Protein Function Prediction by Clustering of Protein-Protein Interaction Network

Chapter · September 2011

DOI: 10.1007/978-3-642-28664-3_4

Ivana Cingovska
École Polytechnique Fédérale de Lausanne
7 PUBLICATIONS **29** CITATIONS

Aleksandra Kovachev
Delivery Hero SE
14 PUBLICATIONS **77** CITATIONS

Kire Trivodaliev
Ss. Cyril and Methodius University in Skopje
45 PUBLICATIONS **1,180** CITATIONS

Slobodan Kalajdziski
Ss. Cyril and Methodius University in Skopje
78 PUBLICATIONS **280** CITATIONS

Protein Function Prediction by clustering of Protein-Protein Interaction Network

Ivana Cingovska, Aleksandra Bogojevska, Kire Trivodaliev, Slobodan Kalajdziski

Ss. Cyril and Methodius University, Faculty of Computer Science and Engineering,
Ruger Boskovic 16, 1000 Skopje, Macedonia
{ivana.chingovska, aleksandra.bogojevska, kiret, skalaj}@finki.ukim.mk

Abstract. The recent advent of high throughput methods has generated large amounts of protein-protein interaction network (PPIN) data. When studying the workings of a biological cell, it is useful to be able to detect known and predict still undiscovered protein complexes within the cell's PPINs. Such predictions may be used as an inexpensive tool to direct biological experiments. Because of its importance in the studies of protein interaction network, there are different models and algorithms in identifying functional modules in PPINs. In this paper, we present two representative methods, focusing on the comparison of their clustering properties in PPIN and their contribution towards function prediction. The work is done with PPIN data from the bakers' yeast (Saccaromyces cerevisiae) and since the network is noisy and still incomplete, we use pre-processing and purifying. As a conclusion new progress and future research directions are discussed.

Keywords: Protein interaction networks, Graph clustering, Protein function prediction.

1 Introduction

The rapid development of genomics and proteomics has generated an unprecedented amount of data for multiple model organisms. As has been commonly realized, the acquisition of data is a preliminary step, and a true challenge lies in developing effective means to analyze such data and endow them with physical or functional meaning [1].

Significant amount of data used for computational function prediction is produced by high-throughput techniques. Methods like Microarray co-expression analysis and Yeast2Hybrid experiments have allowed the construction of large interaction networks. A protein-protein interaction network (PPIN) consists of nodes representing proteins, and edges representing interactions between proteins. Such networks are stochastic as edges are weighted with the probability of interaction. There is more information in a PPIN compared to sequence or structure alone. A network provides a global view of the context of each gene/protein. Hence, the next stage of computational function prediction is characterized by the use of a protein's interaction context within the network to predict its functions. A node in a PPIN is annotated with one or more functional terms. Multiple and sometimes unrelated annotations can occur due to multiple active binding sites or possibly multiple stable tertiary conformations of a protein. The annotation terms are commonly based on ontology, like Gene Ontology (GO) project [2].

One of the main characteristics of the protein interaction networks is that they contain regions or subnetworks densely connected within, but very sparsely interconnected between themselves. This is the main reason for development of methods that perform clustering over the graph representing the protein interaction network. The modular structure of the biological networks in general is proven in [3], where the protein interaction network is clustered using three different approaches. The first one finds the completely connected subgraphs of the network and considers them as clusters. The second one exploits super paramagnetic clustering for data which are not in some metric space. The third approach observes the clustering as an optimization problem, thus maximizing the density of connectedness.

One very important system for clustering protein interaction networks is the MCODE system described in [4]. It performs the clustering in three stages: (1) weightening the nodes in the graph, using the clustering coefficient of the node's neighbourhood, (2) the weighted graph is traversed recursively and molecular complex is formed out of the nodes which have weight above a certain threshold, (3) post processing the results. NetworkBlast [5] is a tool in which every subgraph of the graph of protein interactions is considered as a candidate for functional module. Its modularity is then evaluated by calculating the ratio between the likelihood that it can be set to previously created model of protein complex and the probability that the edges in it are random. An algorithm which detects densely connected subgraphs with n nodes and need at least $n/2$ edges to be deleted in order to break its connectivity is presented in [6]. Markov clustering is used in the algorithm proposed in [7].

The algorithm proposed in [8] is a typical member of the family of algorithms which represent the proteins in some metric space. It calculates the adjacency between two proteins as their probability to have m common neighbours. Afterwards, hierarchical clustering is applied to the obtained distance matrix. Other representative example is the system PRODISTIN [9], which assumes that the Czekanovski-Dice distance between two proteins, which is based on the number of their common neighbours, mirrors their functional distance as well.

The quality of the obtained clusters can be evaluated in couple of ways. One of the criteria rates the clustering as good if the proteins in a cluster are densely connected between themselves, but sparsely connected with the proteins in the rest of the network [10]. Some systems provide tools for generation of graphs with known clusters, which is modelled with the parameters of the explored network [11]. Then the clusters obtained with the clustering algorithm are compared to the known ones. The clustering method can also be evaluated by its ability to reconstruct the experimentally and biologically confirmed protein complexes or functional modules [3][10][12]. For a system for protein function prediction, the most useful property of a clustering method is the functional homogeneity of the clusters.

In this paper we set up a framework for predicting protein function by using clustering in PPIN. We use two clustering methods, one that take into account the graph theory adjusted for PPIN, and other that transforms the PPIN into metric space. The PPIN data we use are from the bakers' yeast (Saccaromyces cerevisiae).

2 Research Methods

The methods for protein function prediction by clustering of PPIN generally consist of three phases, as represented on figure 1.

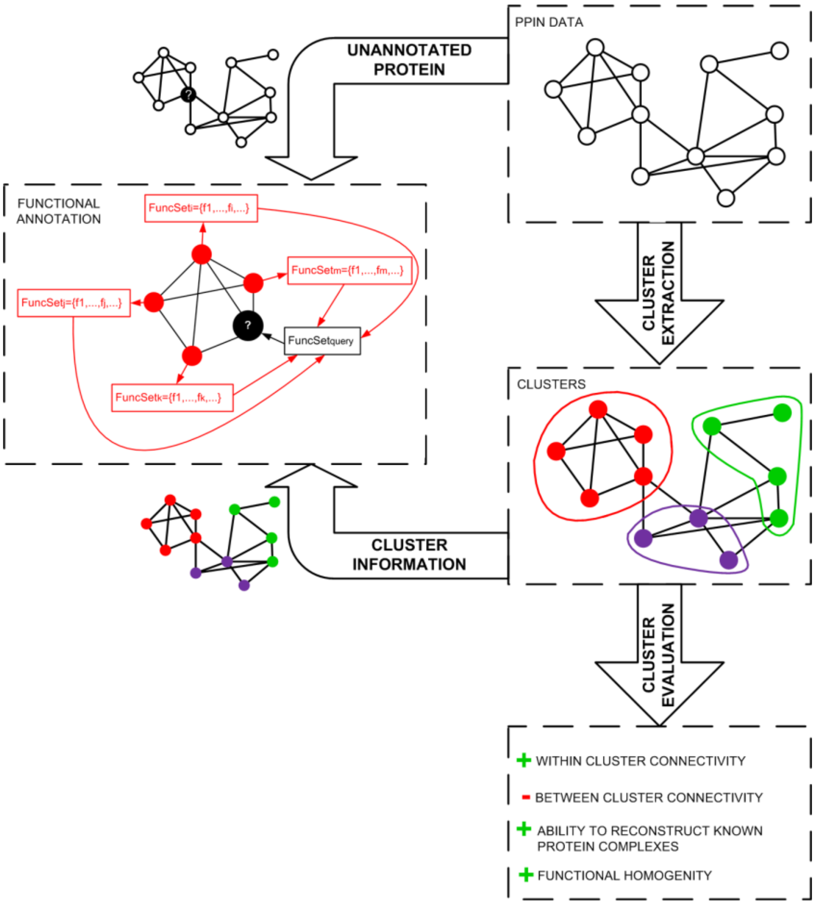

Fig. 1. General framework for protein function prediction by clustering in PPIN

The first phase is the dividing the network in clusters, using its topology or some other information for the nodes or the edges, if such an information is available. The compactness and the characteristics of the obtained clusters are then evaluated in the second phase. From physical aspect the clusters are assessed by the ratio of the number of edges within and between the clusters, and from biological aspect they are assessed by the functional and biological similarities of the proteins in the clusters. This second phase is not mandatory, but it is useful because it can point out what to expect from the function prediction itself. The prediction of the protein annotations for the proteins in the clusters is the task of the third phase.

2.1 Protein-Protein Interaction Data

High-throughput techniques are prone to detecting many false positive interactions, leading to a lot of noise and non-existing interactions in the databases. Furthermore, some of the databases are supplemented with interactions computationally derived with a method for protein interaction prediction, adding additional noise to the databases. Therefore, none of the available databases are perfectly reliable and the choice of a suitable database should be made very carefully.

For the needs of this paper the PPIN data are compiled, pre-processed and purified from a number of established datasets, like: DIP [13], MIPS [14], MINT [15], BIND [16] and BioGRID [17]. The functional annotations of the proteins were taken from the SGD database [18]. It is important to note that the annotations are unified with Gene Ontology (GO) terminology [2]. The GO consists of three structured dictionaries (ontologies): cellular component, biological process and molecular function. Due to the hierarchical structure of GO, the terms are linked between themselves with the relations: 'is_a', 'part_of', and 'regulates'.

The data for protein annotations are not used raw, but are preprocessed as proposed in [20]. First, the trivial functional annotations, like 'unknown cellular compartment', 'unknown molecular function' and 'unknown biological process' are erased. Then, additional annotations are calculated for each protein by the policy of transitive closure derived from the GO. The extremely frequent functional labels (appearing as annotations to more than 300 proteins) are also excluded, because they are very general and do not carry significant information.

After all the preprocessing steps, the used dataset is believed to be highly reliable and consists of 2502 proteins from the interaction of the baker's yeast, has 12708 interactions between them and are annotated with a total of 888 functional labels. For the purposes of evaluating the proposed methods, the largest connected component of this dataset is used, which consists of 2146 proteins.

2.2 Cluster Extraction

We use two different methods for cluster extraction from the PPIN data. The first one, edge-betweenness clustering, found its first use for clustering biological networks in [20], but the scope of this paper was to explore the dependence between the number of obtained clusters and the number of deleted edges for different datasets. The second method relies on spectral analysis of the PPIN. Similar algorithm is used in [21], but only for the purpose of predicting protein interactions rather than annotations.

Edge-Betweenness Clustering

The idea for clustering of networks using the concept of edge-betweenness was first proposed in [22] and is an extension of the concept of node-betweenness, which is an estimate of the centrality of a node in a network. Analogous to the definitions for node-betweenness, the betweenness of an edge is calculated as the number of shortest paths between any two nodes in the graph which pass through that edge. The edges which are between clusters, i.e. which connect two nodes of different clusters have higher betweenness then the edges which connect nodes that belong to the same cluster. By deleting the edges with the highest betweenness, after certain number of iterations the graph will be separated into several components which can be treated as clusters.

The main changing parameter of this algorithm is the number of edges that need to be deleted. In our research, this number in obtained empirically. The betweenness of each edge is recalculated after each iteration, which, regarding to [22] is better strategy then just calculating the edge betweenness of every edge only once at the beginning and then deleting the edges with highest betweenness.

If the number of nodes in the graphs is $|V|$ and the total number of edges between them is $|E|$, then the complexity of the algorithm is $O(|E|^2|V|)$.

Clustering Based on Spectral Analysis of the Protein-Protein Interaction Graph

One of the basic types of graph clustering, according to [23], is the spectral clustering, which performs spectral analysis of the graph's adjacency matrix or some of its derivatives, by finding its eigenvalues and eigenvectors. The first step in the spectral clustering is transforming the initial dataset into a set of points in an n-dimensional space, whose coordinates are elements of n selected eigenvectors. This change in the representation of the data enhances the characteristics of the clusters making them more distinctive. Then a classical clustering algorithm, like k-means for example, can be used, to cluster the data.

Although the initial idea for spectral analysis was intended directly to the adjacency matrix of the graph, the newer algorithms use the Laplacian matrix L, which is derived from the adjacency matrix A as in equation (1).

$$L = D - A\ . \tag{1}$$

In this equation, D is a diagonal matrix whose diagonal element D_{ii} equals the degree of the node i of the graph. Before spectral analysis, the Laplacian matrix needs to be normalized.

The main characteristic of the graphs' Laplacian matrix is the fact that the number k of zero eigenvalues equals to the number of connected components of the graph. The non-zero values of the corresponding eigenvectors are on the indices of the nodes that belong to the corresponding connected component. If those eigenvectors are put as columns of one $|V|$xk matrix, each row represents one node which has only one non-zero value: on the position of the eigenvector of the connected component it belongs to.

If the graph consists of only one connected component, that the Laplacian will have only one non-zero eigenvalue. Let the number of clusters that the graphs should be separated into be k. Taking the k eigenvectors that correspond to the k eigenvalues closest to 0, and transforming the nodes of the graph into the k-dimensional space that they form, all the nodes that belong to one cluster will be situated close in that space. This way the nodes will be brought into a form suitable for using any clustering algorithms, like k-means. The number of clusters is determined empirically.

2.3 Cluster Evaluation

One of the evaluation models which give a general overview of the qualitative differences between the clustering algorithms is proposed in [10] and it provides information whether a cluster has the character of a module or densely connected subgraph. This is highly important, because, according to [3], the term functional module is closely related to subgraphs rich in edges within it. The first necessary criteria for a cluster to be considered as a module is given with (2), where *n* is the number of nodes, k_{in} is the number of edges within the cluster and k_{out} is the number of edges from the cluster to nodes which don't belong to it.

$$\sum_{i=1}^{n} k_{in}^{i} > \sum_{i=1}^{n} k_{out}^{i} . \tag{2}$$

The second criterion requires that collectively, the number of neighbors of each node within the cluster is higher than the number of neighbors from the module to the outside. This criterion is given by (3).

$$\{k_{in}^{1}, k_{in}^{2}, ..., k_{in}^{n}\} >> \{k_{out}^{1}, k_{out}^{2}, ..., k_{out}^{n}\} \tag{3}$$

Whether the module meets these criteria or not is determined by using the Wilcoxon non-parametric statistical test for comparing the distribution of two random variables.

2.4 Functional Annotation Using Clusters

After clustering the PPIN we set up a strategy for annotating the query protein with the adequate functions according to the functions of the other proteins in the cluster where it belongs. The simplest and most intuitive approach would be that each function is ranked by its frequency of appearance as an annotation for the proteins in the cluster. This rank is calculated by the formula (4) and is then normalized in the range from 0 to 1.

$$f(j)_{j \in F} = \sum_{i \in K} z_{ij} \tag{4}$$

where F is the set of functions present in the cluster K, and

$$z_{ij} = \begin{cases} 1, \text{if i - th protein from K is annotated with the j - th function from F} \\ 0, \text{otherwise} \end{cases} \tag{5}$$

3 Results and Discussion

Each protein in the PPIN is streamed through the prediction process one at a time as a query protein. The query protein is considered unannotated, that is we employ the leave-pone out method. Each of the algorithms works in a fashion that ranks the "proximity" of the possible functions to the query protein. The ranks are scaled between 0 and 1 as explained in 2.4. The query protein is annotated with all functions that have rank above a previously determined threshold ω. For example, for ω = 0, the query protein is assigned with all the function present in its cluster. We change the threshold with step 0.1 and compute numbers of true-positives (TP), true- negatives (TN), false-positives (FP) and false-negatives (FN). For a single we consider the TP to be the number of correctly predicted functions, and for the whole PPIN and a given value of ω the TP number is the total sum of all single protein TPs .

To compare performance between different algorithms we use standard measures as sensitivity and specificity (6).

$$sensitivity = \frac{TP}{TP + FN} \qquad specificity = \frac{TN}{TN + FP} \tag{6}$$

We plot the values we compute for the sensitivity and specificity using a ROC curve (Receiver Operating Curve). The *x*-axe corresponds to the false positive rate, which is the number of false predictions that a wrong function is assigned to a single protein, scaled by the total number of functions that do not belong to that particular protein. This rate is calculated with (7).

$$fpr = \frac{FP}{FP + TN} = 1 - specificity \tag{7}$$

The *y*-axe corresponds to the rate of true predictions that is the sensitivity. At last we use the AUC (Area Under the ROC Curve) measure as a numeric evaluator of the ROC curve. The AUC is a number that is equal to the area under the curve and its value should be above 0.5, which is the value that we get if the prediction process was random. The closer the value of AUC to 1, the better is the prediction method.

Before we evaluate the prediction performance of the proposed methods, first we assess their clustering properties on the PPIN. For each of the methods we use a changing parameter as explained in 2.2. For the edge-betweenness method we performed experiments using deletion of 1000 and 1400 edges. For the spectral clustering we experimented with different numbers of eigenvalues starting from 50 up to 300, with a changing step of 50. The results are presented in Table 1.

Table 1. Evaluation results of the clustering methods using method described in 2.3

	changing parameter	**number of clusters**	**clusters meeting module criteria (%)**
edge-betweenness	1000	103	85.44
	1400	217	54.84
spectral clustering	50	50	100.00
	100	100	94.00
	150	150	84.67
	200	200	65.50
	250	250	47.60
	300	300	35.67

As can be concluded from Table 1, the number of clusters which have the nature of a module reduces as the cluster size decreases i.e. as the total number of clusters increases. However, for certain parameters for the both algorithms (1000 removed edges with the edge-betweenness method and 50, 150 and 200 eigenvalues for the spectral clustering method), the percentage of modules among the obtained clusters is sufficiently high. Thus, it is reasonable to presume that the clustering process has produced functional modules.

After evaluating the clustering properties we move towards the evaluation of the function prediction when using the two clustering methods.

Table 2. Function prediction evaluation when using edge-betweenness method

No. of deleted edges	ω =	0,1	0,3	0,5	0,7	0,9	AUC
1000	**sens.**	0,6693	0,4753	0,3266	0,2459	0,1445	0,8610
	fpr	0,0456	0,0136	0,0051	0,0027	0,0011	
1400	**sens.**	0,6651	0,5131	0,3741	0,2872	0,1623	0,8430
	fpr	0,0355	0,0118	0,0046	0,0025	0,0012	

Table 3. Function prediction evaluation when using spectral clustering method

No. of eigenvalues	ω =	0,1	0,3	0,5	0,7	0,9	AUC
50	**sens.**	0,6484	0,4436	0,3246	0,2082	0,1147	0,8644
	fpr	0,0565	0,0159	0,0077	0,0036	0,0014	
100	**sens.**	0,6702	0,4713	0,3376	0,2430	0,1404	0,8590
	fpr	0,0479	0,0142	0,0056	0,0028	0,0010	
150	**sens.**	0,6709	0,5053	0,3620	0,2688	0,1598	0,8531
	fpr	0,0400	0,0128	0,0050	0,0026	0,0012	
200	**sens.**	0,6783	0,5295	0,3859	0,3053	0,1870	0,8458
	fpr	0,0383	0,0116	0,0049	0,0027	0,0013	
250	**sens.**	0,6690	0,5434	0,3987	0,3175	0,2039	0,8381
	fpr	0,0329	0,0121	0,0047	0,0028	0,0015	
300	**sens.**	0,6538	0,5375	0,3952	0,3196	0,2143	0,8283
	fpr	0,0307	0,0121	0,0046	0,0028	0,0017	

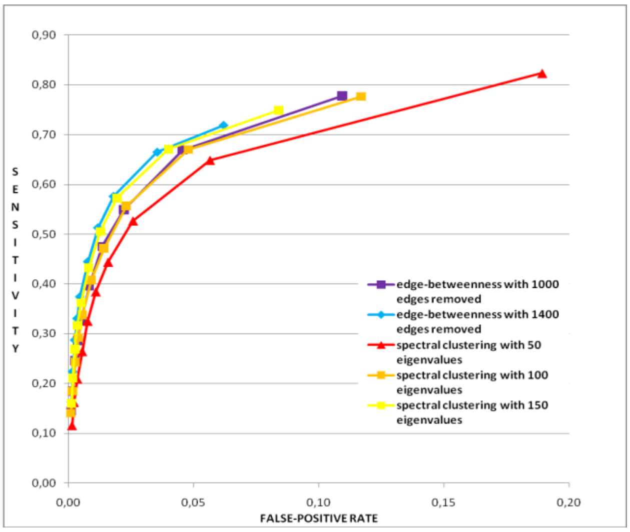

Fig. 2. ROC curves for the function prediction evaluation for the edge-betweenness and spectral clustering method

By comparing the AUC values of the results in Table 2 and Table 3, it can be concluded that the edge-betweenness method performs better when 1000 edges are removed, while the spectral clustering renders best results when only 50 eigenvalues are considered. The spectral clustering method is slightly superior over the edge-betweenness method according to the AUC values, but the edge-betweenness method achieves better sensitivity and false positive rate for $\omega = 0.1$. It is important to notice that for $\omega = 0$ the algorithms achieve very high sensitivity of over 77.78% for the edge-betweenness method and 82.35% for the spectral clustering method in best case. However, the profitability of this result for the spectral clustering method is questionable, because the false positive rate in that case is nearly 20%. Therefore, it would be useful to inquire what the permissible trade-off limit between correctly and incorrectly detected protein functions is. Graphical visualization of the function prediction results is given in Fig. 2.

4 Conclusion and Future Directions

This paper exploits the ability of two graph clustering methods for detecting functional modules and predicting protein functions from PPIN. The methods were tested over one of the richest interactomes: the interactome of the baker's yeast. The first approach uses the edge-betweenness algorithm for graph clustering, while the second one performs spectral clustering over the Laplacian of the adjacency matrix of the PPIN. Due to the fact that the PPIN data contain a lot of false positive interactions, the dataset needs to be preprocessed and purified prior to the functional annotation. This paper also illustrates a general framework for the vast set of algorithms for protein function prediction which are based on clustering of the PPIN. The proposed approaches prove that utilizing clustering of the PPIN has high potential in the task of protein function prediction. The results show that both algorithms achieve high sensitivity and small false positive rate and they both have high AUC values, with some advantage of the edge-betweenness method which has smaller false positive rates. However there is one limitation of our current approach, that is, all of our analyses were performed on unweighted graphs, because our reference PPIN does not contain any information that would enable us to assign reliability values (weights) to the edges. It should be mentioned that if a method can deal with weighted graphs it would be likely to give better performances if the weights reflect the reliability of the links between proteins. Since spectral clustering can deal with weighted graphs, while the edge-betweenness clustering does not take in account any edge weight, future directions for using clustering for the aim of function prediction should follow the spectral clustering approach.

References

1. Yu, G. X., Glass, E. M., Karonis, N. T., Maltsev, N.: Knowledge-based voting algorithm for automated protein functional annotation. PROTEINS: Structure, Function, and Bioinformatics 61, 907-917 (2005)

2. The gene ontology consortium: Gene ontology: Tool for the unification of biology. Nature Genetics 25(1), 25-29 (2000)
3. V. Spirin, L. A. Mirny: Protein complexes and functional modules in molecular networks. PNAS, Vol.100, No.21 (2003)
4. G. D. Bader, C. WV. Hogue: An automated method for finding molecular complexes in large protein interaction networks. BMC Bioinformatics 2003, 4:2 (2003)
5. R. Sharan, T. Ideker, B.Kelley, R. Shamir, RM. Karp: Identification of protein complexes by comparative analysis of yeast and bacterial protein interaction data. Computational Biology 12: 835-846 (2005)
6. N. Przulj, D. A. Wigle, I. Jurisica: Functional Topology in a Network of Protein Interactions. Bioinformatics, Vol. 20, No. 3, 340--348 (2004)
7. N. J. Krogan, G. Cagney, H. Yu, G. Zhong, X. Guo, A. Ignatchenko: Global Landscape of Protein Complexes in the Yeast Saccaromyces cerevisiae. Nature, Vol. 440, 637-643 (2006)
8. M. P. Samanta, S. Liang: Predicting protein functions from redundancies in large-scale protein interaction networks. PNAS, Vol. 100, No. 22 (2003)
9. C. Brun, F. Chevenet, D. Martin, J. Wojcik, A. Guénoche, B. Jacq: Functional classification of proteins for the prediction of cellular function from a protein-protein interaction network. Genome Biology 5:R6, (2003)
10. J. Chen, B. Yuan: Detecting Functional Modules in the Yeast Protein-Protein Interaction Network. Bioinformatics, Vol. 22, No. 18, 2283-2290 (2006)
11. A. Lancichinetti, S. Fortunato, F. Radicchi: Benchmark Graphs for testing Community Detection Algorithms. Physical Review E78, 046110, (2008)
12. A. W. Rives, T. Galitski: Modular Organization of cellular Networks. PNAS Vol. 10, No. 3, 1128-1133 (2003)
13. L. Salwinski, C. S. Miller, A. J. Smith,F. K. Pettit, J. U. Bowie, D. Eisenberg: The Database of Interacting Protein. Nucleic Acids Res. (2004)
14. U. Guldener, M. Munsterkotter, M. Oesterheld, P. Ragel, A. Ruepp, and H. W. Mewes: MPact: the MIPS protein interaction resource on yeast. Nucleic Acids Res. Vol. 34, (2006)
15. A. Chatr-aryamontri, A. Ceol, L. Montecchi Palazzi, G. Nardelli, M. V. Schneider, L. Castagnoli, G. Cesareni: MINT: the Molecular INTeraction database. Nucleic Acids Res. 35, D572--D574 (2007)
16. G. D. Bader, C. W. V. Hogue: BIND–a data spec. for storing and describing biomolecular interactions, molecular complexes and pathways. Bioinformatics, Vol.16(5), 465-477 (2000)
17. B. J. Breitkreutz, C. Stark, M. Tyers; The GRID: The General Repository for Interaction Datasets. Genome Biology, (2003)
18. S. Dwight, M. Harris, K. Dolinski, C. Ball, G. BUnkley. K. Christie, D. Fisk, L. Issel-Tarver, M. Schroeder, G. Sherlock, A. Sethuraman, S. Weng, D. Botstein, J.M. Cherry: Saccharomyces Genome Database (SGD) provides secondary gene annotation using Gene Ontology (GO). Nucleic Acids Research 30(1), (2002)
19. Letovsky, S., Kasif, S.: Predicting protein function from protein/protein interaction data: a probabilistic approach. Bioinformatics 19, i197--i204 (2003)
20. R. Dunn, F. Dudbridge, C. M. Sanderson: The Use of Edge-Betweenness Clustering to Investigate Biological Function in PIN. BMC Bioinformatics 6:39, (2005)
21. Z. Sen, A. Kloczkowski, R. L. Jernigan: Functional Clustering of Yeast Proteins from the Protein-Protein Interaction Network. BMC Bioinformatics 7:355, (2006)
22. M. Girvan, M. E. J. Newman: Community Structure in Social and Biological Networks. PNAS 99(12), 7821--7826 (2002)
23. S. Fortunato: Community Detection in Graphs. Physics Reports 486, 75--174 (2010)