Predicting protein complexes via the integration of multiple biological information

Xiwei Tang[1,3], Jianxin Wang*[1], Yi Pan[1,2]
1. School of Information Science and Engineering, Central South University, Changsha, 410083, China
2. Department of Computer Science, Georgia State University, Atlanta, GA 30302-4110, USA
3. School of Information Science and Engineering, Hunan First Normal University, Changsha, 410205, China
Email: *corresponding author, jxwang@mail.csu.edu.cn; tangxiwei2010@gmail.com; pan@cs.gsu.edu

***Abstract*—Protein complexes are a cornerstone of many biological processes and together they form various types of molecular machinery that perform a vast array of biological functions. An increase in the amount of protein-protein interaction (PPI) data enables a number of computational methods for predicting protein complexes. There are a mass of algorithms detecting complexes only consider the PPI data. However, the PPI data from high-throughout techniques is flooded with false interactions. In fact, the insufficiency of the PPI data significantly lowers the accuracy of these methods.**

In the current work, we develop a novel method named CMBI to discover protein complexes via the integration of multiple biological resources including gene expression profiles, essential protein information and PPI data. First, CMBI defines the functional similarity of each pair of interacting proteins based on the edge-clustering coefficient (ECC) from the PPI network and the Pearson correlation coefficient (PCC) from the gene expression data. Second, CMBI selects essential proteins as seeds to build the protein complex cores. During the growth process, the seeds' essential protein neighbors and the neighbors whose functional similarity (FS) with the seeds are more than the threshold T will be added to the complex cores. After the complex cores are constructed, CMBI begins to generate protein complexes by attaching their direct neighbors with $FS > T$ to the cores. In addition to the essential proteins, CMBI also uses other proteins as seeds to expand protein complexes. To check the performance of CMBI, we compare the complexes discovered by CMBI with the ones found by other techniques by matching the predicted complexes against the reference complexes. We use subsequently GO::TermFinder to analyze the complexes predicted by various methods. Finally, the effect of parameter T is investigated.

The results from GO functional enrichment and matching analyses show that CMBI performs significantly better than the state-of-the-art methods. It means that it's successful for us to integrate multiple biological information to identify protein complexes in the PPI network.

I. Introduction

Protein complexes are a cornerstone of many biological processes and together they form various types of molecular machinery that perform a vast array of biological functions. They are assemblies of proteins, which form many interactions with each other and therefore are cohesive and strongly connected to each other in the context of the larger protein interaction network. In the post genomic era, one of the most challenging tasks is to predict protein complexes from protein-protein interaction network. Many research groups previously used experimental methods to detect protein complexes. However, there are some experimental limitations in these methods. So researchers begin to develop computational approaches for detecting protein complexes.

Recent developments in biotechnology have resulted in an increase in the amount of protein-protein interaction (PPI) data. Pair-wise protein interactions can be modeled as a graph, where vertices are proteins and edges are protein-protein interaction (PPI). Protein complexes correspond to the dense subgraphs of the initial graph [1]. Over the past decade, a wealth of graph clustering algorithms have been proposed and applied to the identification of highly connected nodes in protein interaction graphs [2]–[9]. These methods work well on PPI networks and extracted successfully protein complexes. Nevertheless it has been noticed that protein interaction data produced by high-throughput experiments are often associated with high false positive due to the limitations of the associated experimental techniques, which may have a negative impact on the complex discovery algorithms.

In order to address that particular question, recent studies concentrate on incorporating gene-expression data to help identify protein complexes in PPI network. In fact, interacting proteins are likely to exhibit similar gene-expression profiles. Studies have shown that genes showing a similar pattern of expression tend to have similar function (guilt by association) [10], [11]. In this direction, many approaches have be proposed [12], [13]. However, there are cases where functionally related genes show dissimilar expression profiles or are inversely co-regulated [14]. Therefore, in order to design a more effective complex discovery algorithm, it is necessary to adopt other biological information.

The work described in this paper aims to detect protein complexes in the PPI network by considering multiple biological information such as essential proteins, gene expression profiles and protein complex's inherent organization. To accomplish this goal, a simple algorithm called CMBI (Clustering based on Multiple Biological Information) is developed. Specifically, CMBI first defines the functional similarity of two interacting proteins in the PPI network by combining the edge clustering coefficient (ECC) [15] between the two proteins and the Pearson correlation coefficient (PCC) [16] for the coexpression profiles of pair of genes coding the two proteins. The essential proteins in the PPI network are subsequently selected as

978-1-4673-4398-5/12/$31.00 ©2012 IEEE

'seeds' to grow the complex cores in terms of their essential proteins neighbors or functionally similar neighbors. After the protein complex cores are produced, CMBI builds the protein complexes by including the cores' functionally similar neighbors into the complex cores. Moreover, CMBI also uses other proteins as seeds to generate protein complexes by virtue of the protein complexes' inherent organizations [17].

II. METHOD

A. Preliminaries

The edge clustering coefficient [15] is a measure of degree to which edges in a graph tend to cluster together. It can be defined as

$$ECC(x,\ y) = \frac{Z^{(3)}_{x,\ y}}{\min(k_x - 1, k_y - 1)} \tag{1}$$

where $Z^{(3)}_{x,\ y}$ denotes the number of triangles that include the edge actually in the network, k_x and k_y are degrees of node x and node y, respectively. Then, the meaning of $min(k_x - 1, k_y - 1)$ is the number of triangles in which the edge $ECC(x,\ y)$ may possibly participate at most. ECC gives a value between 1 and 0 inclusive. PCC [16] is a frequently used coefficient to express similarity between two gene expression profiles. For two sequences of gene expressions such as $X = (x_1, \cdots, x_n)$ and $Y = (y_1, \cdots, y_n)$, PCC is estimated by

$$PCC(x,\ y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x_i})(y_i - \bar{y_i})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x_i})^2 \sum_{i=1}^{n}(y_i - \bar{y_i})^2}} \tag{2}$$

where $\bar{x_i}$ and $\bar{y_i}$ are the average expression values of gene X and Y, respectively. The value of PCC is always between minus one and plus one. We propose a new functional similarity (FS) between two interacting proteins by means of the combination of ECC and PCC:

$$FS(x,\ y) = ECC(x,\ y) + PCC(x,\ y) \tag{3}$$

It can be found that $FS(x, y)$ is more than -1 and less than 2.

Given a PPI network, the goal of our algorithm is to output a set of dense subgraphs. We model the network as a undirected graph $G = (V,\ E)$, in which a vertex in vertex set V represents a protein and an edge in edge set E represents an interaction between two distinct proteins. The degree of a vertex $v \in V$ is the number of v's neighbors in G, written as $deg(v)$. We define the $\underline{s}$econdary-level $\underline{d}$egree of a vertex $v \in V$ as follows:

$$sdeg(v) = deg(v) + \sum_{i=1}^{k} deg(u_i) \tag{4}$$

where u is the neighbor of vertex v, k is the number of v's neighbors. The density of G, denoted as $den(G)$, is defined as follows:

$$den(G) = \frac{2 \times |E|}{|E| \times (|E| - 1)} \tag{5}$$

For a vertex $v \in V$, the neighborhood graph of v contains v, all its neighbors and the edges among them. It is defined as $G_v = (V',\ E')$, where $V' = \{v\} \bigcup \{u | u \in V,\ (u,\ v) \in E\}$, and $E' = \{(u_i,\ u_j) \in E,\ u_i,\ u_j \in V'\}$. Besides, we define the neighborhood of a complex core $C = (V_C,\ E_C)$ as $N(C) = \{u | (u,\ v) \in E,\ v \in V_C,\ u \in V,\ u \notin V_C\}$ where V_C is the vertexes in the core and E_C is the edges among V_C. N_v represents the set of v's neighbors where $v \in N(C)$. $|N_v \bigcap V_C|$ is the number of vertices in C connected with v. Therefore, we use $closeness(v,\ C) = \frac{|N_v \bigcap V_C|}{|V_C|}$ to quantity the closeness between the vertex v and the core C.

B. Data sources

Protein interaction data: The yeast PPI data is downloaded from DIP [19], updated on Feb. 28, 2012. The datasets contain 22,570 interactions between 5,023 proteins.

Gene expression data: We download the data from the NCBI Gene Expression Omnibus website [20]. It is available in the form of a $9,335 \times 36$ matrix, includes expression profiles of 9,335 probes under 36 different time points, updated on Apr 14, 2011.

Essential gene data: A list of essential proteins of *S.cerevisiae* are downloaded from MIPS [21], SGD [22], DEG [23] and SGDP [24], which contain 1,285 essential proteins altogether.

C. The CMBI algorithm

CMBI operates in two phases. In the first phase, it uses essential proteins as seeds to grow protein complexes. Specifically, all the essential proteins in the yeast PPI network are curated by comparing protein set V with the known essential protein set. We acquire a protein set denoted as $Ess(v)$ which consists of 1156 essential proteins. The protein complex cores are detected from the neighborhood graph G_v of each vertex $v \in Ess(v)$. Initially, the essential protein seed v is joined in the complex core C. Subsequently, v's each direct neighbor u is checked. If u is essential protein or $FS(v,\ u)$ is more than Threshold T, the neighbor u will be add to C. After all complex cores from essential protein seeds are built, CMBI begins to detect protein complexes. The neighbors of each complex core C are considered. Every protein $w \in N(C)$ with $FS(w,\ v) > T$ ($v \in C$)will be inserted to the complex core C so as to form the complex. Finally, since the number of proteins in the every known complex are more than 1, the predicted complexes only including one protein will be discarded. If a predicted complex consists of all proteins in another predicted complex, the latter will be removed.

At the same time, it can be found that there are a few known protein complexes not containing essential proteins at all. In the second phase, CMBI attempts to discover these complexes. CMBI constructs a set H by collecting the rest of proteins in the yeast PPI network. The proteins in the set H don't belong to the complexes expanded from essential proteins. They are sorted descending on their secondary-level degree. CMBI selects the protein with the maximal secondary-level degree in the set H as seed to build the complex core. Next the seed's neighbors in set H are processed. If the seed's special neighborhood graph SG only including the seed and its neighbors in set H is dense(with $den(SG) > 0.7$ [25], [26]), SG is directly predicted as a complex core; otherwise, the seed's neighbors in set H will be continually removed from SG according to their degrees from low to high until SG is dense. And then the complex core C is generated. The proteins in the core C will be removed from set H. Those cores only consist of one protein or two proteins will be discarded. The remaining proteins in H are sorted descending on their secondary-level degree again. Similarly, other cores also are constructed. The core C is grown from its neighborhood $N(C)$. If $closeness(w,\ C) > 0.5$ [5] where $w \in N(C)$, w will be added to C. After all neighbors in $N(C)$ are handled, the complex is formed. The rest of cores are expanded to complexes in the same way. Algorithm 1 shows the pseudocodes of the CMBI algorithm.

III. Results and Discussions

We have applied the CMBI method on the yeast PPI networks. In this section, we first describe the evaluation methods used in our experiments, and then study the performance of CMBI and the impact of the Threshold T on CMBI. CMBI is compared with eight other clustering algorithms: MCODE [3], MCL [2], CFinder [4], CMC [6], COACH [5], SPICi [7], HC-PIN [8] and ClusterONE [9]. The values of the parameters in each algorithm are selected from those recommended by the authors.

Evaluation methods

One evaluation method we use is to match the generated complexes with known complex set [18], and calculate sensitivity (Sn), specificity (Sp) and f-measure, respectively. We derive 408 typical complexes including two or more proteins from the CYC2008 [18] as the benchmark complex set and use the same scoring scheme used by [3] to determine how effectively a predicted complex matches a reference complex. If two complexes overlap each other, they must share one or more proteins. The Overlap Score (OS) of a predicted complex $vs.$ a benchmark complex is then a measure of biological significance of the prediction, assuming that the reference set of complexes is biologically relevant. OS is calculated by using

$$\text{OS} = \frac{i^2}{a \times b} \tag{6}$$

where, i refers to the number of proteins shared by a predicted complex and a known complex, a is the number of proteins in the predicted complex and b is the number of proteins in the known complex. If OS is 1, it means that a predicted complex has the same proteins as a known complex. On the contrary, when OS equals to 0, there is not a shared protein between the predicted complex and the known complex [3].

The number of true positives (TP) is defined as the number of predicted complexes with OS over a threshold value and the number of false positives (FP) is the total number of predicted complexes minus TP. The number of false negatives (FN) equals the number of known complexes not matched by predicted complexes. Sn and Sp are defined as $TP/(TP + FN)$ and $TP/(TP+FP)$, respectively [3]. F-measure, or the harmonic mean of Sn and Sp, can then be used to evaluate the overall performance of the clustering algorithms [27]:

$$\text{F-measure} = \frac{2 \times Sn \times Sp}{Sn + Sp} \tag{7}$$

In addition, we also use the functional enrichment of GO terms (p-values) as an evaluation measure to check the performance of each clustering algorithm. A complex is associated with a known function by determining whether the number of proteins known to be annotated with the function is enriched, as judged by the hypergeometric distribution. The p-value can be used to determine the probability that a given set of proteins is enriched by a given functional group by random chance. In [25], it is used as a criterion to assign each cluster to a known function. The smaller the p-value, the more evidence the clustering is not random. In terms of GO annotations, a group of genes with a smaller p-value is more significant than the one with a higher p-value. Consider a cluster of size c, with m proteins sharing a particular annotation A. Also assume that there are N proteins in the PPI database, and M of them are known to have annotation A. Given that, the probability of observing m or more proteins that are annotated with A out of N proteins is:

$$P = 1 - \sum_{i=0}^{m-1} \frac{\binom{M}{i}\binom{N-M}{c-i}}{\binom{N}{c}} \tag{8}$$

Based on above formulation, a p-value is calculated for each of three ontologies. In the case of multiple annotations from the same ontology, the one with the smaller p-value is assigned to the cluster as functional annotation. That being said, the p-value without any restriction is not enough to label clusters as significant. Hence we use the recommended cutoff value of 0.01 [28] in order to select significant complex within each ontology. A popular software package for evaluating the statistical significance of GO terms represented in a set of genes extracted from a population is GO::TermFinder [29], which calculates p-values (with Bonferroni correction) using above formula. GO::TermFinder accepts a list of genes of interest and returns a list of GO terms with which the genes are associated, with corresponding P-values and FDR values (if desired) associated with the enrichment of these terms in the gene list. In our experiments, the direct use of GO::TermFinder

978-1-4673-4398-5/12/$31.00 ©2012 IEEE

Algorithm 1 CMBI Algorithm

```
Input:
    PPI network $G = (V, E)$;
    essential protein set $Ess(v)$;
    gene expression profiles;
    similarity threshold $T$;
Output:
    set of protein complexes $SC$ discovered from G;
Description:
  $SC = \phi$; //initialization
  for each vertex $v \in Ess(v)$ do
    construct the core graph of $G_v$, $C = (V_C, E_C)$; //$V_C = \{u|u \in Ess(v)$ or $FS(u, v) > T, u \in G_v\}$
    for each vertex $w \in N(C)$ do  //$N(C)$ includes all direct neighbors of $C$
      if $FS(v, w) > T$ then  //$v \in C$
        insert $w$ into $C$;
    if $C \nsubseteq C_x$ then $SC = SC \cup \{C\}$;//$C_x \in SC$
  $H = V - Q$; //The set Q contains all proteins in the complexes grown from the essential protein seeds.
  sort each $v \in H$ in descending order according to its secondary-level degree;
  for each vertex $v \in H$ do
    construct the core graph of $G_v$, $C = (V_C, E_C)$; //$V_C = \{u|u \in H$ and $den(C) > 0.7, u \in G_v\}$
    $H = H - V_C$;
    if $C > 2$ then //discard those cores only including one protein or two proteins
      for each vertex $w \in N(C)$ do  //$N(C)$ includes all direct neighbors of $C$
        if $closeness(w, C) > 0.5$ then
          insert $w$ into $C$;
      $SC = SC \cup \{C\}$;
  output the complexes in $SC$;
```

is not convenient for analyzing GO enrichment of a vast amount of complexes uncovered by all kinds of algorithms, because this software package can only handle one module at a time. Therefore, combined with the latest version of this toolkit [30], we have used the Perl language to develop a procedure that can automatically process a large number of functional modules in turn.

A. Comparison with the known complexes

After researching the effect of Overlap Score threshold on number of predicted and matched known complexes, Bader *et al.* [3] find that the average and maximum number of matched known complexes drops more quickly from zero until an OS threshold of 0.2 than from 0.2 to 0.9. It indicates that many predicted complexes only have one or a few proteins that overlap with known complexes. An OS threshold value which falls within the region from 0.2 to 0.3 thus seems to filter out most predicted complexes that have insignificant overlap with known complexes. Table I shows the basic information of predicted complexes by various methods when OS is set as 0.2. $\#PC$ is the number of complexes identified by each algorithm. AS is the average size of the complexes detected by each algorithm. MS is the number of proteins in the maximal complex predicted by each algorithm. MKC represents the number of real complexes that match at least a predicted one and MPC is the number of correct predictions which match at least a real complex. PMC is the number of complexes perfectly matching the known complexes. In other words, a prediction has the same proteins with the known complex matched by it. As shown in table I, CMBI detects 793 protein complexes, of which 351 match 161 real complexes. The maximal complex predicted by CMBI contains 114 proteins and the average size of these complexes is 15.80. Besides, CMBI identifies 10 complexes overlapping fully with the known complexes. The properties of complexes discovered by other algorithms also are shown in table I. table II shows the matching comparison in terms of Sn, Sp and F-measure. On DIP data of $S.cerevisiae$, the F-measure of CMBI is 0.50, which is 316.67%, 92.31%, 100.00%, 35.14%, 11.11%, 100.00%, 38.89% and 28.21% higher than MCODE, MCL, CFinder, CMC, COACH, SPICi, HC-PIN and ClusterONE, respectively. Our CMBI algorithm can achieve the highest F-measure by providing the second-highest sensitivity and comparable specificity, which shows that CMBI can predict protein complexes very accurately.

TABLE I
BASIC INFORMATION OF PREDICTIONS.

Algorithms	#PC	AS	MS	MKC	MPC	PMC
CMBI	**793**	**15.80**	**114**	**161**	**351**	**10**
MCODE	59	13.59	82	30	28	2
MCL	928	5.15	122	195	174	12
CFinder	197	13.31	1821	83	75	12
CMC	235	6.13	32	124	119	8
COACH	902	9.18	59	219	319	15
SPICi	574	4.70	48	143	118	7
HC-PIN	277	5.67	118	149	119	20
ClusterONE	371	4.90	24	136	155	6

B. GO analysis

In many studies, the GO has been used as the 'gold standard' to validate the functional relevance of the obtained network modules. In this subsection, as described by *Evaluation methods*, we used the GO biological process (BP) annotation to take GO enrichment analysis with our developed analytical tool based on GO::TermFinder software package [30].

978-1-4673-4398-5/12/$31.00 ©2012 IEEE

TABLE III
BP ANALYSIS OF DISCOVERED COMPLEXES.

Algorithms	**CMBI**	MCODE	MCL	CFinder	CMC	COACH	SPICi	HC-PIN	ClusterONE
$\#SC$	**713**	55	414	122	196	736	297	176	253
$\#PC$	**793**	59	928	197	235	902	574	277	371
Proportion	**89.91%**	93.22%	44.41%	61.93%	83.40%	81.60%	51.74%	63.54%	68.19%
P-score	**13.95**	7.75	5.66	7.19	8.77	7.99	6.13	9.04	8.18

TABLE II
THE MATCHING RESULTS OF EACH ALGORITHM.

Algorithms	Sn	Sp	F-measure
CMBI	**0.57**	**0.44**	**0.50**
MCODE	0.07	0.47	0.12
MCL	0.45	0.19	0.26
CFinder	0.19	0.38	0.25
CMC	0.30	0.51	0.37
COACH	0.63	0.35	0.45
SPICi	0.31	0.21	0.25
HC-PIN	0.31	0.43	0.36
ClusterONE	0.36	0.42	0.39

The p-values (with Bonferroni correction) of protein complexes predicted by each algorithm are calculated. The detected complexes with corrected p-value<0.01 [28] are considered to be significant. Maraziotis *et al.* [13] conclude that the proportion of significant complexes over all identified ones can be used to evaluate the overall performance of various algorithms. In addition to this measure, we also use the average $-log(p - value)$ of predicted complexes to check the performance of the prediction algorithms. table III shows the GO enrichment results of BP. In table III, SC is the number of significant complexes with p-value≤0.01. PC is the number of protein complexes predicted by each method. P-score is the average $-log(p-value)$ of identified complexes. As shown in table III, there are 713 significant complexes in all 793 complexes discovered by CMBI. CMBI predicts higher proportion (89.91%) of significant complexes than other algorithms excepting MCODE. However, MCODE predicts 59 complexes and correctly matches only 30 known complexes as shown in table I. Moreover, table I also shows that $\#PC$ and MKC of MCODE is far fewer than those of other algorithms. More importantly, the P-score of complexes identified by CMBI reaches up to 13.95, which is 80.0%, 146.5%, 94.0%, 59.1%, 74.6%, 127.6%, 54.3% and 70.5% higher than that of complexes detected by MCODE, MCL, CFinder, CMC, COACH, SPICi, HC-PIN and ClusterONE, respectively.

We find a interesting fact that many complexes mined by CMBI don't match any known complexes but they have very low p-values. Due to the incompleteness of the reference complexes, these complexes may provide potential candidate complexes for biologists to validate.

The results from GO analysis demonstrate that the biological significance of complexes identified by CMBI is much stronger than that of complexes discovered by other algorithms.

C. *Effect of the parameter T*

In this experiment, we study the effect of the threshold T on the performance of CMBI. Figure 1 shows the F-measure of CMBI under different values of T. As shown in figure 1, the F-measure of CMBI increases with the increase of T when $T < 1.4$. The size of each complex decreases as T increases in the interval since the number of proteins captured by the essential protein seed drops as T increase. But the F-measure of CMBI is 0.47 and remains unchanged when $T \geq 1.8$. It means that the number of proteins belonging to a complex remains the same. In this case, CMBI doesn't actually consider gene expression information. But even then, it can be see that the F-measure of CMBI is still higher than that of other algorithms. It demonstrates that it is successful for us to incorporate essential proteins information into CMBI. It can also be found that the consideration of gene expression profiles further improves the performance of CMBI when $1.0 \leq T \leq 1.8$. We recommend that the suitable setting of T would be in the range i.e., $T \in [1.2,\ 1.4]$. In fact, the performance of CMBI doesn't change significantly in this interval. The parameter T is set 1.4 in our experiment.

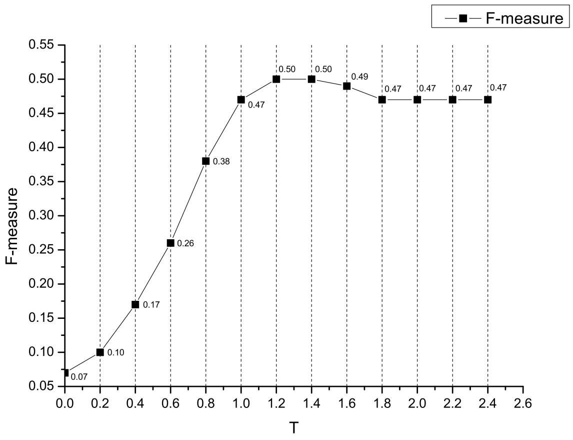

Fig. 1. **The effect of threshold T.** Figure 1 shows how the variation of parameter T affects the F-measure of CMBI.

IV. CONCLUSION

In this research, we develop a new technique called CMBI to identify protein complexes in yeast PPI network by integrating other biological resources into the PPI data. CMBI constitutes protein complexes that originate from a kernel protein set (protein complex core) built up from a seed protein. More

978-1-4673-4398-5/12/$31.00 ©2012 IEEE

specifically, CMBI selects the essential proteins as seeds to generate protein complexes by using gene expression and essential protein information. In addition to the essential proteins, CMBI also chooses other proteins as seeds to construct protein complexes based on the inherent organization of protein complexes. In order to characterize these clusters as protein complexes we check their biological relevance. This is achieved through some criteria such as matching analysis between predicted complexes and known complexes and the functional enrichment analysis of the derived complexes in GO terms. The evaluation and analysis of our predictions demonstrate show that CMBI performs very well and outperforms other other algorithms.

Since the integration of multiple data sources shows great superiority in predicting protein complexes, we will apply the idea to other research areas such as the identification of functional modules including one or more complexes in future.

Acknowledgment

This work is supported in part by the National Natural Science Foundation of China under Grant Nos.61073036, 61003124, the Ph.D. Programs Foundation of Ministry of Education of China No.20090162120073, the Freedom Explore Program of Central South University No.201012200124, High-tech Program of China Hunan Provincial Science and Technology Department Nos.2010GK3049, 2011GK3138, Aid program for Science and Technology Innovative Research Team in Higher Educational Institutions of Hunan Province No.2010212, Hunan Provincial Department of education Science Foundation under Grant No.11C0281, the U.S. National Science Foundation under Grants CCF-0514750, CCF-0646102 and CNS-0831634.

References

[1] A. Tong, B. Drees, G. Nardelli, G. Bader, B. Brannetti, L. Castagnoli, M. Evangelista, S. Ferracuti, B. Nelson, S. Paoluzi, M. Quondam, A. Zucconi, C.W. Hogue, S. Fields, C. Boone, and G. Cesareni, "A combined experimental and computational strategy to define protein interaction networks for peptide recognition modules," *Science*, vol. 295, no. 5553, pp. 321-324, 2002.

[2] S. Van Dongen, "Graph Clustering by Flow Simulation," *In PhD Thesis* University of Utrecht; 2000.

[3] G. Bader and C. Hogue, "An automated method for finding molecular complexes in large protein interaction networks," *BMC Bioinformatics* 2003, **4**:2.

[4] B. Adamcsek, G. Palla, I.J. Farkas, I. Derenyi, and T. Vicsek, "CFinder: locating cliques and overlapping modules in biological networks," *Bioinformatics*, vol. 22, no. 8, pp. 1021-1023, 2006.

[5] M. Wu, X. Li, C.K. Kwoh, and S. Ng, "A Core-Attachment based Method to Detect Protein Complexes in PPI Networks," *BMC Bioinformatics*, vol. 10, no. 169, 2009.

[6] G. Liu, L. Wong, and H.N. Chua, "Complex discovery from weighted PPI networks," *Bioinformatics*, vol. 25, no. 15, pp. 1891-1897, 2009.

[7] P. Jiang and M. Singh, "SPICi: a fast clustering algorithm for large biological networks," *Bioinformatics*, vol. 26, no. 8, pp. 1105-1111, 2010.

[8] J. Wang, M. Li, J. Chen, Y. Pan, "A Fast Hierarchical Clustering Algorithm for Functional Modules Discovery in Protein Interaction Networks," *IEEE Transactions on Computational Biology and Bioinformatics*, vol. 8, no. 3, pp. 607-620, 2011.

[9] T. Nepusz, H. Yuand, and A. Paccanaro, "Detecting overlapping protein complexes in protein-protein interaction networks," *Nature Methods*, vol. 9, no. 5, 471-475, 2012.

[10] N Bhardwaj. and H. Lu, "Correlation between gene expression profiles and protein-protein interactions within and across genomes," *Bioinformatics*, vol. 21, no. 11,pp. 2730-2738, 2005.

[11] C.J. Wolfe, I.S. Kohane, and A.J. Butte, "Systematic survey revals general applicability of 'guilt-by-association' within gene coexpression networks," *BMC Bioinformatics*, vol. 6, no. 79, 2005.

[12] J. Feng, R. Jiang, and T. Jiang, "A Max-Flow Based Approach to the Identification of Protein Complexes Using Protein Interaction and Microarray Data," *IEEE Transactions on Computational Biology and Bioinformatics*,vol. 8, no. 3, pp. 621-634, 2011.

[13] I. Maraziotis, K. Dimitrakopoulou, and A. Bezerianos, "Growing functional modules from a seed protein via integration of protein interaction and gene expression data," *BMC Bioinformatics*, vol. 8, no. 408, 2007.

[14] H. Shatkay, S. Edwards, W.J. Wilbur, M. Boguski, "Genes, themes, and microarray: using information retrieval for large-scale gene analysis," *In Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology, August 16-23, La Jolla, California* Edited by: Altman R, Bailey TL, Bourne P, Gribskov M, Lengauer T, Shindyalov IN, Eyck LFT, Weissig H. AAAI Press, 2000, pp. 317-328.

[15] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto and D. Parisi, "Defining and Identifying Communities in Networks," *Proc. Nat'l Academy of Sciences USA*, vol. 101, no. 9, pp. 2658-2663, 2004.

[16] K.I. Goh, M.E. Cusick, D. Valle, B. Childs, M. Vidal, and A.L. Barabasi, "The human disease network," *Proc. Nat'l Academy of Sciences USA*, vol. 104, no. 21, pp. 8685-8690, 2007.

[17] Z. Dezso, Z.D. Oltvai, A.L. Barabasi, "Bioinformatics Analysis of Experimentally Determined Protein Complexes in the Yeast Saccharomyces cerevisiae," *Genome Res*, vol. 13, pp. 2450-2454, 2003.

[18] S. Pu, J. Wong, B. Turner, E. Cho, and S.J. Wodak, "Up-to-date catalogues of yeast protein complexes," *Nucleic Acids Res*, vol. 37, no. 3, pp. 825-831, 2009.

[19] I. Xenarios, D.W. Rice, L. Salwinski, M.K. Baron, E.M. Marcotte, and D. Eisenberg, "DIP: the database of interacting proteins," *Nucleic Acids Research*, vol. 28, no. 1, pp. 289-291, 2000.

[20] B.P. Tu, A. Kudlicki, M. Rowicka, and S.L. McKnight, "Logic of the yeast metabolic cycle: temporal compartmentalization of cellular processes," *Science*, vol. 310, pp. 1152-1158, 2005.

[21] H.W. Mewes, et al., "MIPS: analysis and annotation of proteins from whole genomes in 2005," *Nucleic Acids Res.*, vol. 34, no. Database issue, pp. 169-172, 2006.

[22] J.M. Cherry, et al., "SGD: Saccharomyces Genome Database," *Nucleic Acids Res.*, vol. 26, no. 1, pp. 73-79, 1998.

[23] R. Zhang, Y. Lin, "DEG 5.0, a database of essential genes in both prokaryotes and eukaryotes," *Nucleic Acids Res.*, vol. 37, no. Database issue, pp. 455-458, 2009.

[24] http://www-sequence.stanford.edu/group/yeast_deletion_project

[25] M. Altaf-Ul-Amin, Y. Shinbo, K. Mihara, K. Kurokawa, S. Kanaya, "Development and implementation of an algorithm for detection of protein complexes in large interaction networks," *BMC Bioinformatics*, vol. 7, no. 207, 2006.

[26] X. Li, C. Foo, and S. Ng, "Discovering protein complexes in dense reliable neighborhoods of protein interaction networks," *CSB*, vol.6, pp. 157-168, 2007.

[27] X. Li, M. Wu, C.K. Kwoh, and S. Ng, "Computational approaches for detecting protein complexes from protein interaction networks: a survey," *BMC Genomics*, vol. 11, no. suppl+1, pp. S3, 2010.

[28] H. Hu, X. Yan, Y. Huang, J. Han, and X. Zhou , "Mining coherent dense subgraphs across massive biological networks for functional discovery," *Bioinformatics*, vol. 21 no. suppl 1, pp. 213-221, 2005.

[29] E.I. Boyle, S. Weng, J. Gollub, H. Jin, D. Botstein, J.M. Cherry, and G. Sherlock, "GO::TermFinder-open source software for accessing Gene Ontology information and finding significantly enriched Gene Ontology terms associated with a list of genes," *Bioinformatics*, vol. 20, no. 18, pp. 3710-3715, 2004.

[30] N. Daraselia, A. Yuryev, S.Egorov, and I. Mazo Iand Ispolatov, "Automatic extraction of gene ontology annotation and its correlation with clusters in protein networks," *BMC Bioinformatics*, vol. 8, 2007.