Algorithms for Detecting Protein Complexes in PPI Networks: An Evaluation Study

Min Wu[1], Xiaoli Li[2], Chee-Keong Kwoh[1]
{wumi0002,asckkwoh}@ntu.edu.sg and xlli@i2r.a-star.edu.sg

[1] School of Computer Engineering, Nanyang Technological University, Singapore
[2] Institute for Infocomm Research, 21 Heng Mui Keng Terrace, Singapore.

Abstract. Since protein complexes play important biological roles in cells, many computational methods have been proposed to detect protein complexes from protein-protein interaction (PPI) data. In this paper, we first review four reputed protein-complex detection algorithms (MCODE[2], MCL[21], CPA[1] and DECAFF[14]) and then present a comprehensive evaluation among them on two popular yeast PPI data[3]. We also discuss their relative strengthes and disadvantages to guide interested researchers.

1 Introduction

Multiple-protein complexes are key molecular entities to perform important cellular functions. For example, the complex "RNA Polymerase II" can produce mRNAs, snoRNAs, and some of the snRNAs. The increasing amount of PPI data has enabled us to detect protein complexes from the PPI networks. Recently, many algorithms (*e.g.*, [2, 12, 15, 1, 14, 4]) have been proposed to detect protein complexes in PPI networks, based on the observation that dense regions in PPI networks often correspond to protein complexes [20]. All these algorithms adopt different strategies to detect protein complexes and thus obtain different results. Therefore, a systematic study is greatly desired for both examining which algorithm is with superiority in a specific situation and investigating the challenges in subsequent researches.

Recently, Brohee and Helden [3] conducted a comparison among four algorithms, Molecular Complex Detection (MCODE) [2], Super Paramagnetic Clustering (SPC) [8], Restricted Neighborhood Search Clustering (RNSC) [12] and Markov Clustering (MCL) [21]. For evaluations, they proposed several new measures, one of which called "separation" apparently favors non-overlapping clusterings. Based on their evaluation measures, their analysis showed that MCL was most superior. For this reason, some subsequent studies adopt MCL to process their own PPI data for protein complex detection [13, 9]. However, proteins can be involved in multiple complexes in real biological systems and protein complexes can thus have overlaps with others. Various evaluation measures [2, 14, 9], which take the overlaps into considerations, have already been utilized to do fair

[3] Supplementary information: http://www.ntu.edu.sg/home2006/wumi0002/PRIB08.html

comparison. In addition, the incomplete MIPS complex data as reference set will make their analysis less convincing. As the complex data is now more enriched and some new methods (*e.g.*, [1], [4] and [14]) emerged, a new evaluation study for protein complex prediction algorithms is necessary.

In this paper, we also compare four algorithms, namely MCODE, MCL, and two new algorithms CPA [1] and DECAFF [14]. A more comprehensive complex data [9] is collected as reference set and different criteria for performance measurement are used to evaluate the quality of predicted complexes and the accuracy of the predictions.

2 Review of algorithms

In this section, we first introduce some basic terminologies for graphs and then review above four algorithms for protein complex detection.

2.1 Terminology

PPI data can be easily modeled as a simple 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. In G, a *walk* is a sequence of vertices where edges exist between two adjacent vertices. The set of all the neighbors of a vertex $v \in V$ is denoted as $N_v = \{u | u \in V, (u, v) \in E\}$ and the degree of v is the cardinality of N_v, written as $deg(v)$. The density of G, denoted as $den(G)$, is defined in equation 1. The neighborhood graph of v is defined as $G_v = (V', E')$, where $V' = \{v\} \cup N_v$, and $E' = \{(u_i, u_j) | (u_i, u_j) \in E, u_i, u_j \in V'\}$. The neighborhood graph G_v is namely the subgraph which consists of all v's immediate neighbors (including v) and all the edges among them. A k-core is a subgraph in which all the vertices have degrees no less than k and the order of a k-core is k if it is not a $(k + 1)$-core. Given two graphs $A = (V_A, E_A)$ and $B = (V_B, E_B)$, the neighborhood affinity between A and B, $NA(A, B)$, is also defined in equation 1 to measure the similarity between them [2].

$$den(G) = \frac{2 \times |E|}{|V| \times (|V| - 1)} \quad \text{and} \quad NA(A, B) = \frac{|V_A \cap V_B|^2}{|V_A| \times |V_B|}. \tag{1}$$

2.2 Algorithms

The four algorithms adopt different strategies to detect dense regions in PPI networks as protein complexes. We will briefly introduce them and then give a short summary.

MCODE [2] detects densely connected regions in PPI networks as protein complexes. It first weights every vertex based on their local neighborhood densities, and then selects seed vertices with high weights as initial clusters and augments these clusters by outward traversing from the seeds.

In the first step, for a vertex v, MCODE first finds the k-core with the highest order in the subgraph formed by N_v, denoted as g_k. The final weight of v is the product of the order and density of g_k.

In the second step, MCODE selects the seed vertex with the highest weight as an initial cluster and conducts a *Depth-First-Search* (*DFS*) from the seed. The cluster grows by including vertices as follows. Given a vertex v and a cluster $C = (V_C, E_C)$, $v \in V_C$, u will be added into C if $u \in N_v$ and $w(u) > (1 - T_w) \times w(v)$, where T_w is the weight threshold for cluster formation. If a cluster can't grow any more, it will be removed from the current PPI network. MCODE will repeat above processes until no more clusters can be detected.

MCODE also has an optional post-processing step. MCODE will filter the clusters that doesn't even contain a 2-core. In some clusters, some vertices with very low weights will also be removed. A "fluff" operation is also introduced for generating overlapping clusters in this post-processing step.

MCL [21], which simulates random walks in graphs (*e.g.*, PPI networks), has recently been used for protein complex detection in [13] and [9].

MCL takes the adjacency matrix of the PPI network as input. It has two operators, called expansion and inflation. In each iteration, expansion operates the matrix by multiplication (*e.g.*, matrix squaring) and inflation by taking the power for each matrix entry. Expansion operator is for assigning new probabilities for all pairs of nodes, where one node is the starting point of walk and another is the destination. Inflation operator changes the probabilities for all the walks, boosting the probabilities of intra-cluster walks and demoting inter-cluster walks. Finally, iterating expansion and inflation will separate the PPI network into many segments, which are predicted as protein complexes. In this way, MCL can only generate non-overlapping clusters.

CPA [1] is proposed to detect protein complexes by keeping track of the periphery information of an augmenting cluster. To form a protein complex, CPA first selects a seed as the initial cluster and then augments it by including vertices one-by-one, which are closely related with the current cluster.

The seed is first selected as the one with the highest weight or the one with the highest degree if all the vertices have weights zero.The initial cluster is a seed and then the cluster grows gradually by including vertices one-by-one from the neighbors of the current cluster. In each iteration, only the priority vertex, which has both a high weight and a high cluster property score with respect to this cluster, will be added into the cluster. Given a cluster c_k and a vertex v_n outside c_k, the cluster property score of v_n with respect to c_k, cp_{nk}, represents the cohesiveness between c_k and v_n.

$$cp_{nk} = \frac{|E_{nk}|}{d_k \times |N_k|}, \tag{2}$$

where d_k is the density of c_k, $|N_k|$ is the number of nodes in c_k and $|E_{nk}|$ is the number of nodes in c_k connecting to v_n.

Once a cluster is generated, it is removed from the PPI network. Thus, CPA can only detects non-overlapping clusters if there is no further process. To generate overlapping clusters, CPA is extended to allow a cluster to include vertices that are closely associated with it but appear in other clusters.

DECAFF [14] is proposed to detect dense and reliable subgraphs as protein complexes in PPI networks. DECAFF first mines dense subgraphs in the neighborhood graphs, then merges subgraphs with high *Neighborhood Affinity* scores and finally filters out some subgraphs with low reliability.

Firstly, based on the hierarchical network model of PPI networks [18], the Hub-Removal Algorithm is proposed to mine multiple possible dense subgraphs in a neighborhood graph.

Secondly, dense subgraphs detected by the Hub-Removal Algorithm combined with local cliques detected by the LCMA algorithm [15] are then processed by the merging algorithm [15]. In particular, two dense subgraphs with high *Neighborhood Affinity* scores will be merged into a larger subgraph.

Thirdly, each interaction will be assigned reliability score based on its experimental sources and the functional information of its two interacting proteins. The reliability of a cluster is then defined as the average score of all interactions within this cluster. Clusters with low reliability scores, which are possibly false predictions, are finally filtered by DECAFF.

Summary. Table 1 summarizes the main features of four algorithms. Although all these algorithms share some common features (*e.g.*, using weighting schemes), they also have their particular features, *e.g.*, the clusters predicted by MCL can cover all the proteins in the PPI network and DECAFF uses additional function information. The coming evaluation part will show how these features effect the performance of the algorithms.

Table 1. Main features of protein complex prediction algorithms.

Algorithms	Main features of algorithms					
	1	2	3	4	5	6
MCODE [2]	Local neighborhood search	Yes	No	No	Yes	No
MCL [13, 9]	Global flow simulation	Yes	No	No	No	Yes
CPA [1]	Local neighborhood search	Yes	Yes	No	Yes	No
DECAFF [14]	Local neighborhood search	Yes	Yes	Yes	Yes	No

Features: 1-main idea of the algorithm; 2-weighting edges or vertices; 3-using density threshold; 4-using functional information; 5-detecting overlapping subgraphs; 6-covering all the nodes in the PPI network.

3 Evaluation Criteria

In this section, we will use different criteria proposed by previous studies to systematically evaluate these four algorithms.

3.1 Co-annotation and co-localization within predicted complexes

Each pair of proteins can have a score, which is the semantic similarity between GO (Gene Ontology) terms annotating them [19, 17, 22]. Thus, a predicted complex can be also assigned a score, which is the average of the GO scores of all protein pairs within it. Since protein complexes are formed to perform a specific cellular function, proteins within the same complex tend to share common functions and be co-localized. The average score of all predicted complexes can therefore be used to evaluate their overall quality [9]. In addition, we will use the method in [22] to calculate the semantic similarity between GO terms and the semantic similarity between two proteins will be defined as the maximum similarity between GO terms annotating them respectively. The "biological process" and "cellular component" taxonomies are used to calculate the co-annotation score and co-localization score respectively.

3.2 Precision, Recall and F-measure

The neighborhood affinity score between a predicted complex p and a real complex b in the benchmark, $NA(p, b)$, is used to determine whether they match with each other. If $NA(p, b) \geq \omega$, they are considered to be matching (ω[4] is set as 0.20 in [2, 14] and 0.25 in [4], respectively.). We assume that P and B are the sets of complexes predicted by a computational method and real ones in the benchmark, respectively. N_{cp} is the number of predicted complexes which match at least a real complex and N_{cb} is the number of real complexes that match at least a predicted one. Precision and recall are defined as follows [4]:

$$\begin{aligned} N_{cp} &= |\{p | p \in P, \exists b \in B, NA(p, b) \geq \omega\}|, \\ N_{cb} &= |\{b | b \in B, \exists p \in P, NA(p, b) \geq \omega\}|, \\ Precision &= \frac{N_{cp}}{|P|} \quad \text{and} \quad Recall = \frac{N_{cb}}{|B|}. \end{aligned} \tag{3}$$

F-measure, as the harmonic mean of precision and recall, can be used to evaluate the overall performance of the different techniques,

$$F = 2 \times Precision \times Recall / (Precision + Recall). \tag{4}$$

3.3 Sensitivity, Positive predictive value and Separation values

Sensitivity (Sn) and positive predictive value (PPV) are also used to evaluate the accuracy of the predictions [3, 9]. Given n benchmark complexes and m predicted complexes, T_{ij} is the number of proteins in common between i^{th} benchmark complex and j^{th} predicted complex. Sn and PPV are defined as:

$$Sn = \frac{\sum_{i=1}^{n} \max_{j} \{T_{ij}\}}{\sum_{i=1}^{n} N_i} \quad \text{and} \quad PPV = \frac{\sum_{j=1}^{m} \max_{i} \{T_{ij}\}}{\sum_{j=1}^{m} T_{\cdot j}}. \tag{5}$$

[4] In our experiments, ω is set as 0.20.

Here N_i is the number of proteins in the i^{th} benchmark complex, and $T_{.j} = \sum_{i=1}^{n} T_{ij}$. Generally, high Sn values shows that the prediction has a good coverage of the proteins in the real complexes and high PPV values indicate predicted complexes are likely to be true positive. The accuracy of a prediction, Acc, is finally defined as the geometric average of sensitivity and positive predictive value,

$$Acc = \sqrt{Sn \times PPV}. \tag{6}$$

A new measure called *separation* [3] is proposed to emphasize the one-to-one correspondence between a predicted complex and a real complex. The separation value for the i^{th} benchmark complex and j^{th} predicted complex, sep_{ij}, is defined as:

$$sep_{ij} = \frac{T_{ij}}{\sum_{i=1}^{n} T_{ij}} \times \frac{T_{ij}}{\sum_{j=1}^{m} T_{ij}}. \tag{7}$$

The complex-wise separation sep_b and the cluster-wise separation sep_p are defined in equation 8. The final geometrical separation value (Sep) is defined as the geometrical mean of sep_b and sep_p.

$$sep_b = \frac{\displaystyle\sum_{i=1}^{n}\sum_{j=1}^{m} sep_{ij}}{n} \quad \text{and} \quad sep_p = \frac{\displaystyle\sum_{j=1}^{m}\sum_{i=1}^{n} sep_{ij}}{m}. \tag{8}$$

4 Experimental results and Discussions

In this section, we systematically compare the performance of four algorithms based on above evaluation criteria. In our experiments, DIP data [23] and MIPS PPI data [10] are used for protein complex prediction. MIPS data consists of 12317 interactions among 4543 proteins and DIP data (released on 12/02/2007) consists of 17203 interactions among 4930 proteins. The protein complexes collected by [9] are used as reference data, which contains (1) manually curated complexes from MIPS [10] and Aloy *et al.* [6] and (2) those extracted from the SGD database [7] based on GO annotations. This reference data consists of 428 real complexes.

For MCL algorithm, the *inflation* parameter is set as 2.1 which maximizes both the accuracy in equation (6) and F-measure when using MIPS data and it is set as 1.9 when using DIP data. For MCODE, the optimal parameters are set as follows: *Depth = 3, VWP = 0, haircut = T, Fluff = T* and *fluff density threshold = 0.2.* In this way, each algorithm obtained predictions as shown in Table 2.

4.1 Co-annotation and co-localization within predicted complexes

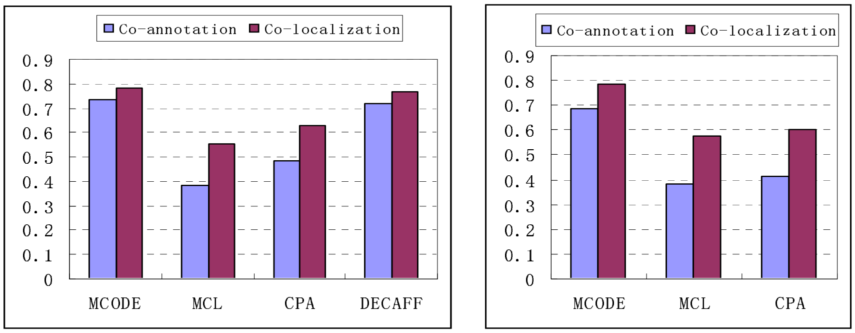
To assess the quality of predicted complexes, we calculated the co-annotation and co-localization scores for the complexes predicted by four algorithms.

Table 2. The results of various algorithms using MIPS and DIP data.

Data	MIPS PPI data				DIP PPI data		
Algorithms	MCODE	MCL	CPA	DECAFF	MCODE	MCL	CPA
# complexes predicted	133	1160	921	1221	183	1117	1140
# proteins covered	694	4543	2198	864	1173	4930	2939
# interactions covered	1298	4226	2102	3264	2789	5519	4441

††Since we have not implemented the DECAFF algorithm, the results of DECAFF using DIP data are not collected in this study.

In figure (1), MCL is observed to be with the lowest functional and localization similarity. The complexes predicted by MCODE have the highest quality. DECAFF also achieves much higher functional and localization similarity than MCL and CPA when using MIPS PPI data. In fact, MCL obtains some clusters that have 3 proteins and only 2 interactions among them and all this kind of clusters are considered to be unpromising and discarded by three other algorithms. In addition, MCL and CPA also detect many clusters that are protein pairs. However, some interactions are even unreliable in MIPS and DIP data. These two facts help to interpret why MCL and CPA predict protein complexes with lower quality in terms of co-annotation and co-localization.



(1) using MIPS PPI data

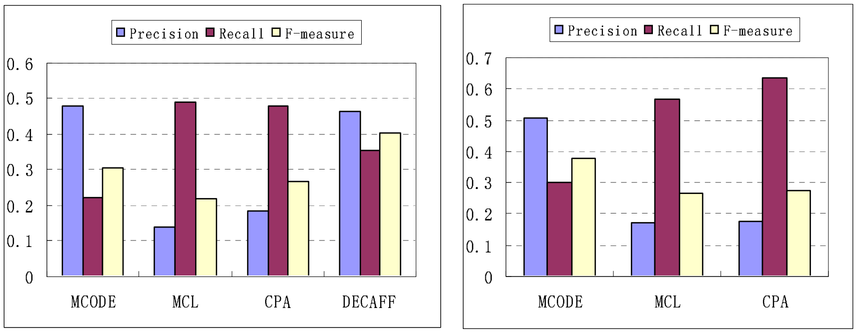
(2) using DIP PPI data

Fig. 1. Co-annotation and co-localization scores of predicted complexes from different algorithms, using two different PPI data.

4.2 Precision, Recall and F-measure

Using MIPS PPI data, MCODE predicted 133 complexes, out of which 64 match 95 real complexes in the benchmark; MCL predicted 1160 complexes, of which 163 match 210 real complexes; CPA detected 921 complexes, of which 170 match 206 real ones and DECAFF detected 1221 complexes, of which 566 match only 152 real ones. Figure 2 shows the overall comparison using precision, recall and F-measure. In terms of finding the most real complexes (*i.e.*, achieving the highest *recall*), MCL and CPA perform best, finding 210 and 206 real ones respectively.

Meanwhile, it is observed that DECAFF and MCODE have a higher *precision* than CPA and MCL. The fact shows that a larger proportion of protein complexes predicted by DECAFF and MCODE correspond to real complexes, consistent with our observation that the protein complexes detected by DECAFF and MCODE are with higher quality in figure 1. Similar results are obtained when using DIP data. In addition, MCODE just predicts a small number of protein complexes and achieves the lowest *recall*, which indicates a severe limitation of MCODE that it will result in many false negatives.



(1) using MIPS PPI data

(2) using DIP PPI data

Fig. 2. The performance comparison for various algorithms (I).

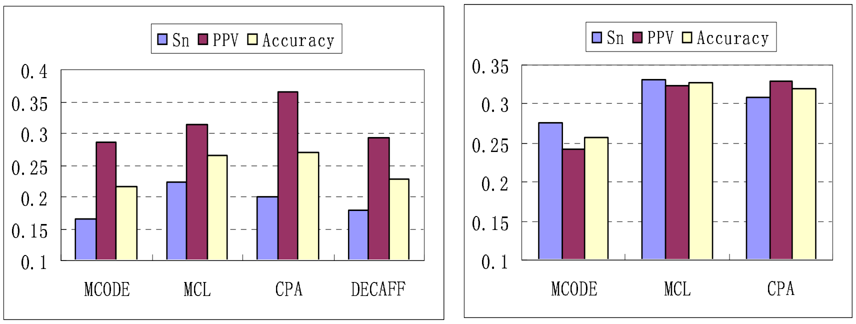
4.3 Sensitivity, Positive predictive value

Figure 3 shows the comparison among four algorithms using the evaluation measures like *sensitivity*, *PPV* and *accuracy*. In terms of these evaluation measures, CPA and MCL perform better than DECAFF and MCODE as shown in Figure 3. We can roughly explain these results based on our previous observations. First, CPA and MCL can cover more proteins as shown in table 2, which will be helpful for increasing their *sensitivity* values. Second, they can indeed detect more real complexes with higher recall as shown in figure 2, such that it is reasonable for them to achieve higher *Sn* and *PPV* values.

4.4 Separation values

High *separation* values indicate that predicted complexes and real ones tend to have one-to-one correspondences. In general, if a clustering algorithm has N_{cp} close to N_{cb} (see equation 3), it often has a high *separation* value. Table (3) shows the *separation* values of different algorithms using MIPS PPI data. It is quite obvious that CPA achieves the highest *separation* value. On the contrary, DECAFF gains the lowest *separation* value since there exist cases that many predicted complexes correspond to the same real one.

In reality, there are some pairs of protein complexes that have large overlaps. For example, "TORC1 complex" and "TORC2 complex" have 2 common proteins in figure 4; RNA Polymerase I, II and III also have large shared subunits



(1) using MIPS PPI data

(2) using DIP PPI data

Fig. 3. The performance comparison for various algorithms (II).

Table 3. The separation values of different algorithms using MIPS PPI data.

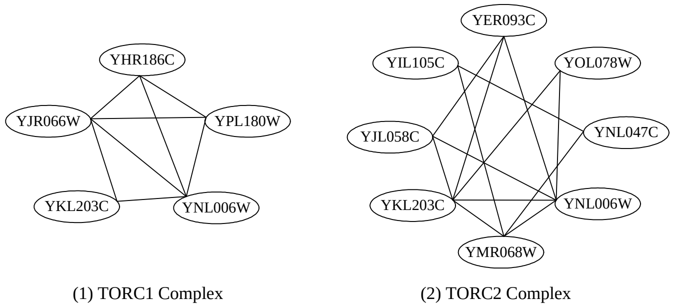
Algorithms	MCODE	MCL	CPA	DECAFF
N_{cp}	64	163	170	566
N_{cb}	95	210	206	153
Complex-wise separation sep_b	0.130	0.323	0.371	0.176
Cluster-wise separation sep_p	0.418	0.119	0.172	0.062
Geometrical separation	0.233	0.196	0.253	0.104

with each other. Although DECAFF predicts protein complexes with the lowest separation value, it therefore has the advantage that it can well separate two protein complexes with large shared subunits. When using MIPS PPI data as shown in figure 5, a complex(ID: 1019) predicted by DECAFF can perfectly match “TORC1 complex”; and another one (ID: 843) with four proteins (YKL203C, YMR068W, YNL006W and YOL078W) is a sub-component of “TORC2 complex”. MCODE and CPA can also partially match these two real complexes as shown in figure 5. On the contrary, MCL has only a predicted complex (ID: 83) which contains all the five proteins in “TORC1 complex” and share 4 common proteins with “TORC2 complex”. Although some techniques can generate overlapping clusters based on non-overlapping ones [2, 1, 9], they still can’t help MCL to separate these two complexes.

Due to the existence of many overlapping real complexes, the *separation* value is not a good measure to evaluate the quality of a clustering. However, some clustering algorithms may generate redundant results (*e.g.*, 3 complexes [ID: 719, 805, 1019] predicted by DECAFF match well with “TORC1 complex”.), cluster-wise separation values would be useful for evaluating the degree of redundancy for these algorithms.

4.5 Discussions

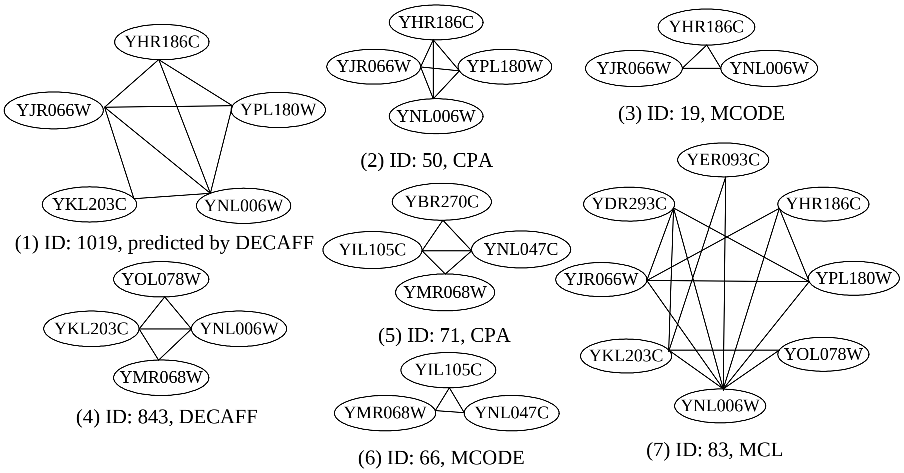
In this section, we conducted a comprehensive evaluation study for the four algorithms. The experimental results show the strengthes and disadvantages of each algorithm.



(1) TORC1 Complex

(2) TORC2 Complex

Fig. 4. Two tor complexes [16], with proteins YKL203C and YNL006W in common.



(1) ID: 1019, predicted by DECAFF

(2) ID: 50, CPA

(3) ID: 19, MCODE

(4) ID: 843, DECAFF

(5) ID: 71, CPA

(6) ID: 66, MCODE

(7) ID: 83, MCL

Fig. 5. Predicted complexes by different algorithms using MIPS data. Predicted complexes (1), (2), (3) and (7) match "TORC1 complex", and (4), (5), (6) and (7) match "TORC2 complex".

MCODE detects protein complexes that are with the highest quality, in terms of the function and localization similarity of proteins within predicted complexes. It also achieves the highest *precision*, indicating the predicted complexes are indeed with high quality. However, it can only predicts a small number of protein complexes, which cover the smallest fraction of proteins and interactions. This results in its low *sensitivity* and many false negative predictions.

MCL predicts protein complexes with high *PPV* and *recall* values. It covers most proteins and interactions and thus achieves the highest *sensitivity* score. MCL has predicted many small clusters (*i.e.*, protein triads with only 2 interactions among them) that are generally agreed as potential false positives. It hence reduces the overall quality of predicted complexes. However, high-quality PPI data may help to increase its *precision*. In addition, MCL can't separate complexes that have large shared components.

CPA also achieves high *PPV* and *recall* values. In comparison, although CPA has a lower *sensitivity* score, it achieves a higher F-measure and accuracy than MCL when using MIPS PPI data and performs comparably with MCL when using DIP PPI data.

DECAFF, like MCODE, predicts complexes with high quality and gains a high *precision* and a low *sensitivity* value. In addition, DECAFF predicts many more complexes than MCODE, achieving better *recall* and *PPV* values. However, although DECAFF can separate complexes that have large shared subunits, it generates complexes with a high redundancy.

5 Conclusions and future directions

While multiple algorithms for detection of protein complexes in PPI networks have been proposed in recent years, different algorithms have their own strengthes and disadvantages. In this study, we first reviewed four reputed algorithms and then performed a comprehensive comparison among them on two different PPI data by using various evaluation criteria.

Compared with MCL and CPA, MCODE and DECAFF predict protein complexes with higher quality and achieve higher *precision* values and lower *sensitivity* values. For further comparison, DECAFF performs better than MCODE in terms of F-measure and accuracy. Unlike MCODE and DECAFF, MCL and CPA obtain higher *recall* and *PPV* values, which indicates that they can detect more genuine complexes. However, the overall quality of their predicted complexes is not as good. In our empirical study, CPA perform slightly better than MCL. However, If each interaction is weighted by its reliability score, MCL is likely to obtain a better performance [3].

This comparative study suggests that new algorithms for protein complex detection should gain high *recall* and *PPV* values like MCL and CPA, as well as increasing the quality of their predicted complexes and achieving high *precision* values like MCODE and DECAFF. In addition, based on the analysis of experimentally detected protein complexes, a protein complex generally contains a core in which proteins are highly co-expressed and share high functional similarity [5]. The complex core is often surrounded by some attachments which are in short-lived cooperation with the core. This kind of organization structure of protein complexes is also strongly supported in [11]. All current methods have not considered the organization of complexes when detecting them. New algorithms should also take this characteristic into consideration.

References

1. M. Amin, Y. Shinbo, K. Mihara, K. Kurokawa, and S. Kanaya. Development and implementation of an algorithm for detection of protein complexes in large interaction networks. *BMC Bioinformatics*, 7:207, 2006.
2. G. Bader and C. Hogue. An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics*, 4:2, 2003.

3. S. Brohee and J. van Helden. Evaluation of clustering algorithms for protein-protein interaction networks. *BMC Bioinformatics*, 7:488, 2006.
4. H. N. Chua, K. Ning, W. K. Sung, H. W. Leong, and L. Wong. Using indirect protein-protein interactions for protein complex prediction. In *CSB*, pages 97–109, 2007.
5. Z. Dezso, Z. D. Oltvai, and A. L. Barabasi. Bioinformatics analysis of experimentally determined protein complexes in the yeast saccharomyces cerevisiae. *Genome Res.*, 13:2450–2454, 2003.
6. P. Aloy *et al.* Structure-based assembly of protein complexes in yeast. *Science*, 303(5666):2026–2029, 2004.
7. S. S. Dwight *et al.* Saccharomyces genome database provides secondary gene annotation using the gene ontology. *Nucleic Acids Research*, 30(1):69–72, 2002.
8. S. Wiseman *et al.* Superparamagnetic clustering of data. *Physical Review*, 57(4):3767–3783, 1998.
9. C. C. Friedel, J. Krumsiek, and R. Zimmer. Boostrapping the interactome: Unsupervised identification of protein complexes in yeast. In *RECOMB*, pages 3–16, 2008.
10. ftp://ftpmips.gsf.de/yeast/PPI/PPI_18052006.tab.
11. A.C. Gavin, P. Aloy, P. Grandi, R. Krause, and M. Boesche *et al.* Proteome survey reveals modularity of the yeast cell machinery. *Nature*, 440(7084):631–636, 2006.
12. A.D. King, N. Przulj, and I. Jurisica. Protein complex prediction via cost-based clustering. *Bioinformatics*, 20(17):3013–3020, 2004.
13. N.J. Krogan, G. Cagney, H. Yu, G. Zhong, and X. Guo *et al.* Global landscape of protein complexes in the yeast saccharomyces cerevisiae. *Nature*, 440(7084):637–643, 2006.
14. X. Li, C. S. Foo, and S. K. Ng. Discovering protein complexes in dense reliable neighborhoods of protein interaction networks. In *CSB*, pages 157–168, 2007.
15. X. L. Li, S. H. Tan, C. S. Foo, and S. K. Ng. Interaction graph mining for protein complexes using local clique merging. *GIW*, 16(2):260–269, 2005.
16. R. Loewith, E. Jacinto, S. Wullschleger, A. Lorberg, and J. Crespo *et al.* Two tor complexes, only one of which is rapamycin sensitive, have distinct roles in cell growth control. *Molecular Cell*, 10:457–468, 2002.
17. P.W. Lord, R.D. Stevens, A. Brass, and C.A.Goble. Semantic similarity measures as tools for exploring the gene ontology. In *Pacific Symposium on Biocomputing*, pages 601–612, 2003.
18. E. Ravasz, A. Somera, D. Mongru, Z. Oltvai, and A. Barabasi. Hierarchical organization of modularity in metabolic networks. *Science*, 297(5586):1551–1555, 2002.
19. P. Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *IJCAI*, pages 448–453, 1995.
20. A.H. Tong, B. Drees, G. Nardelli, G. Bader, and B. Brannetti *et al.* A combined experimental and computational strategy to define protein interaction networks for peptide recognition modules. *Science*, 295(5553):321–324, 2002.
21. S. Van Dongen. *Graph Clustering by Flow Simulation.* University of Utrecht, Netherlands, 2000.
22. J.Z. Wang, Z. Du, R. Payattakool, P. Yu, and C. Chen. A new method to measure the semantic similarity of go terms. *Bioinformatics*, 23(10):1274–1281, 2007.
23. I. Xenarios, L. Salwnski, X. Duan, P. Higney, S.M. Kim, and D. Eisenberg. Dip, the database of interacting proteins: a research tool for studying cellular networks of protein interactions. *Nucleic Acids Research*, 30(1):303–305, 2002.