# Determination of Protein Functional Groups Using the Bond Energy Algorithm

Cvetanka Atanasova, Kire Trivodaliev, Slobodan Kalajdziski

Ss. Cyril and Methodius University, Faculty of Computer Science and Engineering, Ruger Boskovic 16, 1000, Skopje, Macedonia
cvetanka_atanasova@yahoo.com, {kire.trivodaliev, slobodan.kalajdziski}@finki.ukim.mk

**Abstract.** Nowadays it is possible to understand the basic components and organization of cell machinery from the network level due to increased availability of large-scale protein-protein interaction (PPI) data. Many studies have shown that clustering of the protein interaction network (PIN) can be found as an effective approach for identifying protein complexes or functional modules. A significant number of proteins in such PIN remain uncharacterized and predicting their function remains a major challenge in system biology. We propose a protein annotation method based on clustering according to Bond Energy Algorithm (BEA), which first transforms the PIN into matrix form suitable for BEA, and after generating the resulting matrix of the BEA the AUTOCLASS algorithm is performed to obtain the PIN clusters. Protein functions are assigned based on cluster information. Experiments were performed on PPI data from the bakers' yeast and since the network is noisy and still incomplete, we use pre-processing and purifying. Results reveal improvement over previous techniques and the most prominent characteristic of the BEA clustering is that the clustering result is not dependent of the initial number of clusters.

**Keywords:** Bond Energy Algorithm (BEA), protein interactions network, clustering methods, protein function prediction.

## 1 Introduction

Proteins seldom act as single isolated units to perform their functions within the cells. It has been observed that proteins involved in the same cellular processes often interact with each other [1]. Protein-protein interactions are thus fundamental to almost all-biological processes [2]. With the advancement of the high-throughput technologies, such as yeast-two-hybrid, mass spectrometry, and protein chip technologies, huge data sets of protein-protein interactions become available [3]. Such protein-protein interaction data can be naturally represented in the form of networks, which not only give us the initial global picture of protein interactions on a genomic scale but also help us to understand the basic components and organization of cell machinery from the network level.

S. Markovski, M. Gusev (Editors): ICT Innovations 2012, Web Proceedings, ISSN 1857-7288

An important challenge for system biology is to understand the relationship between the organization of a network and its function. It has been shown that clustering protein interaction networks is an effective approach to achieve this goal [4]. Clustering in protein interaction networks is to group the proteins into sets (clusters), which demonstrate greater similarity among proteins in the same cluster than in different clusters. Since biological functions can be carried out by particular groups of genes and proteins, dividing networks into naturally grouped parts (clusters or communities) is an essential way to investigate some relationships between the function and topology of networks or to reveal hidden knowledge behind them.

Classical graph-based agglomerative methods employ a variety of similarity measures between nodes to partition PPI networks, but they often result in a poor clustering arrangement that contains one or a few giant core clusters with many tiny ones [5]. To improve the clustering results, PPI networks were weighted based on topological properties such as shortest path length [6], [7], clustering coefficients [8], node degree, or the degree of experimental validity [9]. As a new type of clustering algorithms, the edge-betweenness was defined as a global measure to separate PPI networks into subgraphs in a divisive manner [10], [11], [12]. Edge-betweenness is the number of shortest paths between all pairs of nodes that run through the edge. It is able to identify biologically significant modular structures, but it requires lots of computational resources. As an approach to coordination of typical clustering algorithms, an ensemble method was proposed to combine multiple, independent clustering arrangements to deduce a single consensus cluster structure [13].

Heuristic rule-based algorithms were proposed to reveal the structure of PPI networks [14], [15]. A layered clustering algorithm was presented, which groups proteins by the similarity of their direct neighborhoods to identify locally significant proteins that links different clusters, called mediators [16]. Power graph analysis transforms biological networks into a compact, less redundant representation, exploiting the abundance of cliques and bicliques as elementary topological motifs [17]. Spectral clustering analysis, which is an appealing simple and theoretically sound method [17], [18], has hardly been studied to partition PPI networks, while it is used for detecting protein complexes [19].

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 [20]. Some systems provide tools for generation of graphs with known clusters, which is modelled with the parameters of the explored network [21]. 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 [12], [19], [20].

In this paper we set up a framework for predicting protein functional groups by using clustering in PIN. We use Bond Energy Algorithm, which first transforms the PIN into matrix form suitable for BEA, and after generating the resulting matrix of the BEA the AUTOCLASS algorithm is performed to obtain the PIN clusters. Protein

functions are assigned based on cluster information. Experiments were performed on PPI data from the bakers' yeast and since the network is noisy and still incomplete, we use pre-processing and purifying. We also performed network weighting based on the annotation correlation between nodes.

## 2 Research Methods

The methods for protein function prediction by clustering of PIN generally consist of three phases. The first phase is dividing the network in clusters, using its topology or some other information for the nodes or the edges, if such information is available. In this paper we use the Bond Energy Algorithm in combination with the AUTOCLASS algorithm in order to obtain the clusters of the PIN. The compactness and the characteristics of the obtained clusters are then evaluated in the second phase. From physical aspect the clusters can be assessed by the ratio of the number of edges within and between the clusters, and from biological aspect they can be assessed by the functional and biological similarities of the proteins in the clusters. This second phase is not mandatory, but might be 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 Preprocessing and Transforming the PIN Data

The protein interactions network represents an annotated graph in which the nodes are the proteins themselves, whereas the links between the proteins in the graph are the interactions among the proteins. The graph can be seen as an annotated graph since every node is assigned with one or more terms corresponding to the functions that are performed by the protein represented by that node. Because there is a symmetry in the nodes (proteins) of the graph, this means that if the protein $P_1$ is in an interaction with the protein $P_2$, than the protein $P_2$ will also be in an interaction with the protein $P_1$. If we mark the multitude of nodes with $J$ and the multitude of connections among the nodes with $V$, than the notation of the graph $G$ would be the following: $G = (J, V)$.

Of great importance for the research is the neighborhood matrix by which the graph $M = (a_{ij})_{i, j \in J}$ is represented. For the neighborhood matrix applies that $a_{ij} = 1$ if the protein $i$ is in an interaction with the protein $j$, namely $a_{ij} = 0$ if the protein $i$ is not in an interaction with the protein $j$. This applies for a non-weighted graph. There are algorithms according to which such non-weighted graphs are prescribed appropriate measures which represent the probabilities of a protein's interaction with the other protein and in such a case the graph converts into a weighted graph for which the formula $a_{ij} = w_{ij}$ applies.

If we mark the multitude of all the protein's functions with $F$, then, besides the neighborhood matrix which represents the interactions network, the annotations matrix $Z = (z_{ij})_{i \in J, j = F}$ is also important where the rows are marked with $| J |$ and the

columns are marked with | $F$ |. In addition, $z_{ij} = 1$ if the protein $i$ is annotated with the function $j$ and $z_{ij} = 0$ if the protein $i$ is not annotated with the function $j$.

After obtaining the matrix representation $M$ of the graph, we can apply the Bond Energy Algorithm in order to reorder the matrix according to the affinity between the proteins from the data set. The matrix $F$ is used during the phase of determining the functions of the proteins after the clustering is performed.

### 2.2 Clustering by Using the Bond Energy Algorithm

Bond Energy Algorithm (BEA) has been widely used for vertical fragmentation in distributed databases. The algorithm was originally proposed by McCormick and Hoffer and Severande [22]. BEA creates clusters by using a non-trivial similarity metric (attribute affinity measures) defined on the elements of the data set. BEA is comprised by two algorithms, the first one is used for ordering the data set to locate the most related elements close together (and to separate the unrelated elements) and the second one is used for creating the groups to determine the best cut of the ordered data set (i.e. create a cluster).

```
input: Attribute Affinity Matrix
output: CA: Clustered Affinity matrix and order list array.
begin
  [initialize; the AA matrix is created]
  CA(*, 1) ← AA(*, 1)
  CA(*, 2) ← AA(*, 2)
  index ← 3
  while index ≤ n do [choose the "best" location for profile AA_index]
  begin
    for i from 1 to index - 1 by 1 do
      calculate cont(AA_i-1, AA_index, AA_i)
    end-for
    calculate cont(AA_index-1, AA_index, AA_index+1) [boundary condition]
    loc ← placement given by maximum cont value
    for j from index to loc by -1 do [shuffle the two matrices]
      calculate CA(*, j) ← CA(*, j - 1)
    end-for
    CA(*, loc) ← AA(*, index)
    index ← index + 1
  end-while
  order the rows according to the relative ordering of the columns
end.
Note: * means for each element in the data set
```

**Fig. 1.** Pseudo-code of the Bond Energy Algorithm

The fundamental task in the first part of the algorithm is to find some means of grouping the attributes in a data set based on the attribute affinity values in AA (Attribute Affinity Matrix). Bond Energy Algorithm takes as input the attribute affinity matrix, permutes its rows and columns, and generates a clustered affinity matrix (CA). Once this measure is maximized the permutations are considered to be done. The pseudo-code of the BEA is presented on the Figure 1.

After obtaining the permutation of the initial matrix, we can apply the AUTOCLASS method [23], which is an unsupervised clustering method that seeks for a maximally good clustering. The structure of this classifying method is shown on Figure 2. AUTOCLASS represents a computer implementation of the Bayesian unsupervised data classification technique. By assigning real or discreet values, AUTOCLASS determines the probable number of classes in which the data and probabilities with which a given object belongs to the assigned class are distributed. As can be seen from the Figure 2, the AUTOCLASS algorithm iterates the number of clusters until it converges. For each element of the dataset, it computes the probabilities for belonging to some of the classes, and in each iteration it reestimates these values according to the current cluster distribution. One of the best features of this method is that the number of classes is automatically determined and there is no need of any previous setting of the number of resulting clusters.

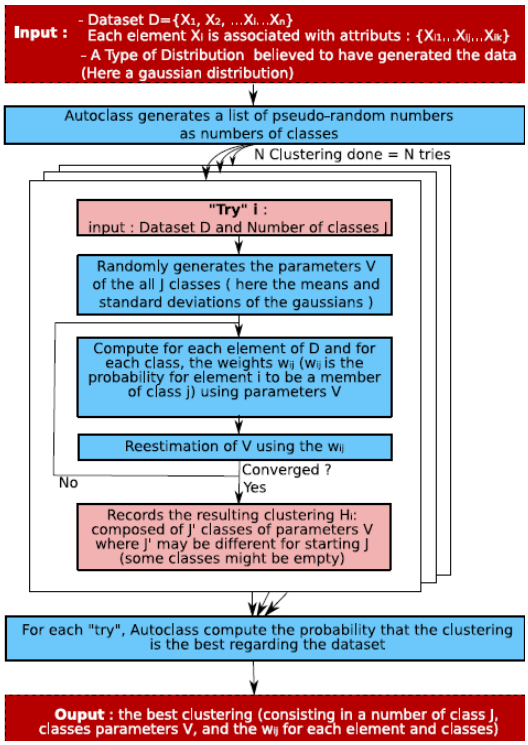


**Fig. 2.** AUTOCLASS Structure

After applying the AUTOCLASS method, the resulting clusters can be further evaluated in order to obtain the efficiency parameters of the clustering method.

### 2.3 Functional Annotation Using Clusters

As we previously mentioned, the whole process consists of several phases. As an underlying data set is a non-weighted protein interactions network.

The first phase includes preprocessing of the data in order to get them in a format which is compatible with the BEA Algorithm. The second phase includes execution of the BEA Algorithm, which provides an ordered matrix which is an input to the AUTOCLASS method. This phase provides the data set clusters and performs an evaluation of the features and compatibility of the clusters found, from a biological aspect, namely, how similar the proteins are within the cluster according to their functional and biological features. Thus, a global functional mark is received for the cluster. This phase is very useful because it indicates what can be expected for the results from the prediction. This phase also calculates the enrichment ratio of the cluster by a certain function, a specification which directly indicates the possible function of the interrogative protein. The prediction of the functions of the non-annotated protein is elaborated in the third phase. Every function receives an appropriate rating depending on the frequency of appearances within the cluster $K$ which contains the interrogative protein. The ratings of every function are received by the formula (1) and are then normalized within range from 0 to 1. All the proteins from the protein interaction network go through the process as non-annotated proteins so that they can get their functions according to the methods elaborated in this research.

$$f(j)_{j\in F} = \sum_{i\in K} z_{ij} \tag{1}$$

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

$$z_{ij} = \begin{cases} 1, \text{if protein i from K has function j from F} \\ 0, \text{otherwise} \end{cases} \tag{2}$$

The last phase includes evaluation of the results.

## 3 Results and Discussion

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 PIN data are compiled, pre-processed and purified from a number of established datasets, like: DIP, MIPS, MINT, BIND and BioGRID.

The functional annotations of the proteins were taken from the SGD database [24]. It is important to note that the annotations are unified with Gene Ontology (GO) terminology [25].

The data for protein annotations are not used raw, but are preprocessed and purified. First, the trivial functional GO 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.

The multitude used for this experiment is highly confidential and is consisted of 2502 proteins, among witch 12708 interactions are noted, with a total of 888 annotated functional marks. The protein interactions network does not represent a linked graph, but is consisted of several components, the biggest of which contains 2146 proteins. This multitude of interactions of the yeast is used for testing and evaluation of the methods for prediction of protein function subjected in this paper. Each protein in the PIN 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-one 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. The query protein is annotated with all functions that have rank above a previously determined threshold $\omega$. For example, for $\omega = 0$, 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 query protein we consider the TP to be the number of correctly predicted functions, and for the whole PIN and a given value of $\omega$ 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 (3).

$$sensitivity = \frac{TP}{TP + FN} \qquad specificity = \frac{TN}{TN + FP} \tag{3}$$

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 (4).

$$fpr = \frac{FP}{FP + TN} = 1 - specificity \tag{4}$$

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. The experimental results are shown on the Table 1, and on the Figure 3.

**Table 1.** Experimental Results. The values of the sensitivity and specificity are obtained for different values of the parameter $\omega$.

| $\omega$ = | **0** | **0,1** | **0,2** | **0,3** | **0,4** | **0,5** | **0,6** | **0,7** | **0,8** | **0,9** |
|---|---|---|---|---|---|---|---|---|---|---|
| sensitivity | 0,82 | 0,82 | 0,77 | 0,71 | 0,64 | 0,57 | 0,49 | 0,42 | 0,34 | 0,26 |
| specificity | 0,35 | 0,35 | 0,20 | 0,11 | 0,07 | 0,04 | 0,03 | 0,02 | 0,01 | 0,01 |
| AUC | | | | | **0.8** | | | | | |

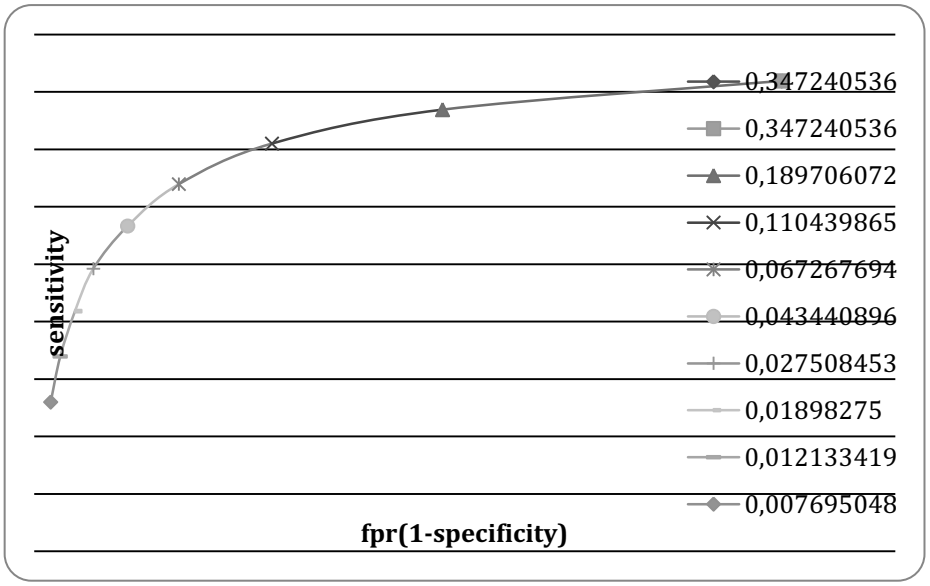


**Fig. 3.** The ROC curve for the obtained experimental results shown on table 1.

From the presented results, it can be noted that the sensitivity is highest in the first column reaching above 82%, which happens when the functions are falsely predicted with 35%. By increasing of $\omega$ for 0,2% it can be noted that the sensitivity decreases to 77% which, in fact, is not that great a margin, but the number of falsely predicted functions is decreased by 15%. Figure 3 shows the visual results of this experiment. It can be also seen from the Figure 3 that sensitivity values are 20% only when 1% of the functions are wrongly predicted. The value of AUC on this classificatory is 0,80.

## 4 Conclusion and Future Work

This paper exploits the possibilities for applying the Bond Energy Algorithm for clustering and detecting the functional modules and predicting protein functions from PIN. The method was tested over one of the richest interactomes: the interactome of

the baker's yeast. Database of Interacting Proteins (DIP) which was used, aimed to integrate the various experimental results from the biochemical analyses of the protein interactions. The protein interactions network is a non-weighted one, which means that we have information whether one protein is in an interaction with another one, and, as such, was used for prediction of the protein functions. We have used the matrix representation of the PIN and applied the BEA and AUTOCLASS in order to obtain the matrix permutation according to the affinity between the proteins. Due to the fact that the PIN data contain a lot of false positive interactions, the dataset needed to be preprocessed and purified prior to the functional annotation. The results show that our algorithm achieves high sensitivity and small false positive rate on PIN graphs and it has a high AUC value.

Our future work will be concentrated on the possibilities for adding weights to non-weighted graphs as well as to make a prediction of the protein functions on a weighted graph. It is also useful to add the additional settings within the Bond Energy Algorithm so that more data can be separated, coming from the weighted protein interactions network. The AUTOCLASS method which determines the clusters can also be modified i.e. improved by trying some other heuristics during its iteration process.

## References

1. von Mering, C., Krause, R., Sne, B.: Comparative assessment of large-scale data sets of protein-protein interactions. Nature 417, (2002) 68-87
2. Hakes, L., Lovell, S. C., Oliver, S. G.: Specificity in protein interactions and its relationship with sequence diversity and coevolution. PNAS 104, (2007) 19
3. Harwell, L. H., Hopfield, J. J., Leibler, S., Murray, A. W.: From molecular to modular cell biology. Nature 402, (1999) c47- c52
4. Brohée, S., van Helden, J.: Evaluation of clustering algorithms for protein-protein interaction networks. BMC Bioinformatics 7, (2006) 48
5. Barabasi, L., Oltvai, Z. N.: Network biology: understanding the cell's functional organization. Nat Rev Genet 5, (2004) 101-113
6. Arnau, V., Mars, S., Marin, I.: Iterative cluster analysis of protein interaction data. Bioinformatics 21, (2005) 364- 378
7. Rives, W., Galitski, T.: Modular organization of cellular networks. PNAS 100, (2003) 1128-1133
8. Friedel, C., Zimmer, R.: Inferring topology from clustering coefficients in protein-protein interaction networks. BMC Bioinformatics 7, (2006) 519
9. Pereira-Leal, J. B., Enright, A. J., Ouzounis, C. A.: Detection of functional modules from protein interaction networks. Proteins 54, (2004) 49-57
10. Dunn, R., Dudbridge, F., Sanderson, C. M.: The use of edge-betweenness clustering to investigate biological function in PINs. BMC Bioinformatics 6, (2005) 39
11. Luo, F., Yang, Y., Chen, C. F., Chang, R., Zhou, J.: Modular organization of protein interaction networks. Bioinformatics 23, (2007) 207–214
12. Newman, M. E., Girvan, M.: Finding and evaluating community structure in networks. Phys Rev E Stat Nonlin Soft Matter Phys 69, (2004) 026113
13. Asur, S., Ucar, D., Parthasarathy, S.: An ensemble framework for clustering protein-protein interaction networks. Bioinformatics 23, (2007) i29–40

14. Gagneur, J., Krause, R., Bouwmeester, T., Casari, G.: Modular decomposition of protein-protein interaction networks. Genome Biol 5, (2004) R57
15. Morrison, J. L., Breitling, R., Higham, D. J., Gilbert, D. R.: A lock-and-key model for protein-protein interactions. Bioinformatics 22, (2006) 2012-2019
16. Andreopoulos, B., An, A., Wang, X., Faloutsos, M., Schroeder, M.: Clustering by common friends finds locally significant proteins mediating modules. Bioinformatics 23, (2007) 1124-1131
17. Royer, L., Reimann, M., Andreopoulos, B., Schroeder, M.: Unraveling protein networks with power graph analysis. PLoS Comput Biol 4, (2008) e1000108
18. Spirin, V., Mirny, L. A.: Protein complexes and functional modules in molecular networks. PNAS 100, (2003) 21
19. Belkin, M., Niyogi, P.: Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. Neural Computation 15, (2003) 1373–1396
20. Chen, J., Yuan, B.: Detecting Functional Modules in the Yeast Protein-Protein Interaction Network. Bioinformatics 18, (2006) 22
21. Lancichinetti, A., Fortunato, S., Radicchi, F.: Benchmark Graphs for testing Community Detection Algorithms. Physical Review E78, (2008) 046110
22. Arabie, P., Hubert, L.: The Bond Energy Algorithm Revisited. IEEE Transaction on Systems, Man and Cybernetics, (1990)
23. Achcar, F., Camadro, J. M., Mestivier, D.: AutoClass@IJM: a powerful tool for Bayesian classification of heterogeneous data in biology. Nucleic Acids Research 37, (2009)
24. Dwight, S., Harris, M., Dolinski, K., Ball, C., Unkley, G. B., Christie, K., Fisk, D., Issel-Tarver, L., Schroeder, M., Sherlock, G., Sethuraman, A., Weng, S., Botstein, D., Cherry, J. M.: Saccharomyces Genome Database (SGD) provides secondary gene annotation using Gene Ontology (GO). Nucleic Acids Research 30, (2002)
25. The gene ontology consortium: Gene ontology: Tool for the unification of biology. Nature Genetics 25, (2000)