Mathematical and Computer Modelling 52 (2010) 2066–2074

Contents lists available at ScienceDirect

Mathematical and Computer Modelling

journal homepage: www.elsevier.com/locate/mcm

Spectral clustering for detecting protein complexes in protein–protein interaction (PPI) networks

Guimin Qin [a,b], Lin Gao [a,*]

[a] School of Computer Science and Technology, Xidian University, Xi'an, Shaanxi 710071, China

[b] Software School, Xidian University, Xi'an, Shaanxi 710071, China

ARTICLE INFO

Article history:
Received 23 September 2009
Received in revised form 25 December 2009
Accepted 31 January 2010

Keywords:
Spectral clustering
Protein complex
PPI networks

ABSTRACT

In this paper, we study spectral clustering for detecting protein complexes in PPI (protein–protein interaction) networks, focusing on two open issues: (i) constructing similarity graphs; and (ii) determining the number of clusters. First, we study four similarity graphs to construct graph Laplacian matrices. Then we propose a method to determine the number of clusters based on the properties of PPI networks. Experimental results on PPI networks from DIP data and MIPS data indicate that each similarity graph shows its strengths and disadvantages, and our finding of the number of clusters improves the clustering quality. Finally, spectral clustering obtains results in detecting protein complexes that are comparable to those obtained from several other typical algorithms.

1. Introduction

Protein–protein interactions (PPIs) play a key role in the assembly of the cell's structural components. They are also crucial for processes ranging from transcription, splicing, and translation to cell cycle control, secretion, and the assembly of enzymatic complexes [1]. Many high-throughput techniques have been used to create PPI networks, making PPI data available for further analysis, although there are false-positive and false-negative instances in the data. It is essential to extract functional modules such as protein complexes or regulatory pathways from global interaction networks [2].

To achieve this goal, many clustering methods, such as MCODE [3], MCL [4], CPA [5] and DECAFF [6], have been applied to the protein interactome graph. These algorithms require specifying parameters, some of which may drastically affect the results. Several comparisons have been made to evaluate the methods [2,7,8], which help biologists choose suitable algorithms to analyze PPI networks.

As effective clustering techniques, spectral clustering methods arise from concepts in spectral graph theory. The basic idea is to construct a similarity graph from the initial data set where each vertex represents a data object, and each weighted edge simply represents the similarity between two objects. Then a Laplacian matrix for this similarity graph is created, and the input space is mapped to the eigenvector space of the similarity graph. Finally, a clustering algorithm is used to cluster the data in the eigenvector space. Spectral methods have been developed effectively for solving a number of graph clustering objectives, including ratio cut and normalized cut. Spectral clustering has been used in image segmentation [9–11], speech separation, circuit layout and complex network decomposition [12]. However, we have not seen anyone apply spectral clustering for PPI network decomposition.

In this study, we applied a spectral clustering method to detect protein complexes in PPI networks, and evaluated the results. In Section 2, we briefly introduce the concepts of spectral graph theory, including graph theory, similarity graphs, and the graph Laplacian. In Section 3, we study four similarity graphs for PPI networks, a method for determining the number

* Corresponding author.
E-mail addresses: gmqin@mail.xidian.edu.cn (G. Qin), lgao@mail.xidian.edu.cn (L. Gao).

0895-7177/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.mcm.2010.06.015

of complexes, spectral clustering, and evaluation criteria. Section 4 shows the experimental results. Finally, we give our conclusion and suggest future work.

2. Spectral graph theory

2.1. Graph notations

Let $G = (V, E)$ be an unweighted, undirected simple graph with vertex set $V = \{v_1, v_2, \ldots, v_n\}$. The adjacency matrix of graph G is the matrix $A = \{a_{ij}\}, i, j = 1, 2, \ldots, n$. $a_{ij} = 1$, if there is an edge between vertex v_i and v_j, and $a_{ij} = 0$ otherwise. For an undirected graph, A is a symmetry matrix, i.e. $a_{ij} = a_{ji}$. The degree of a vertex $v_i \in V$ is defined as $d_i = \sum_{j=1}^{n} a_{ij}$.

The degree matrix D is defined as the diagonal matrix with the degrees $d_1, d_2, \ldots, d_n$. Given a subset of vertices $V_1 \subset V$, we denote its complement $V \setminus V_1$ by $\overline{V_1}$. Two ways of measuring the "size" of a subset $V_1 \subset V$ are $|V_1| =$ the number of vertices in V_1 and $vol(V_1) = \sum_{v_i \in V_1} d_i$.

2.2. Similarity graphs

There are several popular construction methods to transform a given set $x_1, x_2, \ldots, x_n$ of data points with pairwise similarities s_{ij} or pairwise distance d_{ij} into a graph G, which consist of the ε-neighborhood graph, k-nearest neighbor graph, and the fully connected graph [13]. A similarity graph is a weighted undirected graph, with weighted matrix S. The degree matrix D defined on S is the diagonal matrix with the degrees $d_1, d_2, \ldots, d_n$, where $d_i = \sum_{j=1}^{n} s_{ij}$, and $vol(V_1) = \sum_{v_i \in V_1} d_i$.

For network data, we can also construct similarity graphs according to these approaches. Besides, several other construction methods have been studied to transform a network to a similarity graph, which we will discuss in Section 3.

2.3. Graph Laplacians and their basic properties

The main tools for spectral clustering are graph Laplacian matrices. There are three forms of graph Laplacian, defined as follows [13],

$$L = D - S \tag{1}$$

$$L_{sys} = D^{-1/2} L D^{-1/2} = I - D^{-1/2} S D^{-1/2} \tag{2}$$

$$L_{rw} = D^{-1} L = I - D^{-1} S. \tag{3}$$

Eqs. (1)–(3) define unnormalized, normalized symmetric, and normalized random walk graph Laplacians, respectively. These graph Laplacians hold some useful properties for spectral clustering, such as zero eigenvalue, semi-definite matrix, and so on. For more details about graph Laplacians and spectral graph theory, refer to [13], [14].

3. Spectral clustering for PPI networks

3.1. Similarity graphs for PPI networks

A complex network (i.e. a graph) is a special kind of data structure for clustering where a node (i.e. a vertex) represents a data point and a link (i.e. an edge) between two nodes represents their similarity relationship. Suppose an adjacency matrix of a network is given. We discuss four typical similarity graphs for a network, which are based on the adjacency matrix (adjacency similarity, **A**), on common neighbors (common neighbor similarity, **J**), on vertex similarity transmission (transmission similarity, **V**) and on the commute distance (commute similarity, **C**).

3.1.1. Adjacency similarity **A**

The obvious way to construct similarity graph is to use the adjacency matrix A directly. This is also the most popular method that spectral clustering uses for complex networks. It is simple to construct and understand, since two vertices are similar if they are connected to each other, dissimilar if there is no edge between them. However, there are disadvantages. Taking Fig. 1 as an example, the vertex-pair 8–9 (i.e., the two vertices) has the same similarity as vertex-pair 9–10; in fact, we want to assign more similarity value on vertex-pair 8–9 than that on 9–10. Vertex-pairs 1–6, 1–17, and 17–19 are all dissimilar. However, to a certain degree, we cannot catch this information from this method.

3.1.2. Common neighbor similarity **J**

The basic idea is that two vertices are considered structurally equivalent if they share many of the same network neighbors. It seems reasonable that two individuals in a social network have something in common if they share many of the same friends. Let Γ_i be the neighborhood of vertex i in a network, i.e., the set of vertices that are directly connected to i via an edge. Ref. [15] lists several ways of measuring a common neighborhood, and here we focus on Eq. (4) [15].

$$\sigma_{Jaccard} = \frac{|\Gamma_i \cap \Gamma_j|}{|\Gamma_i \cup \Gamma_j|}. \tag{4}$$

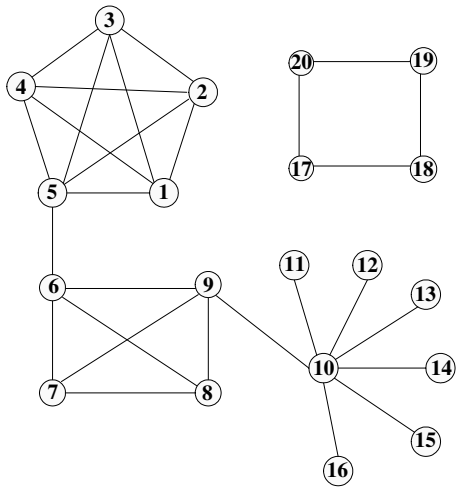

Fig. 1. A sample network.

These ways are reasonable in some way. However, there are also some weak points. Also taking Fig. 1 as an example, vertex-pairs 10–11 and 17–18 have the same similarity value $\sigma_{jaccard}$ of 0, and 11–12, 19–20 have similarity values $\sigma_{jaccard}$ of 0.5 and 1.0, respectively (and 11–12, 18–20 have the same similarity value $\sigma_{jaccard}$ of 1.0). The results are not what we want. Based on this analysis, we rewrite Eq. (4) as Eq. (5), and add a weight 1 to every edge.

$$\sigma = \begin{cases} 1 + \sigma_{Jaccard} & \text{if } (i, j) \in E \\ \sigma_{Jaccard} & \text{otherwise.} \end{cases} \tag{5}$$

3.1.3. Transmission similarity **V**

This type of measure is based on the idea that two vertices are similar if they are connected to other vertices that are themselves similar. Several such algorithms have been proposed, such as in [15]. Here we focus on Newman's method, where the similarity between vertex i and j is defined as

$$S_{ij} = \phi \sum_{v} A_{iv} S_{vj} + \psi \delta_{ij}, \tag{6}$$

where ϕ and ψ are parameters, and A_{iv} is the element of the adjacency matrix A between vertices i and v. This gives good results for Fig. 1, but how to determine the parameters for certain networks is a key problem.

3.1.4. Commute similarity **C**

The commute distance is defined in random walk theory [13,16]. The commute distance c_{ij} between two vertices i and j is the expected time it takes the random walk to travel from vertex i to vertex j and back. The commute distance between two vertices decreases if there many different short paths between them. However, the commute distance only applies to connected networks, so it is not suitable for unconnected networks like Fig. 1.

3.2. The number of complexes in PPI networks

How to determine the number of clusters is a key problem in all clustering applications, and there is no common effective method to obtain the value. For PPI networks which are scale-free networks, there are only a few highly connected proteins, namely hub proteins, and most proteins have small degrees. Maslov et al. further found [17] that the connection between two highly connected proteins (i.e. the hub) is suppressed, and most interactions exist between highly connected and lowly connected proteins. This effect decreases the likelihood of cross talk between different functional modules of the cell, and increases the overall robustness of a network by localizing the effects of deleterious perturbations. Based on this background, we determined the number of complexes by the number of the hub vertices in PPI networks.

Definition 1 (*Hub Vertex*)**.** Given graph $G(V, E)$, a vertex $v \in V$ is a hub vertex if it satisfies the following conditions.

(1) $degree(v) \geq avg - degree(G)$.
(2) The degree of vertex v is larger than that of most its neighbors.

Fig. 2 is a scale-free network, and every two highly connected nodes are not connected with each other directly. In this network, nodes 1, 5, and 13 are hub vertices. After clustering, we expect the hub vertices and their corresponding surrounding nodes to constitute clusters.

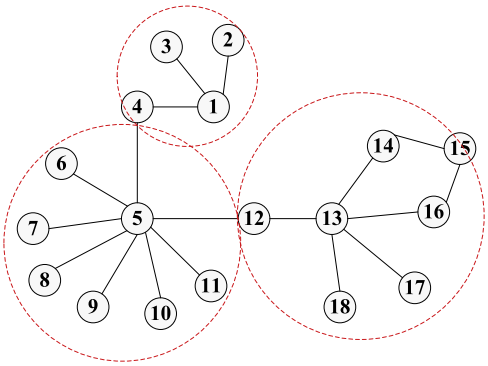

Fig. 2. A scale-free network.

For PPI networks, we determine the number of clusters k based on biological knowledge and the properties of scale-free networks. First, according to reference complexes and PPI networks, there are many two-protein complexes and three-protein complexes, and there are two-vertex and three-vertex components in PPI networks. We deal with this information independently. In this step, we find k_1 clusters. Second, we determine the number of hub vertices k_2 in PPI networks. Finally, we obtain the total number of clusters $k = k_1 + k_2$.

3.3. Spectral clustering

We apply a spectral clustering algorithm based on L_{rw} [11,13] which proceeds as follows.

Input: adjacency matrix $A \in \{0, 1\}_{n\times n}$ of graph G.
Procedure:

1. Construct a similarity graph by one of the ways described in the previous section. Let S be the similarity matrix.
2. Compute the Laplacian L.
3. Determine the number of clusters k according to the previous section.
4. Compute the first k eigenvectors $v_1, v_2, \ldots, v_k$ of the generalized eigenproblem $Lv = \lambda Dv$.
5. Let $V \in R^{n\times k}$ be the matrix containing the vectors $v_1, v_2, \ldots, v_k$ as columns.
6. For $i = 1, 2, \ldots, n$, let $y_i \in R^k$ be the vector corresponding to the i-th row of V.
7. Cluster the points $(y_i)_{i=1,2,\ldots,n}$ in R^k with the *k-means* algorithm into cluster $C_1, C_2, \ldots, C_k$.

Output: Clusters $A_1, A_2, \ldots, A_k$ with $A_i = \{j|y_j \in C_i\}$.

3.4. Evaluation criteria

Different criteria proposed by previous studies are used to evaluate spectral clustering for PPI decomposition. The evaluation criteria [2,8] are defined as follows.

3.4.1. F-measure

Let RC and CC be the real complexes set and the clustering results set, respectively. The neighborhood affinity between a real complex b and a predicted complex p defined in Eq. (7) is used to measure the similarity between b and p. The neighborhood affinity score between b and p is defined as Eq. (8), and a similar definition as Eq. (9).

$$NA(p, b) = \frac{|V_p \cap V_b|^2}{|V_p| \times |V_b|} \tag{7}$$

$$N_{cp} = |\{p|p \in CC, \exists b \in B, NA(p, b) \geq \omega\}| \tag{8}$$

$$N_{cb} = |\{b|b \in RC, \exists p \in P, NA(p, b) \geq \omega\}| \tag{9}$$

where N_b, N_p, B, P and ω are the number of proteins in b, the number of proteins in p, the number of real complexes, the number of predicted complexes and a parameter, respectively. In [8], the parameter ω is set as 0.2. In our experiments, we also set ω as 0.2.

Based on these definitions, the precision, recall, and F-measure are defined as Eqs. (10)–(12), respectively.

$$Precision = \frac{N_{cp}}{|CC|} \tag{10}$$

$$Recall = \frac{N_{cb}}{|RC|} \tag{11}$$

$$F = 2 \times Precision \times Recall/(Precision + Recall). \tag{12}$$

3.4.2. Accuracy

The sensitivity, positive predictive value (PPV), and accuracy are used to evaluate the accuracy of the prediction. Let n and m be the number of real complexes and predicted complexes, respectively. And T_{ij} be the number of proteins in common between the ith real complex and the jth predicted complex. Then, the sensitivity, PPV, and accuracy are defined as Eqs. (13)–(15), respectively.

$$Sn = \frac{\sum_{i=1}^{n} \max_j \{T_{ij}\}}{\sum_{i=1}^{n} N_i} \tag{13}$$

$$PPV = \frac{\sum_{j=1}^{m} \max_i \{T_{ij}\}}{\sum_{j=1}^{m} T_{.j}} \tag{14}$$

$$Acc = \sqrt{Sn \times PPV}. \tag{15}$$

3.4.3. Separation

Separation is proposed to measure the one-to-one correspondence between a predicted complex and a real complex. The separation of the ith real complex and the jth predicted complex is defined as Eq. (16). The complex-wise separation, cluster-wise separation, and separation are defined as Eqs. (17)–(19), respectively.

$$sep_{ij} = \frac{T_{ij}}{\sum_{i=1}^{n} T_{ij}} \times \frac{T_{ij}}{\sum_{j=1}^{m} T_{ij}} \tag{16}$$

$$sep_b = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} sep_{ij}}{n} \tag{17}$$

$$sep_p = \frac{\sum_{j=1}^{m} \sum_{i=1}^{n} sep_{ij}}{m} \tag{18}$$

$$sep = \sqrt{sep_b \times sep_p}. \tag{19}$$

4. Experimental results

We performed a number of experiments to evaluate spectral clustering for protein complex detection in PPI networks.

4.1. Datasets

In our experiments, we use DIP data [18] and MIPS data [19] for protein complex detection. The DIP data (released on 12/02/2007) consist of 4928 proteins and 17 201 interactions, and the MIPS data consist of 4545 proteins and 12 316 interactions. The protein complexes collected by [20] are used as reference data, which consist of 428 complexes.

4.2. Evaluation results

4.2.1. Similarity graphs evaluation

We first compare different similarity graphs. Since there are 428 real complexes, the number of clusters ranges from 400 to 500. In the figures (from Fig. 3 to Fig. 8), **A**, **J**, **V**, and **C** stand for the four types of similarity graph, i.e. adjacency similarity, common neighbor similarity, transmission similarity and commute similarity, respectively. Fig. 3 shows the F-measure values of the four similarity graphs in the DIP PPI network; the x-axis represents the number of predicted complexes. From Fig. 3, the value is greatest for common neighbor similarity and is progressively less for adjacency similarity, transmission similarity, and commute similarity. Furthermore, the former two methods are much better than the latter two. Fig. 4 presents the accuracy values for the four similarity graphs in the DIP PPI network. The accuracy value is greatest for adjacency similarity and progressively decreases for common neighbor similarity, transmission similarity, and commute similarity.

Figs. 5 and 6 show F-measure values and accuracy values of the four similarity graphs in the MIPS data. From Fig. 5, the broken lines are intertwined, so no similarity graph is better than the others. From Fig. 6, the accuracy values decrease in

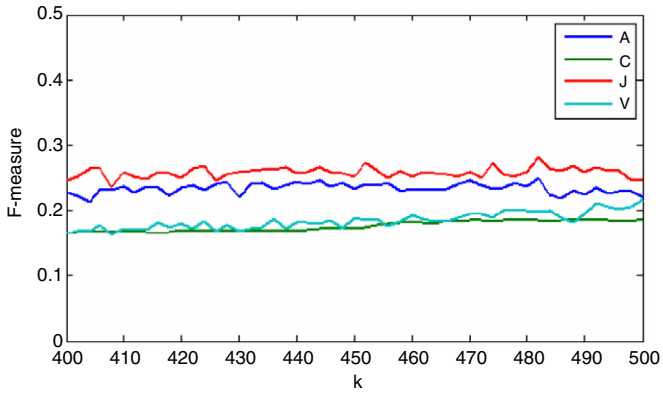

Fig. 3. *F*-measure comparison for the four similarity graphs in the DIP data.

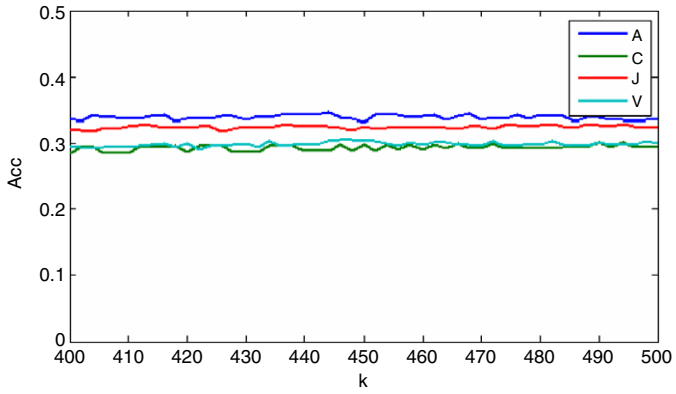

Fig. 4. Accuracy comparison for the four similarity graphs in the DIP data.

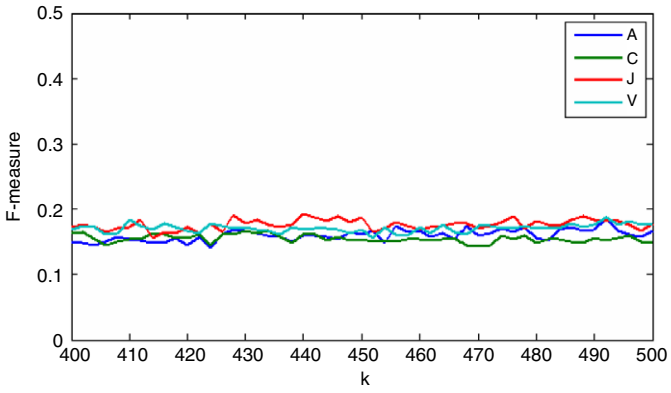

Fig. 5. *F*-measure comparison for the four similarity graphs in the MIPS data.

the order of commute similarity, transmission similarity, adjacency similarity, and common neighbor similarity. However, the differences are very slight.

From the above results, the *F*-measure and accuracy values are fairy low, less than 0.4. Furthermore, different topological networks affect the clustering results of the four similarity graphs.

4.2.2. Similarity graphs and the k-value evaluations

Using our method to determine the number of predicted complexes k, we get 512 clusters and 351 clusters in the DIP data and MIPS data, respectively. We run spectral clustering on the four similarity graphs. Figs. 7 and 8 illustrate the evaluation criteria (*F*-measure, accuracy and separation) of the four similarity graphs and the k values. From Figs. 7 and 8, an improvement is obtained for the k values.

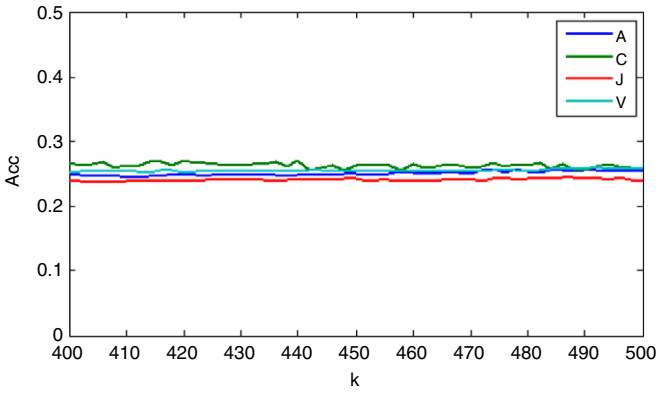

Fig. 6. Accuracy comparison for the four similarity graphs in the MIPS data.

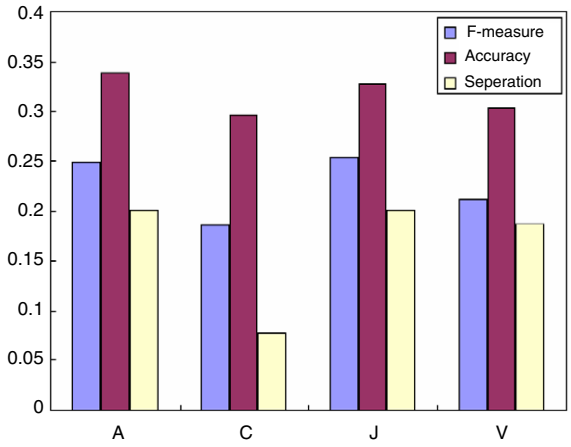

Fig. 7. Comparison for the four similarity graphs in the DIP data.

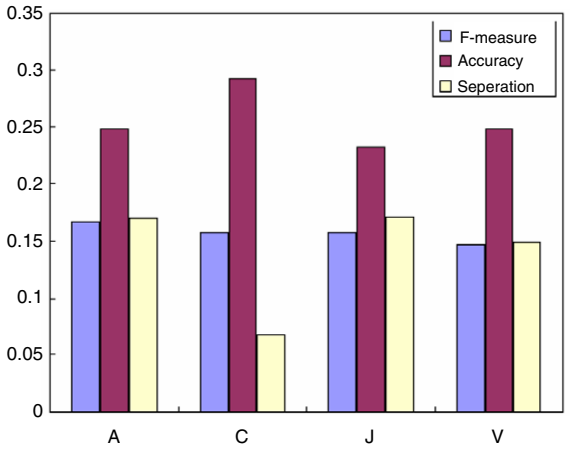

Fig. 8. Comparison for the four similarity graphs in the MIPS data.

In Fig. 7, the F-measure value of common neighbor similarity is maximum, at 0.254, then that of adjacency similarity (0.249), transmission similarity (0.212), and commute similarity (0.185). The accuracy value is greatest for adjacency similarity and progressively decreases for common neighbor similarity, transmission similarity, and commute similarity. From the results, common neighbor similarity and adjacency similarity are suitable for constructing similarity graphs for the DIP PPI data.

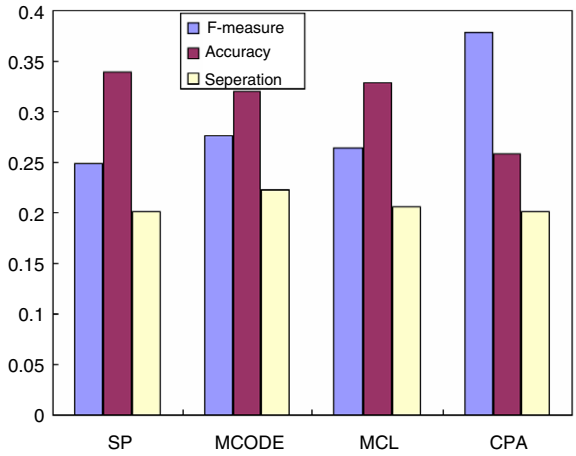

Fig. 9. Comparison for the four algorithms in the DIP data.

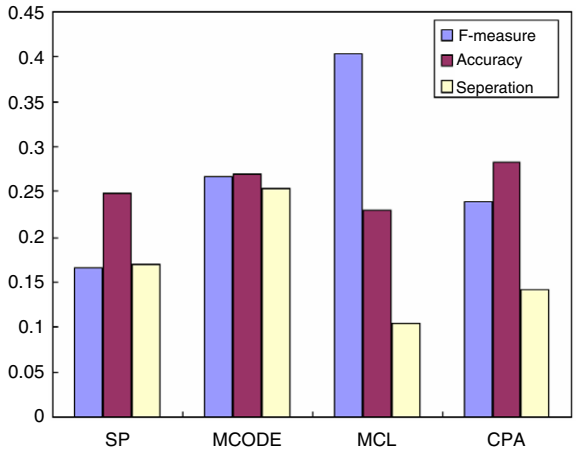

Fig. 10. Comparison for the four algorithms in the MIPS data.

From Fig. 8, the *F*-measure values of adjacency similarity, common neighbor similarity, commute similarity and transmission similarity are 0.166, 0.157, 0.157, and 0.146, respectively. Compared with the other three similarities, the accuracy value of commute similarity is maximum, at 0.292, while the separation value is minimum, at only 0.067. There are only marginal differences among adjacency similarity, common neighbor similarity, and transmission similarity for three evaluation criteria, so these three similarities are used to construct the similarity for the MIPS PPI data.

4.2.3. Comparison with other clustering algorithms

Finally, we compare spectral clustering based on adjacency similarity with three representative complex detection algorithms, namely MCODE [3], MCL [4], and CPA [5]. Figs. 9 and 10 give the comparison results of these algorithms in the DIP and MIPS data, respectively. In the figures, **SP** stands for spectral clustering. The results of MCODE, MCL, and CPA are from [8]. From the figures, no algorithm is better than the other algorithms for all criteria. For the DIP PPI data, spectral clustering gains the minimum *F*-measure value, which means that the predicted complexes detected by spectral clustering do not match the reference complexes better than those complexes detected by other algorithms. But spectral clustering gains the greatest accuracy value. Furthermore, there is less one-to-one correspondence between a real complex and a predicted one found by spectral clustering than by the other algorithms, since the separation value of spectral clustering is minimum.

4.3. Discussion

To evaluate the four similarity graphs, we performed a series of experiments on the DIP and MIPS data, ranging the number of complexes from 400 to 500. Low values of evaluation criteria may partly come from the poor quality of the PPI data, since there are many false-positive and false-negative interactions. By analyzing the ideas behind the similarity graphs, adjacency similarity only considers proteins' direct neighbors, and common neighbor similarity not only considers proteins'

direct interactions, but also two proteins' common neighbors. While transmission similarity extends the ideas, it takes all connected proteins into account. It seems to be more reasonable than the first two methods. However, it gives a poorer result in the DIP data. Commute similarity is based on random walk theory. Commute similarity is not suitable for disconnected networks; therefore we only calculate commute similarity on the main component of PPI networks. The method also gives poor results.

We used our k values to evaluate the four similarity graphs for the DIP and MIPS PPI data, and obtained better results. From all these results, we found that different topologies of networks should adopt appropriate type of similarity graphs, and our method of determining the number of clusters is effective.

We also compared spectral clustering with other clustering algorithms, namely MCODE, MCL, and CPA. These algorithms are famous for detecting complexes. Spectral clustering shows comparable results, which proves that spectral clustering is good for PPI network decomposition. Furthermore, a suitable pre-processing or post-processing may help improve the clustering results.

5. Conclusions

While many algorithms for the detection of protein complexes in PPI networks have been proposed in recent years, different algorithms have their own strengths and disadvantages. In this study, we adopted spectral clustering for PPI network decomposition. We first studied four similarity graphs and proposed a method to determine the number of complexes based on the properties of PPI networks. Then we performed experiments to evaluate spectral clustering for PPI complex detection.

Since there is noise in PPI data, we will do a pre-process step before clustering. Furthermore, PPI networks have many special properties, so we will define a new similarity graph to obtain higher clustering quality.

Acknowledgements

This work was supported by the National Key Natural Science Foundation of China (Grant No. 60933009), Specialized Research Fund for the Doctoral Program of Higher Education (Grant No. 200807010013) and the National Natural Science Foundation of China (Grant No. 60970065).

References

[1] Bjorn H. Junker, Falk Schreiber, Analysis for Biological Networks, John Wiley and Sons, New York, 2008.
[2] Sylvain Brohee, Jacques van Helden, Evaluation of clustering algorithms for protein–protein interaction networks, BMC Bioinformatics 7 (2006) 488.
[3] G. Bader, C. Hogue, An automated method for finding molecular complexes in large protein interaction networks, BMC Bioinformatics 4 (2003) 2.
[4] S. Van Dongen, Graph clustering by flow simulation, Ph.D. Thesis, University of Utrecht, Netherlands, May, 2000.
[5] M. Amin, Y. Shinbo, K. Kurokawa, S. Kanaya, Development and implementation of an algorithm for detection of protein complexes in large interaction networks, BMC Bioinformactics 7 (2006) 207.
[6] X.L. Li, C.S. Foo, S.K. Ng, Discovering protein complexes in dense reliable neighborhoods of protein interaction networks, in: CSB'07, 2007, pp. 157–168.
[7] Jim Vlasblom, Shoshana J. Wodak, Markov clustering versus affinity propagation for the partitioning of protein interaction graphs, BMC Bioinfomatics 10 (2009) 99.
[8] Min Wu, Xiaoli Li, Chee-Keong Kwoh, Algorithms for detecting protein complexes in PPI networks: an evaluation study, in: PRIB08, 2008, pp. 135–146.
[9] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8) (2000) 888–901.
[10] Andrew Y. Ng, Michael I. Jordan, Yair Weiss, On spectral clustering: analysis and an algorithm, in: NIPS'01, 2001.
[11] Stella X. Yu, Jianbo Shi, Multiclass spectral clustering, in: ICCV'03, 2003.
[12] M.E. Newman, Finding community structure in networks using the eigenvectors of matrices, Physical Review E 74 (3) (2006) 036104.
[13] Ulrike von Luxburg, A tutorial on spectral clustering, Max Planck Institute for Biological Cybernetics, Tec. Rep. 149, 2006.
[14] Fan R.K. Chung, Spectral Graph Theory, Washington, Conference Board of the Mathematical Sciences, 1997.
[15] E.A. Leicht, Petter Holme, M.E.J. Newman, Vertex similarity in networks, Physics Review E 73 (2006) 026120.
[16] François Fouss, Alain Pirotte, Jean-Michel Renders, Marco Saerens, Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation, IEEE Transactions on Knowledge and Data Engineering 19 (3) (2007) 355–369.
[17] S. Maslov, K. Sneppen, Specificity and stability in topology of protein networks, Science 296 (2002) 910–913.
[18] 2007 DIP database. [Online]. Available: http://dip.doe-mbi.ucla.edu.
[19] MIPS database. [Online]. Available: ftp://ftpmips.gsf.de/yeast/PPI/PPI_18052006.tab.
[20] C.C. Friedel, J. Krumsiek, R. Zimmer, Bootstrapping the interactome: unsupervised identification of protein complexes in yeast, in: RECOMB'08, 2008, pp. 3–16.