DISCOVERING PROTEIN COMPLEXES IN DENSE RELIABLE NEIGHBORHOODS OF PROTEIN INTERACTION NETWORKS

Xiao-Li Li*
Knowledge Discovery Department, Institute for Infocomm Research, Heng Mui Keng Terrace, 119613, Singapore
**Email: xlli@i2r.a-star.edu.sg*

Chuan-Sheng Foo
Computer Science Department, Stanford University, Stanford CA 94305-9025 USA
Email: csfoo@stanford.edu

See-Kiong Ng
Knowledge Discovery Department, Institute for Infocomm Research, Heng Mui Keng Terrace, 119613, Singapore
Email: skng@i2r.a-star.edu.sg

Multiprotein complexes play central roles in many cellular pathways. Although many high-throughput experimental techniques have already enabled systematic screening of pairwise protein-protein interactions *en masse*, the amount of experimentally determined protein complex data has remained relatively lacking. As such, researchers have begun to exploit the vast amount of pairwise interaction data to help discover new protein complexes. However, mining for protein complexes in interaction networks is not an easy task because there are many data artefacts in the underlying protein-protein interaction data due to the limitations in the current high-throughput screening methods. We propose a novel DECAFF (Dense-neighborhood Extraction using Connectivity and conFidence Features) algorithm to mine for dense and reliable subgraphs in protein interaction networks. Our method is devised to address two major limitations in current high throughout protein interaction data, namely, incompleteness and high data noise. Experimental results with yeast protein interaction data show that the interaction subgraphs discovered by DECAFF matched significantly better with actual protein complexes than other existing approaches. Our results demonstrate that pairwise protein interaction networks can be effectively mined to discover new protein complexes, provided that the data artefacts in the underlying interaction data are taken into account adequately.

1. INTRODUCTION

Multiprotein complexes play central roles in many cellular pathways. Common examples include the ribosomes for protein biosynthesis, the proteasomes for breaking down proteins, and the nuclear pore complexes for regulating proteins passing through the nuclear membrane. Searching for protein complexes is therefore an important research focus in molecular and cell biology. However, while tens of thousands of pairwise protein-protein interactions have been detected by high throughput experimental techniques (e.g. yeast-two-hybrid), only a small subset of the many possible protein complexes has been experimentally determined [1].

Given that the protein complexes are molecular aggregations of proteins assembled from multiple stable protein-protein interactions, researchers have recently begun to explore the possibility of exploiting the current abundant datasets of pairwise protein-protein interactions to help discover new protein complexes (see Section 2). In fact, it has been observed that densely connected regions in the protein interaction graphs often correspond to actual protein complexes [2], suggesting the identities of protein complexes can be revealed as tight-knitted subcommunities in protein-protein interaction maps. This has led to previous works that looked into the mining of cliques [3] or other dense graphical subcomponents [4–7] in the interaction graphs for putative complexes.

However, the protein interaction networks derived from current high throughput screening meth-

*Corresponding author.

ods are not an easy source for mining as there are still many data artefacts in the underlying interaction data due to inherent experimental limitations. In fact, it has been repeatedly shown that the current protein interaction data is still incomplete and noisy [8, 9] and it is important to take this into account when devising algorithms to mine the protein interaction networks. For example, the use of cliques for detecting complexes would be too constraining and cannot provide satisfactory coverage.

In this work, we propose a novel DECAFF (Dense-neighborhood Extraction using Connectivity and conFidence Features) algorithm that is devised to address two major limitations in current high throughout protein interaction data, namely, incompleteness and high data noise. Unlike conventional methods, our DECAFF method specifically mines for maximal dense local neighborhoods (instead of cliques) and filters the unreliable protein complexes by estimating the reliability of each protein interaction in the network. Experimental results with yeast protein interaction data show that the interaction subgraphs discovered by DECAFF matched significantly better with actual protein complexes than other existing approaches. Our results confirm that there are indeed dense graphical subcomponents in the pairwise protein interaction networks that correspond to actual multiprotein complexes, and we could exploit the interactome to help map the protein complexome more effectively by taking in account of the data artefacts in the underlying protein interaction data.

2. RELATED WORKS

By modeling protein interaction data as a large undirected graph where the vertices represent unique proteins and edges denote interactions between two proteins, Ref. 2 was one of the first to reveal that protein complexes generally corresponded to dense regions (highly interconnected subgraphs) in the protein interaction graphs. Ref. 3 then exploited this finding and used cliques (fully connected subgraphs) as a basis to detect protein complexes and functional modules in protein interaction networks. However, the use of cliques was too constraining given that the incompleteness in the currently available interaction data; as a result, the method could only detect fewer protein complexes.

Bader then proposed a novel MCODE algorithm that discovered protein complexes based on the proteins' connectivity values in a protein interaction graph [4]. The algorithm first computes the vertex weighting from its neighbor density and then traverses outward from a seed protein with a high weighting value to recursively include neighboring vertices whose weights are above a given threshold. As the highly weighted vertices may not be highly connected to one another, this approach does not guarantee that the discovered regions are dense. As a result, not all the detected regions correspond to protein complexes. In fact, in the post preprocessing step of the MCODE algorithm, there was a need to filter for the so-called "2-core"s as an attempt to eliminate some obvious non-dense region detected by the algorithm.

Clustering algorithms have also been proposed to identify dense regions in a given graph by partitioning it into disjoint clusters [10–12]. However, these general graph clustering algorithms cluster each vertex (protein) into one specific group which made them inappropriate for this biological application as a protein is often involved in multiple complexes (i.e. clusters) [8, 13]. Another clustering approach was proposed by Ref. 5, which used a restricted neighborhoods search clustering algorithm (RNSC) to predict protein complexes by partitioning the protein-protein interaction network using a cost function. However, like many clustering algorithms, their results depended on the quality of the initial random seeds. In addition, there were relatively fewer complexes predicted by this algorithm, reflecting another limitation of clustering approaches.

In our recent work [6], we proposed the LCMA algorithm (Local Clique Merging Algorithm) to mine the dense subgraphs for protein complexes. Instead of adopting the over-constraining cliques as the basis for protein complexes, LCMA adopted a local clique merging method as an attempt to address the current incompleteness limitation of protein interaction data. Evaluation results showed that LCMA was better in detecting complexes than full clique [3], MCODE [4] and RNSC algorithm [5]. However, LCMA also shared the same drawback as MCODE in that the graphical components detected by the algorithm are not guaranteed to be dense subgraphs.

Most recently, Ref. 7 proposed an algorithm based on the assumption that two nodes that belong to the same cluster have more common neigh-

bors than two nodes that are not in the same cluster. Besides ensuring the high density (≥ 0.7) of a graph, their algorithm also keeps track of its *cluster property*, a numerical measure for measuring whether a dense graph contains more than one dense component. If a graph has a low value for the *cluster property*, then it will be separated into multiple subgraphs. However, given the higher proportion of noisy protein interactions (up to 50%) in current protein interaction networks [9], the formations of clusters will be greatly affected when the algorithm computes the *cluster property*.

In this paper, we propose the DECAFF algorithm which first mines local dense neighborhoods (in addition to local cliques) for each vertex (protein) and then merges these local neighborhoods according to their affinity to form maximal dense regions that correspond to possible protein complexes. In addition, given the potentially high false positive rate in the protein interaction data, DECAFF also filters away possible false protein complexes that have low reliability scores, ensuring that the proteins in the predicted protein complexes are connected by high confidence protein interactions in the underlying network. The overall DECAFF algorithm is described in Section 3.3.

3. THE PROPOSED TECHNIQUES

Mathematically, a protein-protein interaction (PPI) network can be represented as a graph $G_{PPI} = (V_{PPI}, E_{PPI})$, where V_{PPI} represents the set of the interacting proteins and E_{PPI} denotes all the detected pairwise interactions between proteins from V_{PPI}. Our objective is to detect a set of subgraphs $C = \{g = (V, E) \mid |V| \geq 3, V \subseteq V_{PPI}, E \subseteq E_{PPI}\}$, where each g is a *dense subgraph* (possibly overlapping) in G_{PPI} that may correspond to an actual multi-protein complex. Additionally, since many false positive protein interactions in G_{PPI} may be assembled into false protein complexes, we also require that each detected dense graph g has a high *reliability* score.

3.1. Mining for dense subgraphs

Let us first introduce the notion of the local neighborhood graph for each vertex:

Definition 3.1. The local neighborhood graph of a vertex $v_i \in V$ in $G = (V, E)$ is defined as $G_{v_i} = (V_{v_i}, E_{v_i})$, where

$$V_{v_i} = \{v_i\} \cup \{v \mid v \in V, \{v, v_i\} \in E\}, \text{ and}$$

$$E_{v_i} = \{\{v_j, v_k\} \mid \{v_j, v_k\} \in E, v_j, v_k \in V_{v_i}\} \quad (1)$$

In other words, vertex v_i's local neighborhood graph is the subgraph formed by v_i and all its immediate neighbors with the corresponding interactions in G. In this work, we have devised our algorithm to focus first on each vertex's local neighborhood graph in a bottom-up fashion, as it is impractical to directly detect dense subgraphs in a top-down fashion from G_{PPI}, which is usually a very large graph with thousands of vertices and tens of thousands of edges.

Let us now define the notion of the density of a graph :

Definition 3.2. The density of a graph $g = (V, E)$ is defined as its clustering coefficient (cc) [12]:

$$cc(g) = \frac{|E|}{|V| * (|V| - 1|)/2} = \frac{2 * |E|}{|V| * (|V| - 1|)} \quad (2)$$

Note that $0 \leq cc(g) \leq 1$ since the maximum number of edges in an undirected graph $g = (V, E)$ is $|V| * (|V| - 1)/2$. If g is a clique, then $cc(g) = 1$ as it has the maximum number of edges. In this work, we detect putative protein complexes from dense subgraphs of G_{PPI} instead of the conventional requirement for cliques. We define a dense graph as one in which its density is at least $\max(\delta, 0.5)$, where δ is a user-defined threshold to provide for more stringent conditions. The results reported in this paper are based on setting δ as 0.7, which is also the same setting used in the recent work by Ref. 7.

The following theorem indicates that we can adopt a bottom-up approach to discover dense subgraphs from protein interaction network:

Theorem 1. *Every dense neighborhood g in G_{PPI} can be assembled using only the dense neighborhoods of its inner vertices.*

The formal proof for Theorem 1 can be found in Appendix A of the *Supplementary Materials* (which is available at `http://www1.i2r.a-star.edu.sg/~xlli/csb_supp.pdf`). Theorem 1 suggests a strategy of first finding the local dense neighborhoods for each vertex, and then obtaining larger dense neighborhoods by merging these dense sub-regions. As such, DECAFF algorithm mines for dense subgraphs in two steps:

(1) First, we compute the local dense neighborhoods for all the vertices in the given interaction graph G_{PPI}. We use a local clique mining method to locate the local cliques, and then deploy a novel hub-removal technique to heuristically detect local dense subgraphs in each vertex's local neighborhood graph. Such systematic scanning of the local dense neighborhoods in the entire interaction graph will allow DECAFF to discover most of the local dense regions, resulting in significantly higher recall than other algorithms (see Section 4).
(2) Then, we merge the extracted local dense neighborhoods to obtain maximal dense neighborhoods that correspond to larger complexes.

3.1.1. *Mining for local dense subgraphs*

Given that we already have an efficient method for discovering local cliques [6], we first mine for each vertex's local cliques, and then expand the collection of other local dense subgraphs using a hub-removal procedure which we will describe shortly. In this way, we can ensure that both cliques and non-clique dense subgraphs are detected effectively.

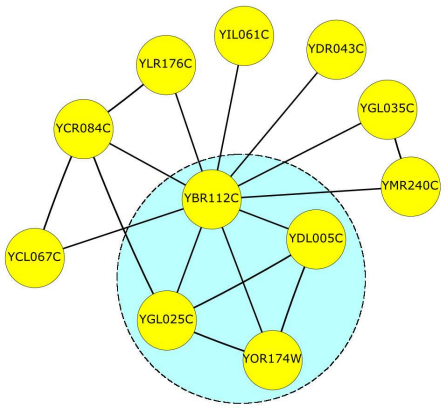

Fig. 1. A local clique obtained from YBR112C's local neighborhood graph

To detect local cliques, we adapt the method from the LCMA algorithm [6] which is basically an elimination process in which the neighborhood vertices of a given vertex are iteratively removed, starting from the least connected vertex (vertex with lowest degree), to increase the overall density of the local neighborhood graph. The details of this step can be found in Appendix B of *Supplementary Materials*. Here, we show an example (Figure 1) of mining a local clique from a local neighborhood graph for the vertex (protein) YBR112C to illustrate how it works. In this case, the neighbors YIL061C, YDR043C, YGL035C, YMR240C, YCL067C, YLR176C, YCR084C were sequentially removed. This results in the final local dense neighborhood shown in the circled area of Figure 1 which is a clique $d = (V, E)$, $V = \{$YBR112C, YDL005C, YOR174W, YGL025C$\}$ and density $cc\,(d) = 1$ ($|V| = 4$, and $|E| = 6$).

Although the LCMA algorithm can obtain the local cliques, an actual protein complex may not be presented as a fully connected subgraph in a protein interaction network for various reasons as previously discussed (e.g. incompleteness of current protein interaction data). There are thus possibly many other dense but non-clique subgraphs for each vertex that could form parts of a target complex. In DECAFF, we devise a Hub Removal algorithm to efficiently detect multiple dense subgraphs with densities larger than the given threshold δ.

In the hierarchical network model proposed by Ref. 14, a biological network is constructed from a small cluster of highly connected nodes by generating replicas of the network at each step and linking the external nodes of the replicated clusters to the central node of the old cluster. This construction procedure suggests a heuristic for recovering the smaller dense clusters in the network by reversing the process, which forms the basis for the Hub Removal algorithm. Basically, we start by removing the most highly connected node (the hub) and its corresponding edges from the network, and then recursively repeating this procedure on its connected components, until a dense cluster is recovered and the removed hub is re-inserted back into the cluster. A more detailed description of this algorithm can be found in Appendix B of *Supplementary Materials*.

Figure 2 shows the results of applying the Hub Removal algorithm to further discover dense subgraphs in the local neighborhoods of the protein YBR112C. While the previous LCMA algorithm could only discover a single fully connected graph {YBR112C, YDL005C, YOR174W, YGL025C} in this neighborhood graph, our recursive Hub Removal Algorithm is able to detect an additional 4 dense subgraphs: {YBR112C, YGL035C,

YMR240C}, {YBR112C, YCR084C, YLR176C}, {YBR112C, YCR084C, YCL067C} and {YBR112C, YDL005C, YOR174W, YGL025C, YCR084C}. Note that as this approach allows the discovery of multiple, possibly overlapping, dense neighborhoods for each vertex, it also allows the possibility of a vertex (protein) participating in multiple complexes.

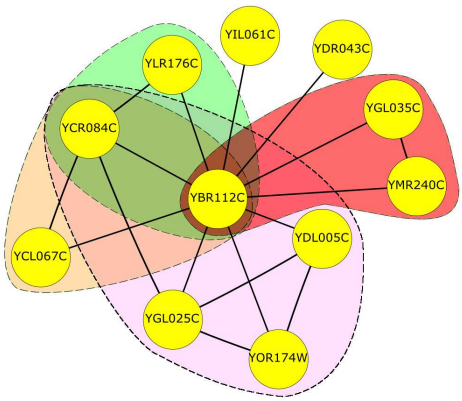

Fig. 2. Multiple dense subgraphs obtained from YBR112C's local neighborhood graph

3.1.2. *Merging for maximal dense neighborhoods*

In an interaction graph with potentially incomplete interaction data, it is likely that a large protein complex is presented in the PPI graph as a composite of multiple overlapping dense neighborhoods. In addition, there is also biological evidence that many complexes are formed by multiple substructures such as subcomplexes [8, 15]. We therefore adopt an additional step to merge the individual local dense neighborhoods (that have been detected in section 3.1.1) using a heuristic that assigns overlapping neighborhoods with comparable sizes a high affinity to be merged.

Definition 3.3. Neighborhood Affinity. Given two neighborhoods (subgraphs) A and B, we define the Neighborhood Affinity NA between them as

$$NA(A,B) = \frac{|A \cap B|^2}{|A| * |B|} \tag{3}$$

Equation 3 quantifies the degree of similarity between neighborhoods. Note that if one neighborhood's size, e.g. $|B|$, is much bigger than $|A|$, then $NA(A,B)$ will be small since $|A \cap B|/|A| < 1$ and $|A \cap B| \ll |B|$. Our heuristic is based on the hypothesis that if two neighborhoods have larger intersection sets and similar sizes, then they are more similar and have a larger affinity.

The merging step takes the set of local dense neighborhoods LDN (comprising local cliques output by the LCMA algorithm and the dense neighborhoods obtained from the Hub Removal Algorithm) and tries to merge neighborhoods that have affinity values greater than a threshold ω. The merging process is performed iteratively until the average density of the subgraphs in LDN starts to fall. The details of the algorithm are provided in Appendix B of *Supplementary Materials*, which also contains a further illustrative example in Appendix C.

3.2. Filtering for reliable subgraphs

In the previous section, we have taken into consideration the presence of possible incompleteness (missing interactions) in the protein interaction datasets by mining for only dense subgraphs and using a merging process to build up larger complexes. However, as it is also well known that many high throughput protein interaction datasets contain a high rate of false positives (noisy interactions), our algorithm could also be susceptible to the presence of false positive interactions especially since we have employed a relatively relaxed graphical constraint to infer protein complexes. To minimize the false detection of complexes assembled with false positive interactions, we perform an additional filtering process on the detected subgraphs (i.e. complexes) by modeling the protein interaction network as a weighted graph where each protein interaction or edge is assigned a weight that corresponds to its reliability, and then filtering those detected dense subgraphs that consist of protein interactions with low reliability.

3.2.1. *Computing reliability of protein interactions*

We begin by assigning a prior reliability to each protein interaction using the approach proposed by Ref. 16. The method first computes a reliability score for each experimental source, since protein interactions discovered through different experimental sources may have different quality. This score is computed using additional biological information on the proteins, and it is defined to be the fraction of inter-

action pairs from each source that shared at least one function. Then, using the reliability score for each experimental source, the method estimates the prior reliability $r_{u,v}$ for each individual protein-protein interaction (u, v) as follows [17, 18]:

Definition 3.4. The prior reliability of a protein-protein interaction pair $r_{u,v}$ is defined as

$$r_{u,v} = 1 - \prod_{i \in ES_{u,v}} (1 - r_i)^{n_{i,u,v}} \quad (4)$$

where r_i is the reliability score of experimental source i, $ES_{u,v}$ is the set of experimental sources from which the interaction (u, v) was observed, and $n_{i,u,v}$ is the number of times that (u, v) was detected in experimental source i. The rule of thumb is that protein interactions discovered through multiple experiments tend to be more reliable.

Note that the reliability score $r_{u,v}$ in Definition 4 computes the confidence of a particular data source. To determine whether a specific interaction detected between a pair of proteins (u, v) is a reliable one, we also need to check whether the proteins u and v shared a function (in this work we use the MIPS functional catalog http://mips.gsf.de/desc/yeast/). Therefore, we compute a posterior reliability $R_{u,v}$ for each protein interaction based on the following three cases. We use R to denote the event that the given interaction is a true interaction (i.e. it is reliable), S to denote the event that the proteins in the given interaction share a common function, D to denote the event that the proteins in the given interaction do not share a common function, and U to denote the event that either protein (or both proteins) have unknown functions.

Case 1: The two proteins share a common function.

In this case, $P(R|S)$, the probability that the interaction is true given that the proteins share a common function can be written as:

$$P(R|S) = \frac{P(S|R) * P(R)}{P(S)} \quad (5)$$

Note that $P(R)$ is the prior reliability, i.e., $P(R) = r_{u,v}$. $P(S)$, the probability that the two proteins have a common function, can be formulated as

$$P(S) = P(S|R) * P(R) + P(S|\neg R) * P(\neg R) \quad (6)$$

Together, the above equations give the posterior reliability $P(R|S)$ as long as we can estimate $P(S|R)$ and $P(S|\neg R)$. In this paper, we estimate $P(S|R)$ using a small-scale experimental data set ss from the DIP protein interaction set (http://dip.doe-mbi.ucla.edu/):

$$P(S|R) = \frac{|\{(p_1, p_2)|share(p_1, p_2), (p_1, p_2) \in ss\}|}{|\{(p_1, p_2)|(p_1, p_2) \in ss\}|} \quad (7)$$

where $share(p_1, p_2)$ denotes that proteins p_1 and p_2 share at least one function.

To estimate $P(S|\neg R)$, we randomly selected 1 million protein pairs that were not present in current protein interaction datasets to form a non-reliable protein interaction set ns. Then, $P(S|\neg R)$ is estimated as follows:

$$P(S|\neg R) = \frac{|\{(p_1, p_2)|share(p_1, p_2), (p_1, p_2) \in ns\}|}{|\{(p_1, p_2)|(p_1, p_2) \in ns\}|} \quad (8)$$

Case 2: The two proteins do not share a common function.

In this case, $P(R|D)$ can be computed as:

$$P(R|D) = \frac{P(D|R) * P(R)}{P(D)} \quad (9)$$

where $P(D)$ and $P(D|R)$ are computed using Equations 10 and 11 respectively:

$$P(D) = 1 - P(S) \quad (10)$$

$$P(D|R) = 1 - P(S|R) \quad (11)$$

Note that both $P(S)$ in Equation 10 and $P(S|R)$ in Equation 11 have already been computed previously in Equations 6 and 7 respectively.

Case 3: Either protein's function is unknown.

In this case, we compute the posterior reliability $P(R|U)$ given that either u or v (or both) has unknown function:

$$P(R|U) = P(S) * P(R|S) + P(D) * P(R|D) \quad (12)$$

Again, all the terms on the right hand side of Equation 12 have already been computed in the previous cases.

Given a protein interaction (u, v), its posterior reliability $R_{u,v}$ can be obtained through the computation of $P(R|S)$, $P(R|D)$ or $P(R|U)$, depending on

the available information of the functions of proteins u and v. Note for those proteins with unknown function, it is also possible to predict their functions by utilizing the topological information of PPI networks and gene expression data [16, 19].

3.2.2. *Computing reliability of detected complexes*

In this work, we detect a putative multiprotein complex as a subgraph $g = (V, E)$. We define its reliability as the average reliability score of all the protein interactions in E:

Definition 3.5. The reliability of a graph $g = (V, E)$ is defined as:

$$reliability(g) = \frac{1}{|E|} \sum_{u,v \in V,(u,v) \in E} R_{u,v} \qquad (13)$$

Suppose the mean and standard deviation of reliability distribution are μ and σ respectively. A subgraph g of G_{PPI} is regarded as a *reliable* if $(reliability(g) - \mu \geq \max(0.5, \gamma) * \sigma)$. γ is a user-defined threshold to provide for more stringent reliability requirement if necessary—the bigger the value of γ, the more reliable the predicted complexes are since their constituent protein interactions are more reliable.

3.3. The overall DECAFF algorithm

The overall DECAFF algorithm is shown in algorithm 1 as follows:

Overall DECAFF algorithm

(1) Run LCMA algorithm to detect the local cliques (stored in set LC) for each protein;
(2) Run Hub Removal algorithm to detect the local dense subgraphs (stored in set DS);
(3) $LDN = DS \bigcup LC$;
(4) Run merging algorithm to merge for maximal dense neighborhoods from LDN, which are stored in set C;
(5) FOR each graph $c \in C$
(6) IF $(reliability(c) - \mu < max(0.5, \gamma) * \sigma)$
(7) $C = C - \{c\}$;
(8) ENDIF
(9) ENDFOR

In algorithm 1, we first compute the local dense neighborhoods for all the vertices in the given interaction graph G_{PPI}. Particularly, step 1 employs a local clique mining method to locate the local cliques, and step 2 then deploys a novel hub-removal technique to detect local dense subgraphs in each vertex's local neighborhood graph. Such systematic scanning of the local dense neighborhoods in the entire interaction graph will allow DECAFF to discover most of the local dense regions (store in LDN in step 3), resulting in significantly higher recall than other algorithms. Then, step 4 merges the extracted local dense neighborhoods in the first two steps to obtain maximal dense neighborhoods that correspond to larger complexes. Finally, from steps 5 to 9, we filter away possible false protein complexes from set C that have low reliability scores, ensuring that the proteins in the predicted protein complexes are connected by high confidence protein interactions in the underlying network. The protein complexes in set C are output as the final predicted complexes.

4. EXPERIMENTS

For evaluation, we applied our DECAFF algorithm on three experimental protein-protein interaction data sets for yeast to facilitate comparisons with various current techniques.

The first dataset was collected by Ref. 4. It was used by both the MCODE algorithm [4] and the LCMA algorithm [6] to mine protein complexes. The dataset was assembled from all machine-readable resources in 2003: Uetz [20], Ito [21], Drees [22], Fromont-Racine [23], Ho [24], Gavin [8], Tong [2], Mewes(MIPS) [25], Costanzo(YPD) [26]. In total, it consists of 15,143 experimentally determined protein-protein interactions among 4,825 yeast proteins.

The second protein interaction dataset was collected from the MIPS database, which consists of 15,456 interactions (of which 12,526 are unique protein interactions) among 4,554 proteins. The data was publicly available from `ftp://ftpmips.gsf.de/yeast/PPI/PPI_18052006.tab` . It was used by Ref. 7 to mine for protein complexes.

The third dataset was collected from the BIOGRID, which consists of 82,633 interactions (of which 51,105 are unique) among 5,299 proteins. The dataset was downloaded from `http://www.thebiogrid.org` [27]. BIOGRID is the most comprehensive data set compared to the two protein interaction datasets above.

4.1. Reference complexes and evaluation metric

We evaluated the experimental results against a reference dataset of known yeast protein complexes retrieved from the MIPS (ftp://ftpmips.gsf.de/yeast/). The protein complexes in this dataset had been curated from the biomedical literature. While it is probably one of the most comprehensive public datasets of yeast complexes available, it is by no means a complete dataset — there are still many yeast complexes that remained to be discovered (hence the motivation for this work). After filtering the predicted protein complexes from the dataset, we obtained a final set of 215 yeast complexes as our benchmark for evaluation. The biggest protein complex, cytoplasmic ribosomes, contains 81 proteins while the average number of proteins in a complex is 6.38.

For assessment, we used the same evaluation metric that was adopted by previous authors for evaluating the MCODE algorithm [4], LCMA algorithm [6], and Md Altaf algorithm [7], whereby neighborhood affinity NA (Definition 3) was used to determine matching between a predicted complex $p \in P$ and a complex $m \in$ MIPS. We consider the two complexes to be matching if $NA(p, m) \geq 0.2$, which was the same threshold used in MCODE, LCMA and Md Altaf algorithm. The set of true positives (TP) is therefore defined as $TP = \{p | NA(p, m) \geq 0.2,\ p \in P,\ m \in \text{MIPS}\}$, while the set of false negatives (FN) is defined as $FN = \{m \mid \forall p(NA(p, m) < 0.2),\ p \in P,\ m \in \text{MIPS}\}$. The set of false positives (FP) is $FP = P - TP$, while the recall and precision are:

$$R = |TP|/(|TP| + |FN|) \quad (14)$$

$$P = |TP|/(|TP| + |FP|) \quad (15)$$

We use the F-measure, which is the harmonic mean of precision and recall, to evaluate the overall performance of the different techniques:

$$F - measure = 2 * P * R/(P + R) \quad (16)$$

Note that it is possible that multiple predicted complexes may correspond to a single reference complex, using the evaluation metric defined above (see definition of TP). Recent work by Gavin *et al.* [28] has shown that protein complexes have a modular structure, consisting of core proteins that are present in multiple complexes, and attachment proteins that are present in only some of them. This modularity of complexes may help to explain why multiple predicted complexes match a single benchmark complex, since the same core proteins may be present in the complexes, albeit with different attachment proteins.

It is also important to note that as our reference complex set MIPS is by no means complete, some predicted complexes which probably are true complexes will be falsely regarded as false positives (FP). As such, the F-measure of the algorithms should be taken for comparative purpose instead of at their absolute values.

4.2. Comparative results

We compared the performance of DECAFF algorithm with current computational techniques, namely, MCODE [4], LCMA [6] and Md Altaf algorithm [7]. Note that the results of MCODE algorithm were only available on their own Bader protein interaction data while the results of Md Altaf algorithm were only available on the MIPS protein interaction data. For fair comparison, we also ran the LCMA and DECAFF algorithms on all the three protein interaction data. Note that all the existing algorithms use the same MIPS complexes as a reference set.

Table 1. Overall performance of MCODE, LCMA, Md Altaf algorithm and DECAFF algorithm.

Method	Dataset	Recall	Precision	F-measure
MCODE	Bader	0.258	0.271	0.264
LCMA	Bader	0.787	0.275	0.408
DECAFF	Bader	0.883	0.392	0.543
Md Altaf	MIPS	0.601	0.111	0.188
LCMA	MIPS	0.725	0.301	0.425
DECAFF	MIPS	0.806	0.416	0.549
LCMA	BIOGRID	0.921	0.214	0.347
DECAFF	BIOGRID	0.955	0.435	0.597

[a]The comparison experiments are performed on the Bader and Hogue, MIPS, and BIOGRID protein interaction data. For all the three protein interaction data, $\omega = 0.30$ and $\gamma = 0.95$ are used in DECAFF algorithm. $\omega = 0.30$ is also used in LCMA algorithm.

Table 1 shows the overall comparison results of the different computational algorithms. Using the same Bader protein interaction data, DECAFF was able to predict 1,736 complexes, of which 681 matched 125 benchmark complexes. Overall, the F-measure of DECAFF on this dataset is 54.3%, which

is 27.9% and 13.5% higher than MCODE and LCMA respectively. Using MIPS protein interaction data, DECAFF predicted 1,220 complexes, of which 508 matched 93 benchmark complexes. On this dataset, DECAFF obtained 54.9% as its F-measure, which is 36.1% and 12.4% higher than Md Altaf algorithm and LCMA algorithm respectively.

On applying our DECAFF algorithm on the most comprehensive protein interaction data BIOGRID, we managed to predict 2,840 complexes, of which 1,235 complexes matched with 157 MIPS complexes. On this comprehensive dataset, DECAFF obtained 59.7% as its F-measure, which is 25.0% higher than the LCMA algorithm. In short, our DECAFF algorithm performed with precision and recall values that are significantly higher than all the other computational techniques in all the three evaluation datasets.

4.3. Effect of the hub removal routine

First, recall that our algorithm detects dense subgraphs in addition to the local cliques for merging, and we devised a novel hub removal routine to heuristically detect multiple dense subgraphs. To investigate the effect of using local dense neighborhoods instead of local cliques as a basis for complex mining, we re-ran our experiments with a version of DECAFF without the hub-removal routine. Interestingly, the precisions of the DECAFF without the hub-removal routine were similar or only slightly worse, whereas the recall decreased significantly at 18.9%, 25.7%, and 22.1% in Bader, MIPS, and BIOGRID interaction data respectively. This shows that in addition to the local cliques, the less graphically-stringent dense local neighborhoods in DECAFF are essential for the effective mining of many more true protein complexes than clique-based methods.

4.4. Effect of parameters ω and γ

Next, note that DECAFF algorithm employs two user-defined parameters ω and γ to control the merging process and to filter unreliable protein complexes respectively. We first investigated how the merging threshold ω affected the performance of the algorithm by running it with values of ω ranging from -1.0 to +1.0 in steps of 0.1, while keeping the filtering threshold fixed at $\gamma = 0.95$.

In all three protein interaction datasets, the effect of varying ω was similar. As ω initially increased, the resulting F-measure increased. However, increasing ω beyond 0.6 resulted in a decreased F-measure. A possible explanation for this is that more merging of the local dense neighborhoods takes place when the $\omega < 0.6$. When ω is increased beyond 0.6, the threshold becomes so strict that merging seldom takes place. However, when ω is set too low (i.e. $\omega < 0.15$), any two local dense neighborhoods will be merged as long as they have at least one common protein. Such indiscriminate merging will result in an increased number of false positives, which explains the lower F-measure values for DECAFF algorithm with low ω values. We found that the optimal values of ω for DECAFF with $\gamma = 0.95$ can be found within a large range of $0.15 < \omega < 0.55$. As such, selecting a suitable value for ω for good performance is not a problem.

To study the effect of the other user-defined constraint γ, which is used to filter unreliable protein complexes detected by DECAFF, we ran DECAFF with γ from -1.0 to +1.0 with $\omega = 0.3$. Generally, increasing γ increased the performance of DECAFF in all the three protein interaction networks, suggesting that the complexes predicted with reliable protein interactions are more likely to be true complexes. When DECAFF is used with an extremely small γ such as -1.0, the filtering step is practically nonexistent. DECAFF performed worst without filtering as the noisy protein interaction data will significantly affect the accuracy of DECAFF. This indicates that the filtering step in DECAFF is also an essential one to ensure good performance.

When compared with $\gamma = -1.0$, DECAFF's precisions with a high $\gamma = 0.95$ (used in this paper) were increased by 9.0%, 6.6%, 19.2% while making a marginal sacrifice on the recall by 3.7%, 3.2% and 3.5% on the Bader, MIPS, and BIOGRID protein interaction datasets respectively. This means that our filtering strategy of reliability is very successful since it can keep most of the true protein complexes (or protein interactions) while filtering away most of the false protein complexes.

More detailed analyses of the effect of these parameters on the performance of DECAFF can be found in Appendix E of Supplementary Materials.

4.5. Analysis of the predicted complexes

We also evaluated the statistical significance of the protein complexes predicted by DECAFF using p-values. Given a predicted complex with n proteins, the p-value computes the probability of observing k or more proteins from the complex by chance in a biological function shared by C proteins from a total genome size of G proteins:

Definition 4.1. The p-value of a predicted complex is defined as:

$$P = 1 - \sum_{i=0}^{k-1} \frac{\binom{C}{i}\binom{G-C}{n-i}}{\binom{G}{n}} \quad (17)$$

In other words, the above p-value measures whether a predicted complex is enriched with proteins from a particular function more than what would be expected by chance. Given that proteins in a protein complex are assembled to perform common biological functions, they are expected to share common functions. As such, true protein complexes should have low p-values, indicating that their collective occurrence within the graphical subcomponents detected by DECAFF did not happen merely by chance.

We evaluated the p-values for all the predicted complexes by incorporating a Bonferroni correction, and we found that majority of our predicted complexes are statistically significant at the 0.01 significance level (Typically, a cut-off α level 0.01 for Bonferroni corrected p-values is chosen such that p-values below the α level are deemed significant). Specifically, 1,729 out of the 1,737 predicted complexes (or 99.5%) detected in the Bader data, 1,205 out of the 1,221 predicted complexes (or 98.7%) detected in the MIPS data, and 2,828 out of the 2,841 predicted complexes detected in the BIOGRID data were deemed significant in terms of the above p-value.

Table 2 shows ten predicted complexes which have very small p-values (thus highly likely to be true protein complexes).

In one of these predicted complexes (ID=4), we found that 9 out of 10 proteins in this predicted complex matched exactly with a 9-protein complex in our MIPS protein complex benchmark. On further analysis, we found that the additional unmatched protein "YKL138C-A" in our predicted complex has actually been recently annotated as part of a DASH complex [29]. This indicates that our method was capable to detect the novel biological knowledge which were absent in the reference data.

In fact, as there were seven out of these ten predicted complexes that can be matched with our MIPS protein complex benchmark, we performed further analysis on the remaining three unmatched complexes (ID=2, 9, 10) to see if they are actual novel protein complexes. Our literature search showed that for one of these unmatched predicted complexes (ID=2), 19 out of its 20 protein members were actually part of the "U4/U6 x U5 tri-snRNP complex" (32 proteins) published by Ref. 30. The other predicted 5-protein complex (ID=9) that was not matched with any of our benchmark complexes was found to match 5 out of the 6 proteins in the "mannosyltransferase complex", a protein complex that is responsible for mannosyltransferase activity [31]. Finally, the third unmatched 5-protein complex (ID=10) predicted by DECAFF was also found to correspond directly with a "nuclear condensin complex", a multisubunit protein complex that plays a central role in the condensation of chromosomes that remain in the nucleus [31].

These results show that while some of our predicted complexes do not match with any of our benchmark MIPS complexes (an incomplete reference set), many of them match very well with actual complexes published in biological literature. Our predicted complexes with low p-values are thus likely to be true protein complexes. In fact, this is further supported by matching the predicted complexes with the known protein complexes from the BIND database [32]: more than half of the predicted complexes (673 out of 1055 complexes) from Bader protein interaction data that did not match with any of our MIPS benchmark complexes matched BIND complexes. Similarly, 256 out of the 712 unmatched predicted complexes from the MIPS protein interaction dataset matched BIND complexes, and 825 out of the 1,605 unmatched predicted complexes from BIOGRID protein interaction data matched BIND complexes.

We also investigated why a number of the reference protein complexes in our MIPS benchmark were not matched by any of our complexes predicted by DECAFF. Out of the 215 benchmark MIPS complexes, 157 were matched with a complex predicted by DECAFF using the most comprehensive

Table 2. Ten predicted complexes with different functions from the BIOGRID protein interaction data.

ID	N	δ	P-value	ω	GO ID	Function	ORFs
1	22	0.892	4.95E-54	0.866	GO:0000119	mediator complex	YBL093C, YBR193C, YBR253W, YDL005C, YDR308C, YER022W, YER111C, YGL025C, YGL151W, YGR104C, YHR041C, YHR058C, YLR071C, YMR112C, YNL236W, YNR010W, YOL051W, YOL135C, YOR174W, YPL129W, YPR070W, YPR168W
2	20	0.858	1.17E-45	0.113	GO:0046540	U4/U6 x U5 tri-snRNP complex	YBL026W, YBR055C, YDR378C, YDR473C, YER029C, YER112W, YER172C, YFL017W-A, YGR074W, YGR091W, YHR165C, YJR022W, YKL173W, YLL036C, YLR147C, YNL147W, YOR159C, YOR308C, YPR178W, YPR182W
3	15	1.000	1.61E-43	0.800	GO:0005669	transcription factor TFIID complex	YBR198C, YCR042C, YDR145W, YDR167W, YER148W, YGL112C, YGR274C, YML015C, YML098W, YML114C, YMR005W, YMR227C, YMR236W, YPL011C, YPL129W
4	10	0.844	2.65E-30	0.900	GO:0042729	DASH complex	YBR233W-A, YDR016C, YDR201W, YDR320C-A, YGL061C, YGR113W, YKL052C, YKL138C-A, YKR037C, YKR083C
5	10	0.956	2.10E-28	1.000	GO:0016514	SWI/SNF complex	YBR289W, YDR073W, YHL025W, YJL176C, YMR033W, YNR023W, YOR290C, YPL016W, YPL129W, YPR034W
6	8	1.000	7.37E-25	1.000	GO:0017119	Golgi transport complex	YER157W, YGL005C, YGL223C, YGR120C, YML071C, YNL041C, YNL051W, YPR105C
7	12	0.742	7.16E-19	0.417	GO:0030014	CCR4-NOT complex	YAL021C, YCR093W, YDL165W, YDR443C, YER068W, YER148W, YGR092W, YGR274C, YIL038C, YNL288W, YNR052C, YPR072W
8	6	1.000	9.90E-19	0.667	GO:0030897	HOPS complex	YDL077C, YDR080W, YLR148W, YLR396C, YMR231W, YPL045W
9	5	1.000	5.17E-15	0.000	GO:0031501	mannosyltransferase complex	YDR245W, YEL036C, YJL183W, YJR075W, YPL050C
10	5	1.000	1.28E-15	0.000	GO:0000796	condensin complex	YBL097W, YDR325W, YFR031C, YLR086W, YLR272C

[a] **ID**:complex ID; **N**: the size of complexes; δ: density of complexes; **p-value**: Corrected p-value of complexes; ω: similarity between the predicted complexes and MIPS Benchmark; **GO ID**: the protein GO function ID; **Function**: the protein function with lowest p-value; **ORFs**: proteins' ORFs in complexes.

BIOGRID dataset. 31 of the unmatched 58 reference complexes appeared as individual protein pairs in the BIOGRID interaction graph. Out of remaining 27 unpredicted reference protein complexes, 22 were undetected by DECAFF as they were present in the interaction graph as very sparsely connected subgraphs with a very low average density of 0.178; only 5 reference protein complexes were mistakenly filtered because they were deemed as unreliable protein complexes. We can expect that the performance of DECAFF should improve further with the availability of better PPI detection technologies that can generate more complete PPI data.

5. Conclusions

While much efforts has been expended on charting the protein interactome, the map for the protein "complexome" has remained comparatively empty. In this paper, we have proposed a robust method for exploiting the protein interaction networks to mine for new protein complexes.

Unlike other current computational techniques, our DECAFF algorithm attempts to identify dense and reliable graphical subcomponents in protein interaction networks that could correspond to actual multiprotein complexes. To address the possibility of missing interactions in the underlying interaction network, we have relaxed the graphical constraint from cliques to local dense neighborhoods. The use of local dense neighborhoods as a basis for mining the interaction graphs also allowed us to be certain that maximal dense neighborhoods can always be found under the merging operation (Theorem 1). As such, the main focus is to detect as many local dense graphs as possible to ensure coverage, and to ascertain the reliability of the component interactions as much as possible to ensure accuracy. For the former, we have employed a novel hub-removal procedure that can effectively mine for multiple and possibly overlapping local dense subgraphs for each protein (vertex). This process caters for the biological possibility of a protein participating in multiple protein

complexes. For the latter, we have devised a novel reliability measure to filter away potential false protein complexes in order to address the possibility of false positives in the underlying protein interaction networks.

We evaluated our DECAFF algorithm using three yeast protein interaction data and found that the performance of DECAFF algorithm is indeed significantly better than all the other existing computational techniques. Our current work has shown that both the network topological information and the interaction reliability information in the interaction map can be exploited together to help discover the underlying elements for mapping the complexome. Further resolution and usage of the algorithm will be for mapping out the "protein complex interactome" by uncovering the interacting links between the complexes and the proteins as well as other biomolecules.

Acknowledgments

We thank the anonymous reviewers for their constructive reviews and the early contributions from Mr Soon-Heng Tan.

References

1. R. P. Sear, *Physical Biology* **1**, 53 (2004).
2. A. H. Y. Tong, *Science* **295**, 321(Jan 2002).
3. V. Spirin and L. A. Mirny, *Proc Natl Acad Sci U S A* **100**, 12123(Oct 2003).
4. G. D. Bader and C. W. V. Hogue, *BMC Bioinformatics* **4**, p. 2(Jan 2003), Evaluation Studies.
5. A. D. King, N. Przulj and I. Jurisica, *Bioinformatics* **20**, 3013(Nov 2004), Evaluation Studies.
6. X.-L. Li, S.-H. Tan, C.-S. Foo and S.-K. Ng, *Genome Informatics* **16**, 260(Dec 2005).
7. M. Altaf-Ul-Amin, Y. Shinbo, K. Mihara, K. Kurokawa and S. Kanaya, *BMC Bioinformatics* **7**, 207 (2006).
8. A.-C. Gavin, M. Bosche, R. Krause, P. Grandi, M. Marzioch, A. Bauer, J. Schultz, J. M. Rick, A.-M. Michon, C.-M. Cruciat, M. Remor, C. Hofert, M. Schelder, M. Brajenovic, H. Ruffner, A. Merino, K. Klein, M. Hudak, D. Dickson, T. Rudi, V. Gnau, A. Bauch, S. Bastuck, B. Huhse, C. Leutwein, M.-A. Heurtier, R. R. Copley, A. Edelmann, E. Querfurth, V. Rybin, G. Drewes, M. Raida, T. Bouwmeester, P. Bork, B. Seraphin, B. Kuster, G. Neubauer and G. Superti-Furga, *Nature* **415**, 141(Jan 2002).
9. C. von Mering and Krause, *Nature* **417**, 399(May 2002), Evaluation Studies.
10. E. Hartuv and R. Shamir, *Information Processing Letters* **76**, 175(Dec 2000).
11. S. van Dongen, Graph clustering by flow simulation, PhD thesis, University of Utrecht, (May 2000).
12. D. J. Watts and S. H. Strogatz, *Nature* **393**, 440(Jun 1998).
13. G. Palla, I. Derényi, I. Farkas and T. Vicsek, *Nature* **435**, 814(Jun 2005).
14. E. Ravasz, A. Somera, D. Mongru, Z. Oltvai and A. Barabási, *Science* **297**, 1551(Aug 2002).
15. A.-C. Gavin, P. Aloy, P. Grandi, R. Krause, M. Boesche, M. Marzioch, C. Rau, L. J. Jensen, S. Bastuck, B. Dumpelfeld, A. Edelmann, M.-A. Heurtier, V. Hoffman, C. Hoefert, K. Klein, M. Hudak, A.-M. Michon, M. Schelder, M. Schirle, M. Remor, T. Rudi, S. Hooper, A. Bauer, T. Bouwmeester, G. Casari, G. Drewes, G. Neubauer, J. M. Rick, B. Kuster, P. Bork, R. B. Russell and G. Superti-Furga, *Nature* **440**, 631(Mar 2006).
16. E. Nabieva, K. Jim, A. Agarwal, B. Chazelle and M. Singh, *Bioinformatics* **21 Suppl 1**, 302(Jun 2005).
17. H. N. Chua, W. K. Sung and L. Wong, *Bioinformatics* **22**, 1623(Jul 2006).
18. M. A. Gilchrist, L. A. Salter and A. Wagner, *Bioinformatics* **20**, 689(Mar 2004), Comparative Study.
19. X.-L. Li, Y.-C. Tan and S.-K. Ng, *BMC Bioinformatics* **7 Suppl 4**, p. S23 (2006), Evaluation Studies.
20. P. Uetz and L. Giot, *Nature* **403**, 623(Feb 2000).
21. T. Ito, T. Chiba, R. Ozawa, M. Yoshida, M. Hattori and Y. Sakaki, *Proc Natl Acad Sci U S A* **98**, 4569(Apr 2001).
22. B. L. Drees and Sundin, *J Cell Biol* **154**, 549 (2001).
23. M. Fromont-Racine and A. E. Mayes, *Yeast* **17**, 95(Jun 2000).
24. Y. Ho and Gruhler, *Nature* **415**, 180(Jan 2002).
25. H. W. Mewes and D. Frishman, *Nucleic Acids Res* **28**, 37(Jan 2000).
26. M. C. Costanzo and M. E. Crawford, *Nucleic Acids Res* **29**, 75(Jan 2001).
27. C. Stark, B. J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz and M. Tyers, *Nucleic Acids Res* **34**, 535(Jan 2006).
28. A.-C. Gavin, P. Aloy, P. Grandi, R. Krause, M. Boesche, M. Marzioch, C. Rau, L. J. Jensen, S. Bastuck, B. Dumpelfeld, A. Edelmann, M.-A. Heurtier, V. Hoffman, C. Hoefert, K. Klein, M. Hudak, A.-M. Michon, M. Schelder, M. Schirle, M. Remor, T. Rudi, S. Hooper, A. Bauer, T. Bouwmeester, G. Casari, G. Drewes, G. Neubauer, J. M. Rick, B. Kuster, P. Bork, R. B. Russell and G. Superti-Furga, *Nature* **440**, 631(Mar 2006).
29. J. J. L. Miranda, P. D. Wulf, P. K. Sorger and S. C. Harrison, *Nat Struct Mol Biol* **12**, 138 (2005).
30. S. W. Stevens and J. Abelson, *Proc Natl Acad Sci U S A* **96**, 7226 (1999).
31. J. Jungmann, J. C. Rayner and S. Munro, *J Biol Chem* **274**, 6579 (1999).
32. G. D. Bader, I. Donaldson, C. Wolting, B. F. Ouellette, T. Pawson and C. W. Hogue, *Nucleic Acids Res* **29**, 242(Jan 2001).