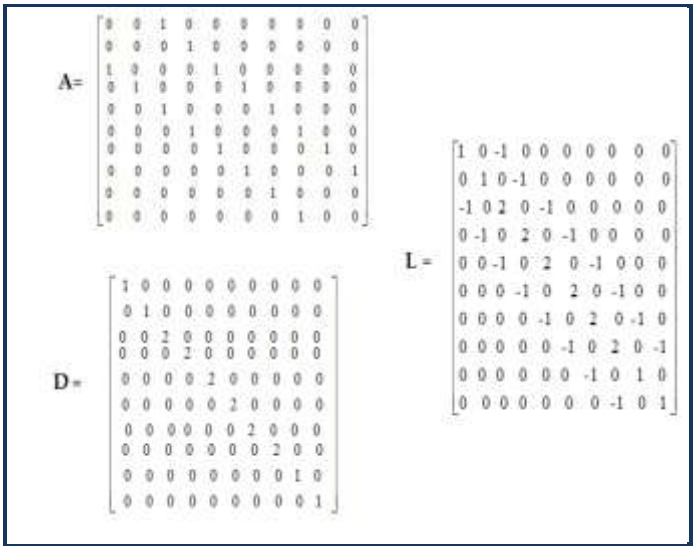

Figure 2: Representation of Adjacency matrix **(A)**, Diagnol matrix **(D)**, and Laplacian matrix **(L)**.

Eigen values and eigenvectors of the Laplacian matrix are computed. S*pectral partitioning* makes use of the spectral value, its corresponding eigen vectors contain all significant topological information about the graph. The eigen values and corresponding eigen vector of the Laplacian matrix is as shown in **Table 1 (see supplementary material).**

The Fiedler vector bisects the graph into two partitions based on the sign of the corresponding vector entry. By repeatedly applying *Spectral partitioning* algorithm, the desired numbers of partitions can be obtained,

Software Input:
The user provides the PDB (Protein Data Bank) file that can either be uploaded from the http://www.rcsb.org (PDB) site or from the local disk. The clustering is performed on activating the '*Spectral Partitioning*' button. The user is prompted to select the threshold atomic distance before partitioning. The '*Cluster Again*' button will result in partitioning the selected partition. This can be continued until the desired number of partition is obtained. The Graphical user interface of *Spec*P tool is specified in **(Figure 3).**

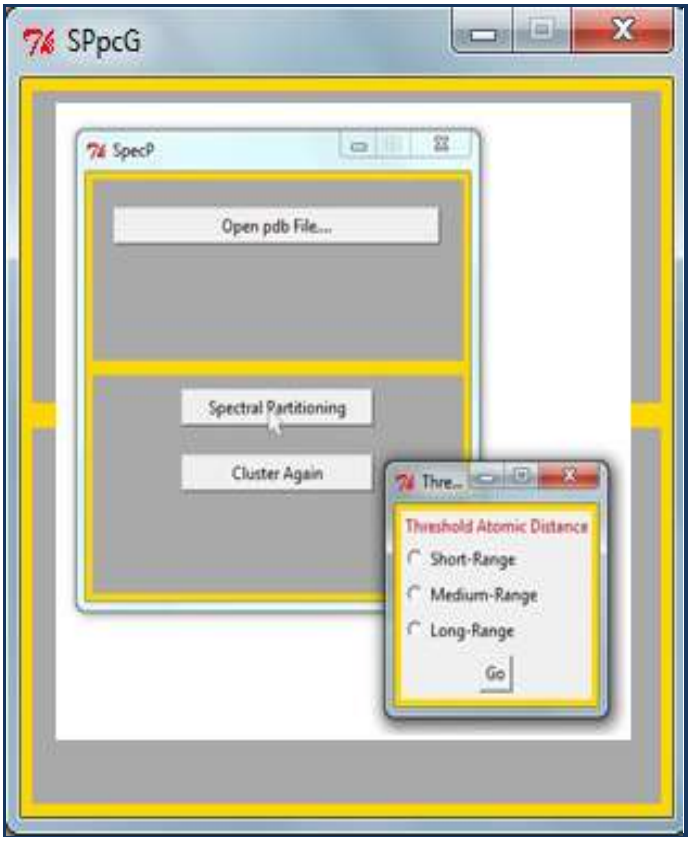

Figure 3: Graphical user interface of ***Spec*P**

Software Output:
Figure 4 (A), 4(B), & 4(C), depicts the screenshot of the first two S*pectral Partitions* of the protein (PDB id 4q21) generated from *Spec*P tool along with its Protein Contact Network and Protein Contact Map.

The 'Cluster Again' option spectrally partitions the selected partition. **(Figure 5),** displays the *Spec*P tool producing 2, 3, 4, 5 partitions of the protein (PDB id4Q21).

Caveat & Future development:
*Spec*P is a stand-alone package developed in Python 2.6 on Windows platform. GUI was designed using Tkinter. Numpy and Scipy modules were used for scientific computing, Igraph, Networkx modules were added to create and manipulate graphs, and to represent and manipulate complex network structures respectively. MatplotLib and Pylab were imported to plot data and generate output in different formats.

Future version will be a web-based application capable of performing spectral partitioning on the basis of surrounding hydrophobicity and build to compute clustering coefficient, cyclic coefficient, characteristic path length, associative coefficient and triangle density from the of the Protein Contact Network generated.

ISSN 0973-2063 (online) 0973-8894 (print)
Bioinformation 9(10): 545-548 (2013)

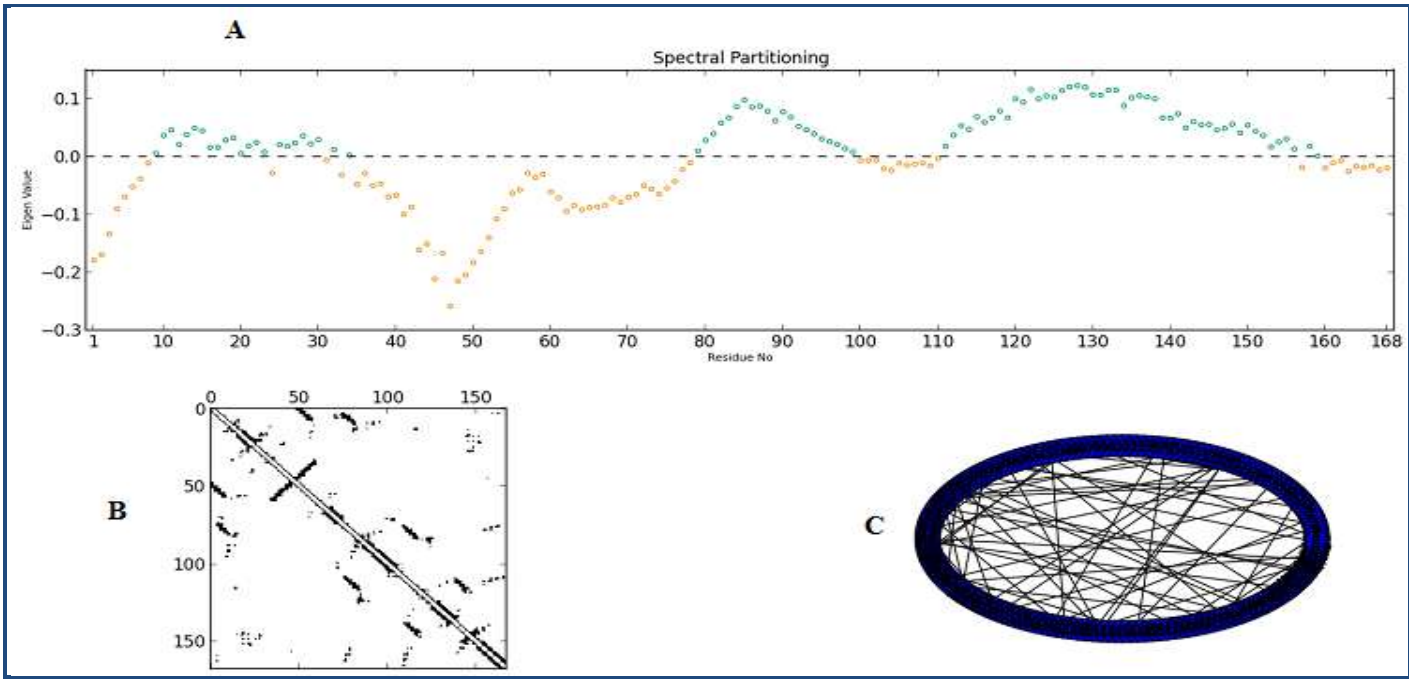

Figure 4: A) Two clusters obtained by spectral partitioning; **B)** The Protein Contact Map; **C)** Protein Contact Network of protein **(**PDB id 4q21).

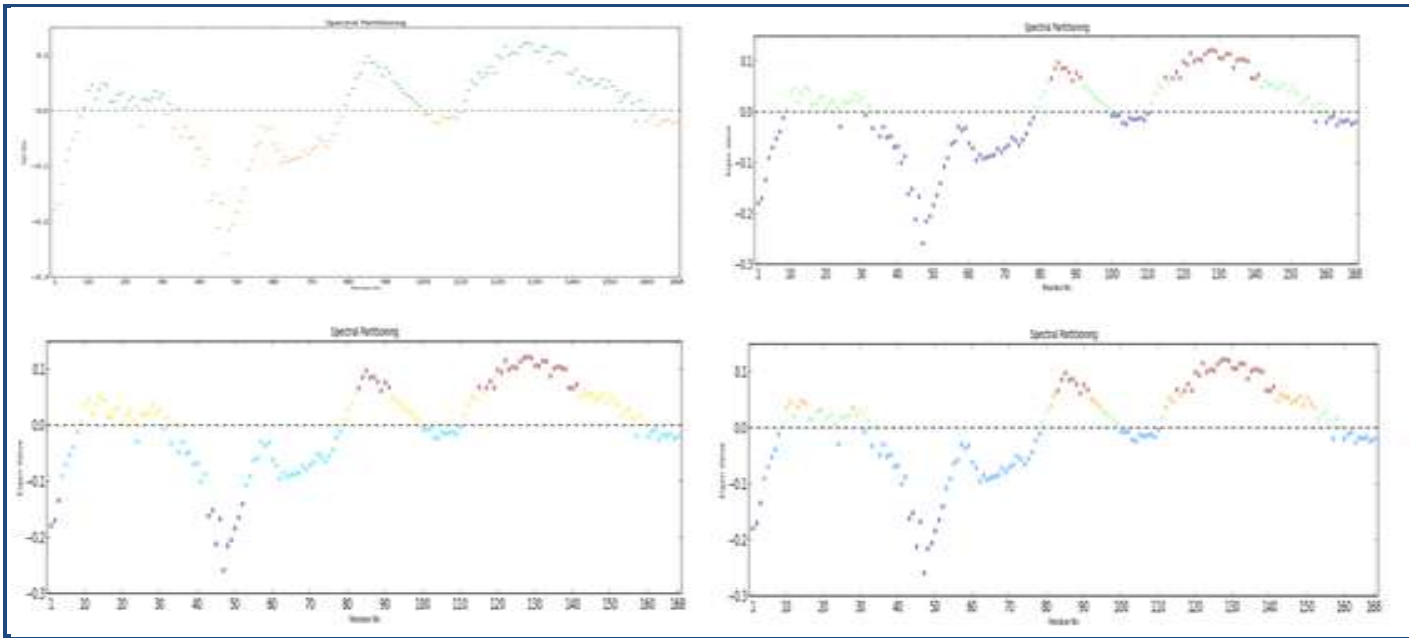

Figure 5: Spectral Partitioning of the protein (PDB id4Q21) into 2, 3, 4, 5 partitions

References:

[1] Franz Rendl *et al. Annals of Operations Research.* 1995 155

[2] Saraswathi Vishveshwara *et al. Journal of Theoretical and Computational Chemistry.* 2002 **1:**1

[3] A. Giuliani *et al. Chem Rev.* 2013 **113:** 1598 [PMID: 23186336]

[4] Hui Kian Ho1 *et al. Bioinformatics.* 2008 24: 2935 [PMID: 18977780]

[5] Newman ME, *Phys Rev E Stat Nonlin Soft Matter Phys.* 2006 **74:** 036104 [PMID: 17025705]

Edited by P Kangueane

Citation: **Namboodiri & Kripadas,** Bioinformation 9(10): 545-548 (2013)

BIOINFORMATION

open access

Supplementary material:

Table 1: The first 10 eigen values and the corresponding Fiedler vector of protein PDB- id 4q21

Eigen values	17.0959	16.602	16.5396	15.8352	15.5813	15.0826	0.000	14.8224	14.7300	14.4600
Nodes	Eig. Vec	Eig. Vec	Eig. Vec	Eig. Vec	Eig. Vec	Eig. Vec	Eig. Vec	Eig. Vec	Eig. Vec	Eig. Vec
1	-0.0001	-0.0011	-0.0004	-0.0016	0.0002	0.0019	0.0772	-0.0001	-0.0044	0.0002
2	0.000	0.0004	0.0001	-0.0008	0.0006	0.0004	0.0772	-0.0016	-0.0071	-0.0074
3	0.0004	0.0022	0.0008	0.0005	0.0038	0.0044	0.0772	-0.0028	0.0035	0.0189
4	-0.0004	-0.0161	-0.0055	-0.0094	0.0002	0.0194	0.0772	0.0121	0.0284	0.0928
5	-0.0008	0.0451	0.0179	0.1222	-0.0786	-0.1282	0.0772	0.0952	0.3302	0.0933
6	0.0102	0.2531	0.0947	0.2716	-0.0408	-0.1346	0.0772	0.0748	0.3378	0.127
7	-0.0152	0.4127	0.1406	0.2995	-0.157	0.0347	0.0772	-0.1198	-0.216	-0.088
8	-0.0903	0.2203	0.0579	-0.1443	-0.0748	0.0325	0.0772	0.0888	-0.1822	-0.1265
9	-0.0822	0.113	0.0258	-0.1724	-0.0051	-0.0097	0.0772	0.1206	-0.0347	-0.0327
10	-0.1317	-0.0354	0.0035	-0.0485	0.0405	-0.1154	0.0772	0.0724	0.1016	0.0796