k-Partite Cliques of Protein Interactions: A Novel Subgraph Topology for Functional Coherence Analysis on PPI Networks

Qian Liu[a], Yi-Ping Phoebe Chen[b], Jinyan Li*[,a]

[a]*Advanced Analytics Institute, University of Technology Sydney, Sydney, Australia.*
[b]*Department of Computer Science and Computer Engineering, La Trobe University, Melbourne, Australia*

Abstract

Many studies are aimed at identifying dense clusters/subgraphs from protein-protein interaction (PPI) networks for protein function prediction. However, the prediction performance based on the dense clusters are actually worse than a simple guilt-by-association method using neighbour counting ideas. This indicates that the local topological structures and properties of PPI networks are still open to new theoretical investigation and empirical exploration. We introduce a novel topological structure called *k-partite cliques of protein interactions*—a functionally-coherent but not-necessarily-dense subgraph topology in PPI networks—to study PPI networks. A k-partite protein clique is a maximal k-partite clique comprising two or more nonoverlapping protein subsets between any two of which full interactions are exhibited. In the detection of PPI's maximal k-partite cliques, we propose to transform PPI networks into induced $\mathcal{K}$-partite graphs where edges exist only between the partites. Then, we present a maximal k-partite clique mining

Email address: `jinyan.li@uts.edu.au` (Jinyan Li*[,])

(MaCMik) algorithm to enumerate maximal k-partite cliques from $\mathcal{K}$-partite graphs. Our MaCMik algorithm is then applied to a yeast PPI network. We observed interesting and unusually high functional coherence in k-partite protein cliques—the majority of the proteins in k-partite protein cliques, especially those in the same partites, share the same functions, although k-partite protein cliques are not restricted to be dense compared with dense subgraph patterns or (quasi-)cliques. The idea of k-partite protein cliques provides a novel approach of characterizing PPI networks, and so it will help function prediction for unknown proteins.

Keywords:
k-Partite Protein Cliques, $\mathcal{K}$-partite Graphs, Maximal k-Partite Clique, Protein Functional Coherence

1. Introduction

As complete genome sequence data of many organisms become widely available, one of the key challenges in post-genomic biology is to understand and predict protein functions at a proteomic scale. Different approaches have been taken to deal with this challenge. The classical way is to use sequence similarity (King et al., 2001), gene fusion (Anton J. Enright and Ouzounis, 1999), phylogenetic profile (Pellegrini et al., 1999), patterns of gene expression (Zhou et al., 2002), or phenotype data (Clare and King, 2002) to predict protein functions. With the rapid development of high-throughput techniques for the detection of protein-protein interactions (PPIs), large-scale protein interactions' data have been generated recently, and become another abundant resource to study various problems in biological systems (Hu et al., 2012; Li et al., 2012a; Ren et al., 2011; Shen et al., 2010; Zheng et al., 2012; Hu et al., 2011b), especially including the prediction of unknown functions of proteins (Hu et al., 2011a).

PPI data are usually represented by graphs (PPI networks) with proteins standing for vertices and protein interactions for edges. One of the basic questions to study PPI networks is to find biologically relevant functional groups of proteins in PPI networks, i.e. those subgraphs with a high functional coherence (Pandey et al., 2008). A well-known observation is that a protein's direct neighbors may more likely share similar functions with itself than its distant neighbors (Hishigaki et al., 2001; Schwikowski et al., 2000). However, a protein's indirect neighbors can also have substantial function similarity with itself as claimed by Chua *et al.* (Chua et al., 2006)—22.7% yeast proteins actually shared the functions of their exclusively indirect/Level-2 neighbors, while only 1.6% yeast proteins had similar functions to their exclusively Level-1 neighbors. These two seemingly conflicting ideas (Hishigaki et al., 2001; Schwikowski et al., 2000; Chua et al., 2006) actually belong to the scope of neighbour/link-based approaches to the study of PPI networks (Sharan et al., 2007). In general, neighbour-based methods take the functional annotations from interacting neighbours of a target protein for function prediction. Some of these methods simply use local topology of direct and/or indirect neighbours and predict functions of unannotated proteins based on neighbour counting (Schwikowski et al., 2000; Hishigaki et al., 2001; Chua et al., 2006). Some consider the global topology of PPI networks and use graph theoretic methods (Vazquez et al., 2003) such as a flow-based algorithm (Nabieva et al., 2005). Some others design probabilistic approaches such as Markov random field (Kui et al., 2002; Letovsky and Kasif, 2003). A more complicated method is to use multiple networks (Deng et al., 2004; Lee et al., 2006) or multiple other data sources, such as genetic interactions and coexpression interactions (Joshi et al., 2004) and sematic similarity between function classes (Moosavi et al., 2013; Jiang and McQuay, 2012), to enrich the information of PPI networks for neighbour-based functional prediction.

Another direction in the study of PPI networks includes cluster-based methods for the prediction of protein functions (Sharan et al., 2007). The idea is that highly connected protein groups may take part in the same biological process or protein complexes (Bader and Hogue, 2003; Rives and Galitski, 2003; Spirin and Mirny, 2003). From a PPI network, protein clusters are first generated, and then the function information of the protein clusters instead of individual proteins are transferred to those un-annotated proteins. Cluster-based methods can be characterized into several categories with regard to the techniques to detect clusters (Sharan et al., 2007). The first category is based solely on network topology properties (Bader and Hogue, 2003; Sharan et al., 2005; Altaf-Ul-Amin et al., 2006). The second category uses the hierarchical clustering method to locate the protein clusters (Rives and Galitski, 2003; Samanta and Liang, 2003; Maciag et al., 2006). The other categories use non-hierarchical graph clustering methods (Brun et al., 2004), such as Markov clustering algorithm (Enright et al., 2002), highly connected subgraph algorithm (Przulj et al., 2004) and clique percolation (Adamcsek et al., 2006; Han et al., 2007) or dense subgraphs mining methods (Spirin and Mirny, 2003). Other types of biological data can be also combined with PPI networks for cluster detection (Tanay et al., 2004, 2005), including gene expression data (Luscombe et al., 2004), genetic interactions (Kelley and Ideker, 2005), phenotypic profiles (Haugen et al., 2004), semantic similarity of function classes (Zhu et al., 2010) and sometimes multiple PPI networks (Jaeger et al., 2010; Li et al., 2012b).

It is interesting to note that the simple neighbour counting method can outperform the cluster-based methods (Sharan et al., 2007; Song and Singh, 2009). It is also important to note that high-precision function prediction can be achieved for conserved but not necessarily dense modules (Jaeger et al., 2010). These suggest that the local topology of PPI networks contains many new properties to explore for accurate prediction of protein functions.

We propose a novel subgraph topology in this work to conduct a functional coherence analysis on PPI networks. Our idea is to transform a protein interaction network into a $\mathcal{K}$-partite graph. Then we develop a mining algorithm to derive maximal k-partite clique subgraphs, including maximal bicliques/tricliques/quadricliques. All these maximal k-partite cliques, specially termed as k-partite protein cliques, are those subgraphs of the $\mathcal{K}$-partite graph which have full interactions between pairs of these k partites but do not have any connections within each partite. Our k-partite protein cliques cover the topological properties of both protein's direct and indirect neighborhoods. It is true that when the size of the partites is small and the number k of partites is large, maximal k-partite cliques actually mimic dense graphs (Spirin and Mirny, 2003) or clique/quasi-clique patterns (Adamcsek et al., 2006; Han et al., 2007; Bu et al., 2003), which are mostly employed functional information from proteins' direct neighbors. However, our maximal k-partite cliques are not restricted to dense subgraphs especially when the size of partites is large. Thus, our k-partite protein cliques are a novel type of subgraph topology to study protein interaction networks.

The biclique topology concept of protein interactions share similar ideas with the bipartite subgraph definition (Thomas et al., 2003) and the lock-and-key model (Morrison et al., 2006) which are actually originated from complementary domain interactions. An example of the bicliques is about those interactions between proteins containing the classical SH3 domain and the proline-rich peptides (Ren et al., 1993). Bipartite subgraphs have been also applied to the BPM (between-pathway model) motif problems (Brady et al., 2009). Protein triclique topological structure has been also studied before, for example, the tripartite complexes, such as CASK participated CASK-Velis-Mint 1 complex and CASK-Velis-Caskin 1 complex (Tabuchi et al., 2002), and gH-gL-gQ complex and gH-gL-gO complex (Mori et al., 2004). Furthermore, graphical approaches can be used to provide an intuitive picture or useful insights for helping analyzing complicated relations in biological problems (Lin and Lapointe, 2013), as demonstrated by many previous studies on a series of important biological topics, such as enzyme-catalyzed reactions (Chou and Forsen, 1980; Zhou and Deng, 1984; Chou, 1989; Andraos, 2008; Lin and Neet, 1990; Chen et al., 2010a), protein pathway networks (Chen et al., 2010b; Huang et al., 2011), inhibition of HIV-1 reverse transcriptase (Althaus et al., 1993a,b), inhibition kinetics of processive nucleic acid polymerases and nucleases (Chou et al., 1994), drug metabolism systems (Chou, 2010), and using wenxiang diagram or graph (Chou et al., 2011) to study protein-protein interactions (Zhou, 2011a; Kurochkina and Choekyi, 2011; Zhou, 2011b; Zhou and Huang, 2013).

For a $\mathcal{K}$-partite graph in this work, like bipartites, each partite represents a set of vertices of the same kind, and the edges between different partites indicate a certain relationship between those partites. $\mathcal{K}$-partite graphs outweigh the traditional homogeneous graphs in many applications due to that the real-world data usually involves multiple attributes or multiple types of objects and their relationship, such as different functions in PPI networks. Thus, $\mathcal{K}$-partite graphs provide a good proximity of the real-world heterogeneous data. However, mining maximal k-partite cliques from $\mathcal{K}$-partite graphs is at least as hard as NP-hard edge biclique problem, i.e. the problem of finding a maximal weight biclique from an edge weighted graph (Peeters, 2003). Several noteworthy efforts have been taken to obtain useful patterns from $\mathcal{K}$-partite graphs, including biclique model (Li et al., 2007a), quasi-biclique (Li et al., 2008), CLICKS (Zaki and Peters, 2005), star-structure model (Gao et al., 2006), iterative propagation model (Wang et al., 2003) and hidden structure model (Long et al., 2006). However, unlike the research results on bipartite graphs, those methods did not suggest an efficient solution because the problem with the k-partite graphs is more complicated than those of bipartites.

In this work, we design a Maximal k-partite Clique Mining (MaCMik) algorithm by using a divide-and-conquer strategy and a consensus technique; the consensus technique is employed to handle the conflict when partites of maximal k-partite cliques are produced. We apply the MaCMik algorithm to a $\mathcal{K}$-partite graph of a yeast PPI network to detect interesting topological patterns of k-partite protein cliques, such as maximal bicliques, maximal tricliques and maximal quadricliques. These topological patterns of k-partite protein cliques, in particular the partites in k-partite protein cliques, possess high protein functional coherence. We believe that these results can suggest a novel way to understand PPI networks and to help reliable function prediction of proteins. [1]

2. $\mathcal{K}$-partite Graphs and Maximal k-partite Clique Subgraphs

A $\mathcal{K}$-partite graph is denoted by $G = \langle\{V_i, E_{ij} \mid i, j = 1, 2, \cdots, \mathcal{K}, i \neq j\}\rangle$, where V_i (a partite) is a set of vertices, $E_{ij} \subseteq V_i \times V_j$ is a set of edges between V_i and V_j, and $\mathcal{K}$ is the number of the partites in this graph. This definition is similar to that of (Zaki and Peters, 2005). An example of $\mathcal{K}$-partite graphs when $\mathcal{K} = 3$ is shown in Figure 1 where the three partites are V_1, V_2 and V_3 (in blue, red and green respectively), and $E_{1,2}$ are the gradient lines between the magenta and red nodes, $E_{1,3}$ are those between magenta and green, while $E_{2,3}$ are those between green and red.

A k-partite subgraph $G = \langle\{V_i, E_{ij} \mid i, j = 1, 2, \cdots, k, i \neq j\}\rangle$ is a k-partite clique if and only if each $E_{ij} = V_i \times V_j$. We also denote it simply as $\mathcal{G} = \langle\{V_i \mid i = 1, 2, \cdots, k\}\rangle$ by omitting the edges. When $k = 3$ or $k = 4$, $\mathcal{G}$ is specially called a triclique or a quadriclique. In the extreme case of

[1] This work is a substantially revised and updated version of our BIBM 2009 conference paper (Liu et al., 2009). The results are updated significantly based on a new version of a yeast PPI network. The literature review is updated by two thirds. This work is newly compared with traditional dense subgraph approaches, and highlights the advantages of k-partite protein clique approach such as the detection of functional coherent but not necessarily dense subgraphs, an important finding consistent with biological observations (Sharan et al., 2007; Song and Singh, 2009; Jaeger et al., 2010).

$k = 2$, k-partite cliques are exactly bicliques. We also say k-partite cliques to be rank-higher than *(k-1)*-partite cliques. For example, quadricliques are rank-higher than tricliques.

Suppose that $\mathcal{G}' = \langle\{V_i' \mid i = 1, 2, \cdots, k\}\rangle$ is a k-partite clique of G, $\mathcal{G}'$ is a maximal k-partite clique of G if and only if for any proper k-partite clique $\mathcal{G}'' = \langle\{V_i'' \mid i = 1, 2, \cdots, k\}\rangle$ of G, $\mathcal{G}' \subseteq \mathcal{G}''$ is false where $V_i' \subseteq V_i'', \forall i = 1, 2, \cdots, k$. For example, Figure 1 shows a maximal triclique with $V_1 = \{v_{1,1}, v_{1,2}, v_{1,3}, v_{1,4}\}$, $V_2 = \{v_{2,1}, v_{2,2}, v_{2,3}\}$ and $V_3 = \{v_{3,0}, v_{3,1}\}$ in dark blue/red/green. The definition of maximal k-partite cliques implies that, in a k-partite graph, every k-partite clique is an element or covered by an element in the set of maximal k-partite cliques of the k-partite graphs.

These k-partite clique definitions show that k-partite cliques have stringent all-versus-all connection between the pairs of partites. The all-versus-all connection constraint is highly advantageous on a data set with less noises and errors. On some data sets including PPI networks, there are many false positives and false negatives: false negatives will result in missing maximal k-partite cliques (it generally cannot be overcome due to the lack of experimental evidence in PPI networks.), while false positives can make detected maximal k-partite cliques become k-partite quasi-cliques or even meaningless sometimes. Actually, the all-versus-all connection constraint can be relaxed to define k-partite quasi-cliques by a similar way to defining quasi-bicliques in (Li et al., 2008). k-partite quasi-cliques are able to tolerate some noise data. However, this work focuses on the problem of how to mine maximal k-partite cliques from PPI networks.

3. Mining Maximal k-partite Cliques from a PPI Network

To study the functional topology of proteins and their neighborhood proteins, we detect maximal k-partite cliques from PPI networks. Our method consists of three steps: (i) constructing the induced $\mathcal{K}$-partite graph from a PPI network; (ii) designing an algorithm to mine maximal k-partite cliques; (iii) detecting maximal k-partite cliques from $\mathcal{K}$-partite graphs of real-life PPI networks.

3.1. Constructing the $\mathcal{K}$-partite Graphs from PPI Networks

Given a PPI network g, let its maximal size of the maximal cliques be p, then many induced $\mathcal{K}$-partite graphs G can be constructed with not least than p partites. In this work, we only consider the $\mathcal{K}$-partite graphs with a minimum size of partites. It is clear that the minimum size of partites in G is p. Even so, the time complexity to obtain such graphs is $O(N^p)$ where N is the number of vertices in g. Fortunately, the best induced $\mathcal{K}$-partite graphs should be most condense with least partites. That is, the topological patterns should be involved in as less partites as possible. Thus, our heuristic to construct these graphs is that proteins with more partners and the partites with more proteins are considered first. Specifically, we transform a PPI network g into a $\mathcal{K}$-partite graph by using the following process:

i. get the degree number (the number of interacting partners in g) for each protein PP, and rank proteins based on their degree;

ii. produce p empty partites;

iii. add a protein PP with the highest degree into the corresponding partite i if and only if $^{(a)}$ PP has no interaction with any proteins of partite i and $^{(b)}$ partite i has the most proteins among those partites satisfying $^{(a)}$; $i > p$ indicates a newly added partite if no existing partites satisfy $^{(a)}$.

iv. remove PP;

v. repeat (iii) and (iv) until every protein is in a partite. Finally, $\mathcal{K}$ is the number of partites in the $\mathcal{K}$-partite graph produced.

We would like to note that p is the maximal size of the maximal cliques in an original acyclic PPI network, while $\mathcal{K}$ is the number of the partites in a $\mathcal{K}$-partite graph ($\mathcal{K} \geq p$), and k is the number of the partites in a k-partite clique.

3.2. Maximal k-partite Clique Mining (MaCMik) Algorithm

To design the algorithm for mining maximal k-partite cliques from the induced $\mathcal{K}$-partite graph of PPI networks, we first examine a relationship of maximal k-partite cliques $\mathcal{G}(k)$ with its *(k-1)*-partite cliques $\mathcal{G}'(k-1)$. By definition, any *(k-1)*-partite subgraph $G(k-1)$ of $\mathcal{G}(k)$ is a k-partite clique $\mathcal{G}'(k-1)$. That is, $\mathcal{G}'$(k-1) $\subseteq \mathcal{G}(k)$ where each V_i in $\mathcal{G}'$(k-1) $\subseteq V_i$ in $\mathcal{G}(k)$. Thus, if $\mathcal{G}'(k-1)$ does not exist, $\mathcal{G}(k)$ does not exist either; there is no need to produce rank-higher maximal k-partite cliques than $\mathcal{G}'(k-1)$. This observation can be used to prune useless candidate searching when producing rank-high maximal k-partite cliques.

In addition, according to the implication of maximal k-partite cliques in Section 2, every k-partite cliques $\mathcal{G}'(k-1) \subseteq \mathcal{G}(k-1)$, where $\mathcal{G}(k-1)$ are the corresponding maximal *(k-1)*-partite cliques. Therefore, the straightforward method to detect maximal k-partite cliques is to assemble maximal *(k-1)*-partite cliques with the kth-partite, and maximal *(k-1)*-partite cliques can be obtained in the similar way. That is, we can employ a divide-and-conquer strategy to produce maximal k-partite cliques as follows:

(i) obtain maximal *(k-1)*-partite cliques from *(k-1)*-partite graphs of the first *(k-1)* partites.

(ii) detect maximal bicliques from bipartite graphs consisting of the kth-partite and each partite of other *(k-1)*-partites

(iii) merge those maximal bicliques and maximal *(k-1)*-partite cliques together to obtain maximal k-partite cliques.

The way to obtain maximal *(k-1)*-partite cliques is similar to the above process for maximal k-partite cliques. This is a recursive process until *(k-k_i)*-partite graphs, where k_i is the recursive times, are bipartite graphs. Maximal bicliques from bipartite graphs can be detected by the LCM-MBC algorithm (Li et al., 2007a). Thus, there are two vital components in the above process: detecting maximal bicliques and merging maximal bicliques with maximal *(k-1)*-partite cliques.

3.2.1. Detecting Maximal Bicliques

Given a bipartite graph $G = \langle V_1, V_2, E_{12} \rangle$, the LCM-MBC algorithm needs two parameters, q_1 and q_2, to control the minimum number of vertices in each partite of maximal bicliques. This constraint is to avoid producing small and meaningless bicliques. When detecting maximal k-partite cliques, the constraint of the minimum size is much more complicated. More importantly, in k-partite cliques with $k \geq 3$, even if each partite has one vertex, this k-partite clique is still interesting due to that it is a clique in a general graph. Thus, both q_1 and q_2 are set to one here for LCM-MBC. That is, all maximal bicliques are produced by LCM-MBC.

3.2.2. Merging Maximal Bicliques with Maximal (k-1)-partite Cliques

When merging maximal bicliques with maximal *(k-1)*-partite cliques, there may be a conflict between different partitions of the partites which are both in maximal bicliques and in maximal *(k-1)*-partite cliques. For example, given a tripartite graph $G = \langle V_1, V_2, V_3, E_{12}, E_{13}, E_{23} \rangle$, a set of maximal bicliques between V_1 and V_2 or between V_1 and V_3 can be obtained from E_{12} or E_{13}. But the partitions on V_1 by E_{12} and by E_{13} may be partially different, and this is a conflict. In this work, a consensus strategy is used to handle the conflict. That is, only the common vertices in the conflicting partites between maximal *(k-1)*-partite cliques and maximal bicliques will be considered in rank-higher maximal k-partite cliques.

In a k-partite graph G, suppose that $\mathcal{G}(k-1) = \langle \{V_i' \mid i = 1, 2, \cdots, k-1\} \rangle$ is its corresponding maximal *(k-1)*-partite cliques without the kth partite. To get maximal k-partite clique $\mathcal{G}$, the kth partite is merged into $\mathcal{G}(k-1)$ in the following way. Firstly, for partite i in k-partite graphs, $i = 1, 2, \cdots$,k-1, maximal bicliques $\mathcal{G}(ik)_b$ can be obtained from the bipartite graph consisting of partite i and partite k, and V_i and V_{k_i} are vertex sets of $\mathcal{G}(ik)_b$ for partite i and partite k. To handle the difference of $V_{k_i}s, i = 1, 2, \cdots$,k-1, the consensus strategy is used first time. That is, the common vertex set of different V_{k_i}s, $V_k^c = \bigcap_{i=1}^{k-1} V_{k_i}$, will be used as the kth partite of $\mathcal{G}$. Secondly, for partite i

Algorithm 1 Function consensus_Partites: A consensus strategy to produce partites in maximal $\underline{k}$-partite cliques

```
Require: 1) $\mathcal{G}(ik)_b$: maximal bicliques from bipartite graphs consisting of
      the $k$th-partite and the $i$th-partite of $G_k$, $i = 1, 2, \cdots$,k-1
      2) $\mathcal{G}(k-1)$: maximal (k-1)-partite cliques of the (k-1)-partite graph
find the common vertex set of the $k$th-partite of $G_k$, $V_k^c = \bigcap_{i=1}^{k-1} V_{k_i}$ where
      $V_{k_i}$ are the vertices of the $k$th-partite involved in $\mathcal{G}(ik)_b$
if $V_k^c$ is empty then
    there is no $k$-partite clique in $G_k$
else
    for all partite $i$ in $\mathcal{G}(k-1)$ do
        $V_i' =$ vertex sets of partite $i$ involved in maximal (k-1)-partite cliques
            $\mathcal{G}(k-1)$
        $V_i =$ vertex sets of partite $i$ involved in maximal bicliques $\mathcal{G}(ik)_b$
        $V_i^c = V_i' \cap V_i$
        if $V_i^c$ is empty then
            there is no $k$-partite clique in $G_k$
        else
            replace partite $i$ in $\mathcal{G}(k-1)$ with $V_i^c$
        end if
    end for
    maximal $k$-partite cliques $\mathcal{G}(k) = \mathcal{G}(k-1) \cup V_k^c$
    remove redundant $\mathcal{G}(k)$
end if
```

in $\mathcal{G}(k-1)$, the vertex partition in V_i' and V_i may not equal to each other completely. To handle such difference, the consensus strategy is used again. That is, the common vertex set in partite i of $\mathcal{G}(ik)_b$ and of $\mathcal{G}(k-1)$, $V_i^c = V_i' \cap V_i$, $i = 1, 2, \cdots$,k-1, will replace the vertex set of partite i in $\mathcal{G}(k-1)$. Finally, $\mathcal{G}$ can be produced by the assembly of corresponding V_i^cs and V_k^c. If any of V_j in $\mathcal{G}$ is empty, $j = 1, 2, \cdots, k$, $\mathcal{G}$ does not exist; that is, there is no maximal k-partite cliques for k-partite graphs G. There are some redundant maximal k-partite cliques to be removed. However, this method guarantees to produce the complete set of maximal k-partite cliques. The pseudo code of this method is shown in Algorithm 1.

Algorithm 2 gives the entire pseudo code of our MaCMik algorithm to mine the maximal k-partite cliques.

3.3. Detecting Maximal k-partite Cliques from PPI Networks

We apply the MaCMik algorithm to the induced $\mathcal{K}$-partite graphs of PPI networks to detect maximal k-partite cliques. As the time complexity and space complexity are too high to obtain rank-higher k-partite cliques, we consider only maximal bicliques, maximal tricliques and maximal quadricliques of PPI networks in this work.

4. Functional Coherence in k-partite Protein Cliques of a Yeast PPI Network

The dataset under our test and evaluation is the DIP (Database of Interacting Proteins) yeast PPI network (the January 31, 2013 release). This PPI network contains 4,892 proteins with identical OLN (Ordered Locus

Algorithm 2 Maximal k-partite Cliques Mining (MaCMik) Algorithm

```
Require: PPI network $g$
convert $g$ into an induced $\mathcal{K}$-partite graph $G$ with as less partites as possible.
use LCM-MBC to mine maximal bicliques $\mathcal{G}(ij)_b$ for any pair of partites $i$ and $j$ in $G$, $i, j = 1, 2, \cdots, k, i < j$
for all $k$ from 3 to $\mathcal{K}$ do
    set maximal $k$-partite cliques $\mathcal{G}(k)$={}
    for all $k$-partite graph $G_k$ in $G$ do
        assume that $\mathcal{G}$(k-1)=$\langle\{V_i'|i = 1, 2, \cdots$,k-1$\}\rangle$ are maximal (k-1)-partite cliques of the (k-1)-partite graph with the first (k-1) partites of $G_k$
        assume that $\mathcal{G}(ik)_b$ are maximal bicliques from bipartite graphs consisting of the $k$th-partite and the $i$th-partite of $G_k$, $i = 1, 2, \cdots$,k-1
        get maximal $k$-partite cliques through Function consensus_Partites($\mathcal{G}$(k-1), $\mathcal{G}(ik)_b$) in Algorithm 1, and add them into $\mathcal{G}(k)$.
    end for
    output maximal $k$-partite cliques $\mathcal{G}(k)$
end for
```

Names) mapping in UniProt and 21,851 non self-interactions. This network was transformed into a 12-partite graph G. Our MaCMik algorithm was then applied and detected 76,409 maximal tricliques and 53,462 quadricliques together with 15,740 maximal bicliques from G. We further studied the functional coherence of proteins in these biclique, triclique and quadriclique patterns. (Bicliques with less than three proteins were excluded from our analysis due to that they were more likely noise patterns.)

In the examination of protein functional coherence, we made use of a functional annotation scheme, the FunCat 2.1 functional classification scheme (Ruepp et al., 2004), which was downloaded from the Comprehensive Yeast Genome Database of the Munich Information Center for Protein Sequences (MIPS). The FunCat scheme is organized like a tree structure with up to six levels of increasing specificity. In this work, the root of FunCat is referred to as Level 0; its children are referred to as Level 1, etc. That is, Level L's children are referred to as Level L+1, where $L = 0, 1, \cdots, 5$. Level 6 has no child. Our functional coherence evaluation is based on Level 1 and Level 2 only. Level 1 functions cover 18 categories (including the category of unknown functions) in the coarse-grained level, and Level 2 functions spread on 80 categories. In Level 1, there are 956 proteins with unknown functions on the yeast PPI network, and among them, 38 proteins cannot be assigned into any k-partite protein cliques (mainly maximal bicliques).

4.1. Functional Coherence in k-partite Protein Cliques and in Their Partites

The functional coherence was examined not only on entire maximal k-partite cliques, also on their separate partites at the two levels of functional specificity. In our evaluation of the functional coherence on the partites, those partites with only one protein were not considered, as their functional coherence deems to be 100%. Figure 2 displays the distribution of the functional coherence, where the horizontal axis represents the size of maximal k-partite cliques, and the vertical axis represents the average percentage distribution of the functions shared in the maximal k-partite cliques (in Figure 2a and

Figure 2b) or in their partites (in Figure 2c and Figure 2d) with the same k-partite clique size. In Figure 2, the lines with 'plus' signs represent the percentage of the main functions which are shared by the majority proteins in the maximal k-partite cliques or in their partites, and the lines with 'crosses' represent the percentage of the discordant functions distributed among the remaining proteins with known functions, and the lines with 'circles' represent the percentage of proteins with unknown functions. The red, green and blue colors are for the maximal bicliques, maximal tricliques, and maximal quadricliques, respectively.

Figure 2a and Figure 2b show a clear picture that most of the proteins in the same maximal k-partite cliques share the same functions. They also show that the functional coherence in the quadricliques is generally higher than that of tricliques which in turn is generally higher than that of bicliques. Meanwhile, it can be seen that the k-partite cliques with smaller size, such as protein size from 3 to 6, more likely share the same functions than those with a bigger size. Another interesting point is that the k-partite cliques of small size are actually quasi-cliques in general graphs. This is in agreement to the research results by (Bu et al., 2003) which claimed that many quasi-cliques in PPI networks share the same functions.

Again in (Bu et al., 2003), the quasi-partites were detected by using negative eigenvalues in spectral analysis. Because of difficulties in spectral analysis, the functional coherence of quasi-partites, especially of their partites, was not comprehensively studied in (Bu et al., 2003). In this work, it is easy to examine whether the proteins in the same partites of maximal k-partite cliques share the same functions or not. As shown in Figure 2c and Figure 2d, most proteins in the same partites of the maximal k-partite cliques share the same functions.

Through our investigation and analysis on the data presented in Figure 2, the following interesting points can be summarized:

(i) maximal k-partite cliques with a smaller size are more likely to share the same functions than k-partite cliques with a bigger size. The maximal k-partite cliques of protein sizes from 3 to 6, which actually correspond to quasi-cliques of (Bu et al., 2003) in general PPI graphs, may be biologically relevant functional groups.

(ii) rank-higher maximal k-partite cliques contain more functional coherence information than rank-lower *(k-1)*-partite cliques.

(iii) the separate partites in k-partite cliques have higher function coherence than those of the entire k-partite cliques, and those partites in maximal k-partite cliques may also be biologically relevant functional groups. This observation is consistent with an earlier work (Chua et al., 2006) which claimed that the fraction of indirect neighbor partners sharing the same functions was much higher than the fraction of interacting proteins.

4.2. Examples of Maximal k-partite Cliques with High Functional Coherence

We highlight six examples (Table 1 and Table 2) to detail the high functional coherence in maximal k-partite cliques and in their partites under the level 2 functions. In Table 1, 75% of proteins in the biclique example have the same functions 20.01—transported compounds (substrates). The function 12.01 (ribosome biogenesis) and 42.16 (mitochondrion) are 100% shared by the proteins in the triclique whose topology is shown in Figure 3. The function 14.13 (protein degradation) is also 100% shared in the quadriclique proteins. In the example of the biclique as shown in Figure 4, the two proteins YHR140W and YHL042W are not annotated with any function, while the protein YEL002C has three different functions rather than the function 12.01. However, one of its functions in Level 2 is '01.05: C-compound and carbohydrate metabolism', while the main function 20.01 in the biclique includes a more specific function 'C-compound and carbohydrate transport'. Thus, the discordant function 01.05 is, although not the same as, closely related to the main function 20.01 in the example biclique. Thus, the two unclassified proteins YHR140W and YHL042W are more likely to share the functions which are closely related to the main function 20.01. Meanwhile, the cellular component of YHL042W should include '(mitochondrial outer) membrane' according to Saccharomyces Genome Database (SGD), while the cellular components of all other proteins include ”Membrane”, ”Integral To Membrane” or ”Nuclear Membrane/Envelope” according to BioGRID. All the proteins in the biclique could be in a same cellular component. The cellular component information is a good indicator, although not a direct functional evidence, of protein functions for YHR140W and YHL042W.

Table 2 indicates that separate partites can have a higher percentage of proteins sharing the same functions than the entire k-partite clique. The mainly shared functions within partites may also differ from the one mainly shared by proteins in the corresponding k-partite clique. In the example biclique of Table 2, the main functions in the biclique is 32.01 (stress response) shared by 70% of the proteins(7 out of 10); however, the functions 32.01 and

14.01 (protein folding and stabilization) are both shared by all five (100%) proteins in the partite 1. The similar conclusion can be drawn from the example triclique and quadriclique of Table 2. In the quadriclique, YBR159W, YCR034W and YLR372W from partite 1, YPL076W from partite 2, and YGR060W from partite 3 form a quasi-clique to share the function 01.06 (lipid, fatty acid and isoprenoid metabolism), which is different from the main function 20.09 (transport routes) shared by the majority of proteins in the quadriclique.

Based on these analysis, we note that, on one hand, our method can detect, as done by previous works (Adamcsek et al., 2006; Han et al., 2007), interesting cliques or dense subgraphs that share the similar functions when the k-partite cliques have small size of proteins; on the other hand, our method is able to detect those k-partite cliques with large size partites which are not necessary to be dense. What is more interesting is that k-partite cliques with both small size and large size partites and especially their partites have a high functional coherence. This provides a possible explanation why cluster-based methods for functional prediction are not able to outperform simple guilt-by-association predictions: cluster-based methods generally detect dense subgraphs in PPI networks and cannot capture non-dense but functionally similar subgraphs such as k-partite cliques with large size partites. Thus, the novel approach of k-partite cliques provides new insights into the topological structure of biologically relevant functional groups in PPI networks.

5. Conclusion and Future Works

In this work, we have proposed to use k-partite clique subgraphs to characterize biologically relevant functional groups of proteins in a PPI network. We have proposed to transform protein interaction networks into $\mathcal{K}$-partite graphs for mining maximal k-partite cliques by our MaCMik algorithm. Our investigation and analysis on k-partite clique subgraphs show that proteins in a k-partite clique often have a high functional coherence and the separate partites in a k-partite clique are also highly to be in biologically relevant functional groups.

As a future work, we will improve the idea of k-partite protein cliques from several aspects. Firstly, statistical evaluation is an option to pinpoint out the biologically most relevant functional groups from k-partite clique subgraphs. Secondly, as PPI networks contain both false negative and false positive interactions, k-partite quasi-cliques, which relax the strict all-versus-all interaction constraint imposed by k-partite cliques, may overcome the problem of false negative interactions. Thirdly, a score method (Li et al., 2007b) based on the reliability of different experiments and detection times of the interactions is also helpful to eliminate the effect of false positive interactions. Fourthly, a framework will be designed to predict functions of proteins using k-partite cliques. Finally, we will make efforts to provide a web-server for the method presented in this work, since user-friendly and publicly accessible web-servers represent the future direction for developing practically more useful models, simulated methods or predictors (Chou and Shen, 2009).

References

Adamcsek, B., Palla, G., Farkas, I. J., Derenyi, I., Vicsek, T., 2006. CFinder: locating cliques and overlapping modules in biological networks. Bioinformatics 22 (8), 1021–1023.

Altaf-Ul-Amin, M., Shinbo, Y., Mihara, K., Kurokawa, K., Kanaya, S., 2006. Development and implementation of an algorithm for detection of protein complexes in large interaction networks. BMC Bioinformatics 7 (1), 207.

Althaus, I., Chou, J., Gonzales, A., Deibel, M., Chou, K., Kezdy, F., Romero, D., Aristoff, P., Tarpley, W., Reusser, F., 1993a. Steady-state kinetic studies with the non-nucleoside HIV-1 reverse transcriptase inhibitor U-87201E. J Biol Chem 268 (9), 6119–24.

Althaus, I., Gonzales, A., Chou, J., Romero, D., Deibel, M., Chou, K., Kezdy, F., Resnick, L., Busso, M., So, A., 1993b. The quinoline U-78036 is a potent inhibitor of HIV-1 reverse transcriptase. J Biol Chem 268 (20), 14875–80.

Andraos, J., 2008. Kinetic plasticity and the determination of product ratios for kinetic schemes leading to multiple products without rate laws new methods based on directed graphs. Canadian Journal of Chemistry 86 (4), 342–357.

Anton J. Enright, Ioannis Iliopoulos, N. C. K., Ouzounis, C. A., 1999. Protein interaction maps for complete genomes based on gene fusion events. Nature 402, 86–90.

Bader, G., Hogue, C., 2003. An automated method for finding molecular complexes in large protein interaction networks. BMC Bioinformatics 4 (1), 2.

Brady, A., Maxwell, K., Daniels, N., Cowen, L. J., 2009. Fault tolerance in protein interaction networks: Stable bipartite subgraphs and redundant pathways. PLoS ONE 4 (4), e5364.

Brun, C., Herrmann, C., Guénoche, A., 2004. Clustering proteins from interaction networks for the prediction of cellular functions. BMC Bioinformatics 5, 95.

Bu, D., Zhao, Y., Cai, L., Xue, H., Zhu, X., Lu, H., Zhang, J., Sun, S., Ling, L., Zhang, N., Li, G., Chen, R., 2003. Topological structure analysis of the protein-protein interaction network in budding yeast. Nucl. Acids Res. 31 (9), 2443–2450.

Chen, L., Feng, K.-Y., Cai, Y.-D., Chou, K.-C., Li, H.-P., 2010a. Predicting the network of substrate-enzyme-product triads by combining compound similarity and functional domain composition. BMC Bioinformatics 11 (1), 293.

Chen, L., Huang, T., Shi, X.-H., Cai, Y.-D., Chou, K.-C., 2010b. Analysis of protein pathway networks using hybrid properties. Molecules 15 (11), 8177–8192.

Chou, K., 1989. Graphic rules in steady and non-steady state enzyme kinetics. J Biol Chem. 264 (20), 12074–9.

Chou, K., 2010. Graphic rule for drug metabolism systems. Curr Drug Metab. 11 (4), 369–78.

Chou, K., Forsen, S., 1980. Graphical rules for enzyme-catalysed rate laws. Biochem J. 187 (3), 829–35.

Chou, K., Kezdy, F., Reusser, F., 1994. Review: Steady-state inhibition kinetics of processive nucleic acid polymerases and nucleases. Analytical Biochemistry 221, 217–230.

Chou, K., Lin, W.-Z., Xiao, X., 2011. Wenxiang: a web-server for drawing wenxiang diagrams. Natural Science 3 (10), 862–865.

Chou, K.-C., Shen, H.-B., 2009. REVIEW: Recent advances in developing web-servers for predicting protein attributes. Natural Science 1 (2), 63–92.

Chua, H. N., Sung, W.-K., Wong, L., 2006. Exploiting indirect neighbours and topological weight to predict protein function from protein–protein interactions. Bioinformatics 22 (13), 1623–1630.

Clare, A., King, R. D., 2002. Machine learning of functional class from phenotype data. Bioinformatics 18 (1), 160–166.

Deng, M., Tu, Z., Sun, F., Chen, T., 2004. Mapping gene ontology to proteins based on proteinprotein interaction data. Bioinformatics 20 (6), 895–902.

Enright, A. J., Van Dongen, S., Ouzounis, C. A., 2002. An efficient algorithm for large-scale detection of protein families. Nucleic Acids Research 30 (7), 1575–1584.

Gao, B., Liu, T.-Y., Ma, W.-Y., 2006. Star-structured high-order heterogeneous data co-clustering based on consistent information theory. In: ICDM '06. IEEE Computer Society, Washington, DC, USA, pp. 880–884.

Han, K., Cui, G., Chen, Y., 2007. Identifying functional groups by finding cliques and near-cliques in protein interaction networks. In: FBIT. pp. 159–164.

Haugen, A., Kelley, R., Collins, J., Tucker, C., Deng, C., Afshari, C., Brown, J. M., Ideker, T., Van Houten, B., 2004. Integrating phenotypic and expression profiles to map arsenic-response networks. Genome Biology 5 (12), R95.

Hishigaki, H., Nakai, K., Ono, T., Tanigami, A., Takagi, T., 2001. Assessment of prediction accuracy of protein function from protein–protein interaction data. Yeast 18 (6), 523–531.

Hu, L., Huang, T., Shi, X., Lu, W.-C., Cai, Y.-D., Chou, K.-C., 2011a. Predicting functions of proteins in mouse based on weighted protein-protein interaction network and protein hybrid properties. PLoS ONE 6 (1), e14556.

Hu, L.-L., Feng, K.-Y., Cai, Y.-D., Chou, K.-C., 2012. Using protein-protein interaction network information to predict the subcellular locations of proteins in budding yeast. Protein and Peptide Letters 19 (6), 644–651.

Hu, L.-L., Huang, T., Cai, Y.-D., Chou, K.-C., 2011b. Prediction of body fluids where proteins are secreted into based on protein interaction network. PLoS ONE 6 (7), e22989.

Huang, T., Chen, L., Cai, Y.-D., Chou, K.-C., 2011. Classification and analysis of regulatory pathways using graph property, biochemical and physicochemical property, and functional property. PLoS ONE 6 (9), e25297.

Jaeger, S., Sers, C., Leser, U., 2010. Combining modularity, conservation, and interactions of proteins significantly increases precision and coverage of protein function prediction. BMC Genomics 11 (1), 717.

Jiang, J. Q., McQuay, L. J., 2012. Predicting protein function by multi-label correlated semi-supervised learning. IEEE/ACM Transactions on Computational Biology and Bioinformatics 9 (4), 1059–1069.

Joshi, T., Chen, Y., Becker, J. M., Alexandrov, N., Xu, D., 2004. Genome-scale gene function prediction using multiple sources of high-throughput data in yeast saccharomyces cerevisiae. OMICS 8 (4), 322–333.

Kelley, R., Ideker, T., 2005. Systematic interpretation of genetic interactions using protein networks. Nat Biotech 23, 561–566.

King, R. D., Karwath, A., Clare, A., Dehaspe, L., 2001. The utility of different representations of protein sequence for predicting functional class. Bioinformatics 17 (5), 445–454.

Kui, M. D., Zhang, K., Mehta, S., Chen, T., Sun, F., 2002. Prediction of protein function using protein-protein interaction data. Journal of Computational Biology 10, 947–960.

Kurochkina, N., Choekyi, T., 2011. Helixhelix interfaces and ligand binding. Journal of Theoretical Biology 283 (1), 92–102.

Lee, H., Tu, Z., Deng, M., Sun, F., Chen, T., 2006. Diffusion kernel-based logistic regression models for protein function prediction. Omics : a journal of integrative biology 10 (1), 40–55.

Letovsky, S., Kasif, S., 2003. Predicting protein function from protein/protein interaction data: a probabilistic approach. Bioinformatics 19 (suppl 1), i197–i204.

Li, B.-Q., Huang, T., Liu, L., Cai, Y.-D., Chou, K.-C., 2012a. Identification of colorectal cancer related genes with mRMR and shortest path in protein-protein interaction network. PLoS ONE 7 (4), e33393.

Li, J., Liu, G., Li, H., Wong, L., 2007a. Maximal biclique subgraphs and closed pattern pairs of the adjacency matrix: A one-to-one correspondence and mining algorithms. IEEE Trans. Knowl. Data Eng. 19 (12), 1625–1637.

Li, J., Sim, K., Liu, G., Wong, L., 2008. Maximal quasi-bicliques with balanced noise tolerance: Concepts and co-clustering applications. In: SDM. pp. 72–83.

Li, M., Wu, X., Wang, J., Pan, Y., 2012b. Towards the identification of protein complexes and functional modules by integrating ppi network and gene expression data. BMC Bioinformatics 13 (1), 109.

Li, X.-L., Foo, C.-S., Ng, S.-K., 2007b. Discovering protein complexes in dense reliable neighborhoods of protein interaction networks. In: LSS 6th Annual International Conference on Computational Systems Bioinformatics. pp. 157–168.

Lin, S., Lapointe, J., 2013. Theoretical and experimental biology in one — a symposium in honour of Professor Kuo-Chen Chous 50th anniversary and Professor Richard Gieges 40th anniversary of their scientific careers. Journal of Biomedical Science and Engineering 6 (4), 435–442.

Lin, S. X., Neet, K. E., 1990. Demonstration of a slow conformational change in liver glucokinase by fluorescence spectroscopy. J Biol Chem 265 (17), 9670–9675.

Liu, Q., Chen, Y.-P. P., Li, J., 2009. High functional coherence in k-partite protein cliques of protein interaction networks. In: BIBM. IEEE Computer Society, pp. 111–117.

Long, B., Wu, X., Zhang, Z. M., Yu, P. S., 2006. Unsupervised learning on k-partite graphs. In: KDD. pp. 317–326.

Luscombe, N. M., Madan Babu, M., Yu, H., Snyder, M., Teichmann, S. A., Gerstein, M., 2004. Genomic analysis of regulatory network dynamics reveals large topological changes. Nature 431 (7006), 308–312.

Maciag, K., Altschuler, S. J., Slack, M. D., Krogan, N. J., Emili, A., Greenblatt, J. F., Maniatis, T., Wu, L. F., 2006. Systems-level analyses identify extensive coupling among gene expression machines. Mol Syst Biol 2 (1), msb4100045–E1–msb4100045–E14.

Moosavi, S., Rahgozar, M., Rahimi, A., 2013. Protein function prediction using neighbor relativity in protein-protein interaction network. Computational Biology and Chemistry 43 (0), 11–16.

Mori, Y., Akkapaiboon, P., Yonemoto, S., Koike, M., Takemoto, M., Sadaoka, T., Sasamoto, Y., Konishi, S., Uchiyama, Y., Yamanishi, K., 2004. Discovery of a Second Form of Tripartite Complex Containing gH-gL of Human Herpesvirus 6 and Observations on CD46. J. Virol. 78 (9), 4609–4616.

Morrison, J. L., Breitling, R., Higham, D. J., Gilbert, D. R., 2006. A lock-and-key model for protein–protein interactions. Bioinformatics 22 (16), 2012–2019.

Nabieva, E., Jim, K., Agarwal, A., Chazelle, B., Singh, M., 2005. Whole-proteome prediction of protein function via graph-theoretic analysis of interaction maps. Bioinformatics 21 Suppl 1.

Pandey, J., Koyuturk, M., Subramaniam, S., Grama, A., 2008. Functional coherence in domain interaction networks. Bioinformatics 24 (16), i28–34.

Peeters, R., 2003. The maximum edge biclique problem is NP-complete. Discrete Applied Mathematics 131, 651–654.

Pellegrini, M., Marcotte, E. M., Thompson, M. J., Eisenberg, D., Yeates, T. O., 1999. Assigning protein functions by comparative genome analysis: protein phylogenetic profiles. PNAS 96 (8), 4285–4288.

Przulj, N., Wigle, D., Jurisica, I., 2004. Functional topology in a network of protein interactions. Bioinformatics 20 (3), 340–348.

Ren, L.-H., Shen, Y.-Z., Ding, Y.-S., Chou, K.-C., 2011. Bio-entity network for analysis of proteinprotein interaction networks. Asian Journal of Control 13 (5), 726–737.

Ren, R., Mayer, B., Cicchetti, P., Baltimore, D., 1993. Identification of a ten-amino acid proline-rich SH3 binding site. Science 259 (5098), 1157–1161.

Rives, A. W., Galitski, T., 2003. Modular organization of cellular networks. Proceedings of the National Academy of Sciences 100 (3), 1128–1133.

Ruepp, A., Zollner, A., Maier, D., Albermann, K., Hani, J., Mokrejs, M., Tetko, I., Güldener, U., Mannhaupt, G., Münsterkötter, M., Mewes, H. W., 2004. The FunCat, a functional annotation scheme for systematic classification of proteins from whole genomes. Nucleic Acids Res 32 (18), 5539–5545.

Samanta, M. P., Liang, S., 2003. Predicting protein functions from redundancies in large-scale protein interaction networks. Proceedings of the National Academy of Sciences 100 (22), 12579–12583.

Schwikowski, B., Uetz, P., Fields, S., 2000. A network of protein-protein interactions in yeast. Nat Biotechnol 18 (12), 1257–1261.

Sharan, R., Ideker, T., Kelley, B., Shamir, R., Karp, R. M., 2005. Identification of protein complexes by comparative analysis of yeast and bacterial protein interaction data. Journal of computational biology 12 (6), 835–846.

Sharan, R., Ulitsky, I., Shamir, R., 2007. Network-based prediction of protein function. Molecular Systems Biology 3 (1).

Shen, Y.-Z., Ding, Y.-S., Gu, Q., Chou, K.-C., 2010. Identifying the hub proteins from complicated membrane protein network systems. Medicinal Chemistry 6 (3), 165–173.

Song, J., Singh, M., 2009. How and when should interactome-derived clusters be used to predict functional modules and protein function? Bioinformatics 25 (23), 3143–3150.

Spirin, V., Mirny, L. A., 2003. Protein complexes and functional modules in molecular networks. PNAS 100 (21), 12123–12128.

Tabuchi, K., Biederer, T., Butz, S., Sudhof, T. C., 2002. CASK participates in alternative tripartite complexes in which Mint 1 competes for binding with caskin 1, a novel CASK-binding protein. J Neurosci 22, 4264–4273.

Tanay, A., Sharan, R., Kupiec, M., Shamir, R., 2004. Revealing modularity and organization in the yeast molecular network by integrated analysis of highly heterogeneous genomewide data. Proceedings of the National Academy of Sciences of the United States of America 101 (9), 2981–2986.

Tanay, A., Steinfeld, I., Kupiec, M., Shamir, R., 2005. Integrative analysis of genome-wide experiments in the context of a large high-throughput data compendium. Molecular Systems Biology 1 (msb4100005), E1–E10.

Thomas, A., Cannings, R., Monk, N. A., Cannings, C., 2003. On the structure of protein-protein interaction networks. Biochem Soc Trans 31 (Pt 6), 1491–1496.

Vazquez, A., Flammini, A., Maritan, A., Vespignani, A., 2003. Global protein function prediction from protein-protein interaction networks. Nat Biotechnol 21, 697–700.

Wang, J., Zeng, H., Chen, Z., Lu, H., Tao, L., Ma, W.-Y., 2003. ReCoM:

reinforcement clustering of multi-type interrelated data objects. In: SIGIR '03. ACM, New York, NY, USA, pp. 274–281.

Zaki, M. J., Peters, M., 2005. CLICKS: Mining subspace clusters in categorical data via k-partite maximal cliques. Data Engineering, International Conference on 0, 355–356.

Zheng, L.-L., Li, Y.-X., Ding, J., Guo, X.-K., Feng, K.-Y., Wang, Y.-J., Hu, L.-L., Cai, Y.-D., Hao, P., Chou, K.-C., 2012. A comparison of computational methods for identifying virulence factors. PLoS ONE 7 (8), e42517.

Zhou, G., Deng, M., 1984. An extension of Chou's graphic rules for deriving enzyme kinetic equations to systems involving parallel reaction pathways. Biochem J. 222 (1), 169–76.

Zhou, G.-P., 2011a. The disposition of the LZCC protein residues in wenxiang diagram provides new insights into the proteinprotein interaction mechanism. Journal of Theoretical Biology 284 (1), 142–148.

Zhou, G. P., 2011b. The structural determinations of the leucine zipper coiled-coil domains of the cGMP-dependent protein kinase i alpha and its interaction with the myosin binding subunit of the myosin light chains phosphase. Protein and peptide letters 18 (10), 966–978.

Zhou, G.-P., Huang, R.-B., 2013. The pH-triggered conversion of the PrPc to PrPsc. Current Topics in Medicinal Chemistry 13 (10), 1152–1163.

Zhou, X., Kao, M. C., Wong, W. H., 2002. Transitive functional annotation by shortest-path analysis of gene expression data. PNAS 99 (20), 12783–12788.

Zhu, W., Hou, J., Chen, Y.-P. P., 2010. Semantic and layered protein function prediction from PPI networks. Journal of Theoretical Biology 267 (2), 129–136.

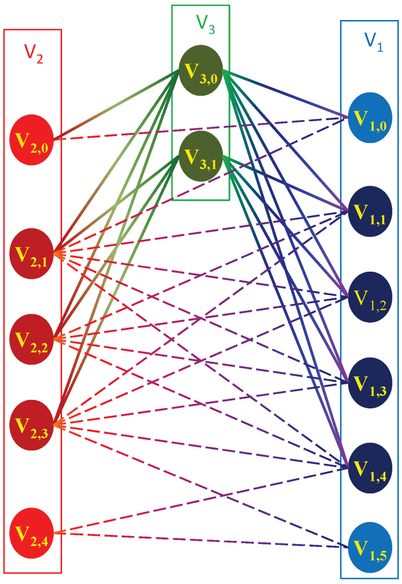
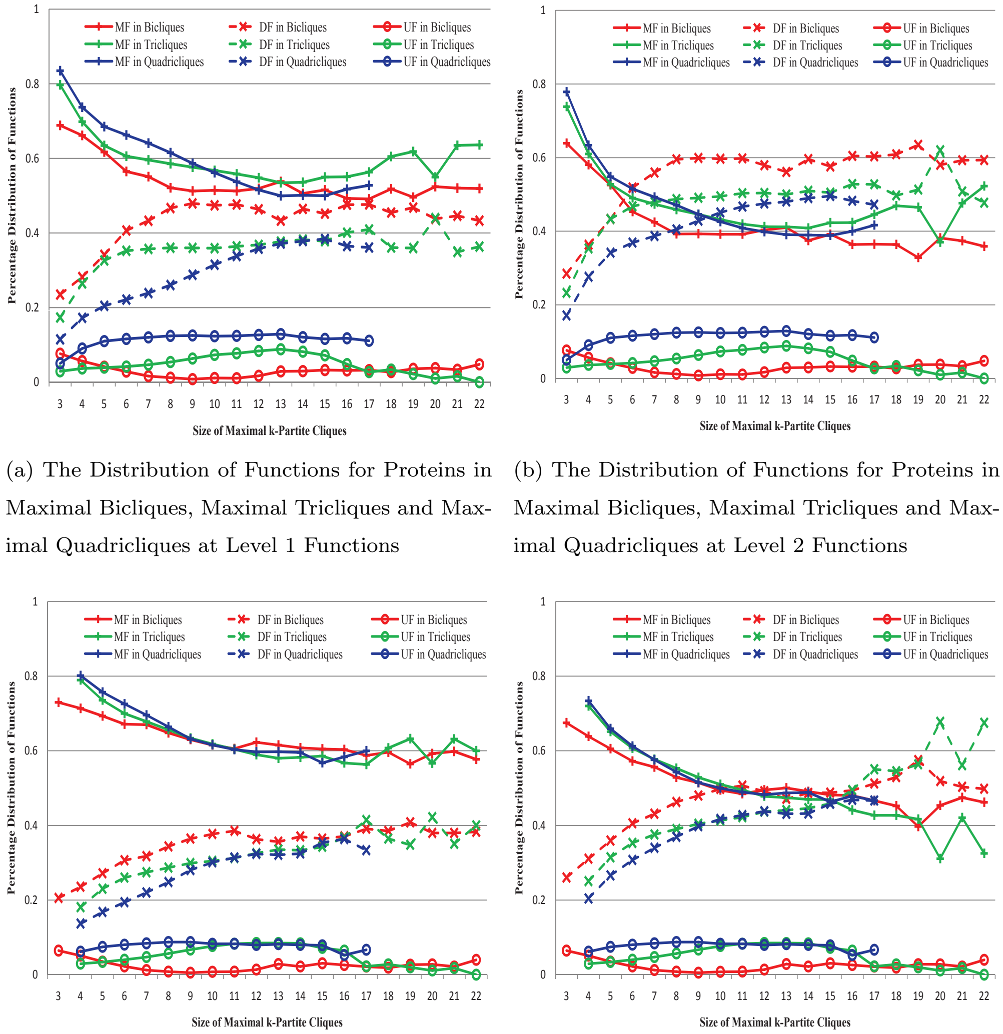


Figure 1: An example of a tripartite graph G and its maximal tricliques (best viewed in color). The three partites, V_1, V_2, and V_3, are in (dark) blue, (dark) red, and green respectively. All vertices in each rectangle belong to the same partite, while gradient-colors lines represent interactions between vertices. The vertices in dark red/green/blue form a maximal triclique in G.



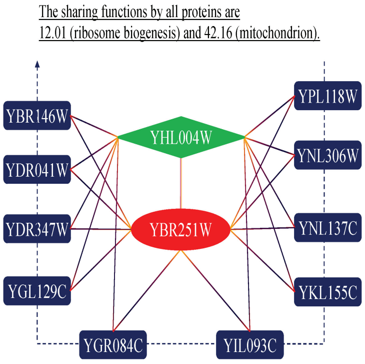
(a) The Distribution of Functions for Proteins in Maximal Bicliques, Maximal Tricliques and Maximal Quadricliques at Level 1 Functions

(b) The Distribution of Functions for Proteins in Maximal Bicliques, Maximal Tricliques and Maximal Quadricliques at Level 2 Functions

(c) The Distribution of Functions for Proteins in Each Partite of Maximal Bicliques, Maximal Tricliques and Maximal Quadricliques at Level 1 Functions

(d) The Distribution of Functions for Proteins in Each Partite of Maximal Bicliques, Maximal Tricliques and Maximal Quadricliques at Level 2 Functions

Figure 2: Comparison of Functional Coherence in Maximal Bicliques/Tricliques/Quadricliques (best viewed in color). MF, DF and UF represent Main Functions, Discordant Functions and Unknown Functions respectively in maximal k-partite cliques.



The sharing functions by all proteins are
12.01 (ribosome biogenesis) and 42.16 (mitochondrion).
YBR146W
YDR041W
YDR347W
YGL129C
YHL004W
YBR251W
YPL118W
YNL306W
YNL137C
YKL155C
YGR084C
YIL093C

Figure 3: An example of the triclique in Table 1 where proteins in red ellipse, green diamond and dark blue rectangles represent three partites.

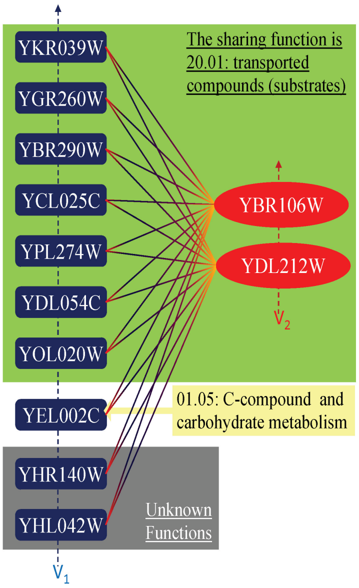


Figure 4: An example of the biclique in Table 1 where proteins in dark blue rectangles have full connections with proteins in red ellipses, and the regions in green, yellow and dark gray show the functional information.

Table 1: Three Examples of Protein Function Coherence in Maximal Bicliques/Tricliques/Quadricliques. The **boldface numbers** are the main functions shared by the majority proteins in the corresponding *k*-partite cliques. 99 indicates 'UNCLASSIFIED PROTEINS'.

k-partite cliques	Partites	Proteins	Functions
Biclique	1	YBR106W	01.04.04/**20.01**.01.07.07
		YDL212W	14.04/14.07/16.01/**20.01**.07/ 20.09.07.03/30.01/32.01.07
	2	YKR039W	**20.01**.07/20.09.18
		YHR140W	99
		YGR260W	**20.01**.23/**20.01**.25
		YHL042W	99
		YBR290W	14.04/**20.01**.01.01/ 20.09.13/34.01.01.01
		YEL002C	01.05.03.02.04/10.03/14.07.02.02
		YCL025C	**20.01**.07
		YPL274W	**20.01**.03/**20.01**.07
		YDL054C	**20.01**.03/20.03
		YOL020W	**20.01**.07/20.09.18
Triclique	1	YBR251W	**12.01.01/42.16**
	2	YBR146W	**12.01.01/42.16**
		YDR041W	**12.01.01/42.16**
		YDR347W	**12.01.01/42.16**
		YGL129C	**12.01.01/42.16**
		YGR084C	**12.01.01/42.16**
		YIL093C	**12.01.01/42.16**
		YKL155C	**12.01.01/42.16**
		YNL137C	**12.01.01/42.16**
		YNL306W	**12.01.01/42.16**
		YPL118W	**12.01.01/42.16**/12.04.01
	3	YHL004W	**12.01.01/42.16**/16.03.03
Quadriclique	1	YER012W	14.07.11/**14.13**.01.01/ 32.01/43.01.03.09
		YGL004C	14.07.11/**14.13**
	2	YFR004W	14.07.11/**14.13**.01.01
	3	YDL097C	**14.13**.01.01/16.07
	4	YDL007W	01.04/14.07.11/ **14.13**.01.01/16.19.03
		YER021W	**14.13**.01.01
		YGL048C	01.04/10.03.01/14.07.11/ **14.13**.01.01/16.19.03
		YOR261C	**14.13**.01.01

Table 2: Three Examples of Protein Function Coherence in Partites of Maximal Bicliques/Tricliques/Quadricliques. The **boldface numbers** are the main functions shared by the majority proteins in partites of the corresponding *k*-partite cliques, and the **boldface partites** are our concerned ones whose percentage of proteins sharing main functions in the current partites is higher than the percentage of proteins sharing main functions in the partites' corresponding *k*-partite cliques. The *italic numbers* are the main functions shared by the majority proteins in the *k*-partite cliques. 99 indicates 'UNCLASSIFIED PROTEINS'.

k-partite cliques	Partites	Proteins	Functions
Biclique	**1**	YDR212W	**14.01**/16.01/***32.01***.07/42.04
		YJL111W	**14.01**/16.01/***32.01***.07/42.04
		YJL008C	**14.01**/16.01/***32.01***.07/42.04
		YDL143W	**14.01**/16.01/***32.01***.07/42.04
		YLR259C	**14.01**/14.04/16.03.01/ 20.01.10/20.09.04/***32.01***
	2	YDR030C	10.01.05.01
		YNL317W	11.04.03.01/11.04.03.05/16.03.03
		YGL190C	01.04/10.03.01.03/10.03.03/12 /14.07.03/*32.01*/40.01/42.04.03/ 42.29/43.01.03.05
		YBR198C	10.01.09.05/10.03.01.01.01/ 11.02.03.01.01/11.02.03.04/ 14.07.04/42.10.03
		YER173W	10.01.05.01/10.01.05.03.01/ 10.03.01.02/10.03.01.03/16.03.01/ 18.02.01/*32.01*.09
Triclique	**1**	YBR127C	20.01.01.01/20.01.15/20.03.22/ **20.09**.13/34.01.01.03
		YDL126C	01.04/*10.03*.01/14.13.01.01/ 16.19.03/20.01.10/**20.09**.07.27/ 40.10.02
		YFL039C	*10.03*.01/*10.03*.03/*10.03*.05.01/ *10.03*.05.03/11.02.03.04/14.04/ 14.07.04/16/20.01.10/**20.09**.07/ **20.09**.14.02/**20.09**.16.09.03/ **20.09**.18.09.01/32.01.03/40.01/ 42.01/42.04/42.10.03/42.29/ 43.01.03.05/43.01.03.09
		YML085C	*10.03*.01.01.11/*10.03*.04.05/ *10.03*.04.09/41.01.01/42.04/42.10
		YNL064C	*10.03*.01/14.01/14.04/18.02.01.01/ **20.09**.04/**20.09**.07/32.01
	2	YDL017W	01.04/10.01.03.03/10.01.11/ *10.03*.01/*10.03*.02/14.07.03
		YDL059C	10.01.05.01/10.01.05.03.03/ 16.01/32.01.09/42.10.03
		YDL200C	10.01.05.01/32.01.09/32.05.01.03
		YOL133W	*10.03*.01.01.03/*10.03*.01.01.09/ 11.02.03.04/14.07.05/14.10/ 14.13.01.01/16.01/16.19.03
	3	YGL137W	20.09.07.03
Quadriclique	**1**	YAL007C	14.04/**20.09**.07.03
		YBR159W	01.05/*01.06*
		YCR034W	01.05/*01.06*/**20.09**.13/34.11.03.07/ 40.01/43.01.03.05/43.01.03.09
		YLR372W	*01.06*.05/**20.09**.07.06/43.01.03.05
		YOR016C	14.04/**20.09**.16
	2	YPL076W	*01.06*.02.01/14.07.01
	3	YGR060W	*01.06*.06.11
	4	YCL025C	20.01.07
		YHR140W	99