

Bounded Diameter Clustering Scheme For Protein Interaction Networks

Nassim Sohaee, Christian V. Forst *

Abstract—Dense subgraphs of Protein-Protein Interaction (PPI) graphs are assumed to be potential functional modules and play an important role in inferring the functional behavior of proteins. PPI graphs are known to be scale free, and this property makes the process of isolation of dense subgraphs very hard. This paper describes a new graph theoretic clustering algorithm, that detects densely connected regions in large PPI graph. The method is based on finding bounded diameter subgraphs around a seed node. The algorithm has the advantage over other graph clustering methods of being very simple and efficient. This algorithm is tested on yeast PPI graph and the results are compared with some of the existing algorithms.

Keywords: *Protein network, protein complex, dense subgraph, clustering*

1 Introduction

Proteins are important components of life. It is known that proteins do not act individually. In fact, to fully understand the cell machinery, simply listing, identifying and determining the function of proteins in isolation is insufficient [1]. Biological processes are performed by groups of proteins rather than by an individual protein. The interactions between proteins are important for many biological functions. Protein-protein interactions (PPI) are fundamental for virtually every process in a living cell. Information about these interactions improves our understanding of diseases and can provide the basis for new therapeutic approaches. An investigation of PPI mechanisms begins with the representation and characterization of the PPI graph structure. The simplest representation is a graph, with proteins as nodes and two nodes are connected if and only if the corresponding proteins interact physically [17]. Since the information about protein interactions predominantly originates from experimental data, different experimental environments may result in different outputs. Hence, we can consider the reliability of each interaction as the edge weight of the interaction graph. In a weighted graph model, the weight of each edge is a positive number between 0 and 1. The closer to 1, the more reliable is the interaction.

The PPI graph is a scale free network, which means the degree distribution follows a power law [11, 15, 16, 26]. Hence, most proteins participate in few interactions, and few proteins participate in a large number of interactions. The topology of the PPI graph consist of a few central cores of proteins while a significant amount of interactions and the rest of the proteins are either disconnected or connected to a small number of proteins. Moreover, nodes with high degrees are usually not connected to each other [20].

1.1 Clustering PPI networks

The goal of clustering a graph is to find a set of nodes that share some common properties. Similarly, clustering in biological networks is being applied to identify some biological relevant functions. Specifically, in PPI graphs the goal is to find proteins with similar functionalities. Proteins in a highly connected subgraph of PPI usually share a common function [5]. Therefore, a highly connected subgraph like clique or near clique in a PPI graph can be used to predict the function of uncharacterized proteins [7]. Finding a clique with a maximum size is a NP-complete problem. Hence, some approximation algorithms have been developed to give some near optimal solutions.

Spirin and Mirny [23] proposed an enumeration method to count the number of complete subgraph of a given graph. In general, listing all complete subgraphs of given graph is NP-complete. However, they used the fact that if a subgraph is not a clique none of its supergraphs are cliques. The process starts with a small size clique and adds new nodes to enlarge the clique as much as possible. This approach has several drawbacks. First, it is just finding all fully connected subgraphs; however, in reality, we may possibly have a near clique and not a fully connected subgraph. Second, the enumeration method is very slow, especially when the graph has a large number of nodes and edges. To overcome these problems, they introduced a new approach for finding highly connected, but not necessarily fully connected subgraphs. In their later approach they formulated the problem as an optimization problem to identify a set of n nodes that maximize the object function $\frac{2m}{n(n-1)}$, where m is the number of edges in the induced graph over n nodes. Hence, when the objective function is 1, the subgraph is a clique. A

*Department of Clinical Sciences, University of Texas Southwestern Medical Center, 5323 Harry Hines Blvd, Dallas, TX 75390-9066; {nassim.sohaee, christian.forst}@utsouthwestern.edu

Monte Carlo approach is used to optimize the procedure. Samanta and Liang [21] used a statistical method to find proteins that share a significantly large number of common neighbors. Subsequently, they found the statistical significance of each pair of proteins, and combined the pairs of proteins with least value. Bader and Hogue [2] described a clustering algorithm based on vertex weighting by local neighborhood density and outward traversal from a locally dense seed protein to isolate the dense regions according to given parameters. Yu and et al. [1, 27] introduced a soft clustering method. This method softly assigns data to clusters in a probabilistic way. A hierarchical clustering derived in this method merges the low level clusters into high level ones. Cui et al. [7] introduced an algorithm for finding cliques and near cliques in PPI network. They proved that their method results in clusters that share at least one common function. Bader, Christopher and Higue [2] presented a clustering algorithm, MCODE. The molecular complex detection algorithm, MCODE, consists of three phases: vertex weighting, complex prediction and optimal post processing step. The weighting of nodes is based on the core clustering coefficient instead of standard clustering coefficient to increase the weight of heavily interconnected vertices. Once the weights are computed, the algorithm traverses the weighted graph in a greedy fashion to isolate densely connected regions. The post processing step filters or adds proteins based on connectivity criteria.

So far, one of the most successful clustering procedures used in deriving complexes from protein interaction networks seems to be the Markov Cluster algorithm, MCL[9]. Unlike most hierarchical clustering procedures, this algorithm considers the connectivity properties of the underlying network. Recently, in a novel promising clustering procedure termed Affinity Propagation, AP, was proposed [10]. Affinity propagation identifies representative examples within the dataset by exchanging real-valued messages between all data points. Points are then grouped with their most representative exemplar to give the final set of clusters. Vlasblom and Wodak [25] compared the performance of the AF and MCL procedures. Their analysis shows that the MCL procedure is significantly more tolerant to noise and behaves more robustly than the AP algorithm. MCL thus remains the method of choice for identifying protein complexes from binary interaction networks.

There is another type of clustering which was proposed by Girvan-Newman [12]. The Girvan and Newman algorithm is a methods used to detect modules in complex systems. The notion of a module is related to clustering, though it isn't quite the same. A module consists of a subset of nodes within which the node-node connections are dense, and the edges to nodes in other modules are less dense [12]. The original Girvan-Newman algorithm does not include a clear definition of module. Several def-

inition of modules based on different criteria have been proposed. Lue and et al. defined a module in a network as a subgraph that has more internal edges than external edges [19]. They also provided an algorithm to find the modules in a PPI graph.

As we mentioned above, many network cluster identification algorithms have been developed. However, each algorithm might dissect a network from a different aspect that may provide different insight on the network partition. Dong, Bing and Han [8] evaluated the performance of different cluster detection algorithms. They compared the biological coherence of the network clusters by different algorithms. In particular, they evaluated the clusters found by the algorithms MCL, MCODE and Girvan-Newman. The comparison of the resulting network clusters indicated that clusters found by MCL and MCODE have a higher biological coherence than those by the Girvan-Newman algorithm.

1.2 Our Contribution

Accumulating results suggest that proteins with smaller hop distance in PPI are most likely to have similar functionalities. In this paper we present a new method of clustering. The goal of the algorithm is to find clusters that the distance of any two proteins within the cluster is bounded. The Bounded Diameter algorithm, BD, is a recursive algorithm that finds a maximal bounded diameter subgraph of PPI graph. The algorithm recursively calls a subroutine that take a maximal bounded diameter subgraph of diameter d and returns a maximal bounded diameter subgraph of diameter $d + 1$. Finally, the results of our method on Yeast PPI are compared against the results of MCODE and MCL algorithms. Based on [25] and [8] the clusters found by MCL and MCODE have a higher biological coherence.

2 Clustering Algorithm

The classic computational approach to find protein complexes is based on the connections in the network. The common principal in this method is that proteins that lie closer to one another in PPI graph are more likely to have similar functions. As it can be seen in Figure 1, there is a high correlation between the similarity of two proteins and their hop distance in PPI graph [22]. Hence, the closer the two proteins are in the network the more similar are their function.

One of the simplest methods to predict the function of a protein is based on the known function of its direct neighbors. Although this method is very simple, it is not effective as it does not assign very significant values. One of the reasons for this could be that the topology of the network is not fully considered. Experiments with proteins suggest that dense subgraphs of PPI graph represent a biological meaningful unit such as functional module or

protein complex. In literature, density of a graph is being measured by different methods. Mostly, a graph is called dense if the number of edges is close to the total number of edges in a complete graph with the same number of nodes. Considering that a complete graph over n node has $\frac{n(n-1)}{2}$ edges, it is natural to consider the density factor of a graph G to be $\frac{2m}{n(n-1)}$, where m is the number of edges in G . Finding all subgraphs of a given graph with a fix density factor is an NP-complete problem.

As we mentioned previously in this paper, there are a lot of algorithms proposed to find approximately maximum dense subgraphs. To the best of our knowledge, none of these algorithms consider a bounded diameter near clique subgraphs. As Figure 1 suggested, proteins with similar functionalities are usually within two hop distance from each other. This is the main motivation of our algorithm.

2.1 Definition and Notations

Let $G = (V, E)$ be an undirected graph, and H be a subset of V . We define $deg_H(v)$ to be the number of neighbors of v in H , and $deg(v) = deg_G(v)$. $N(v)$ is the set of direct neighbors of node v and $N_H(v) = N(v) \cap H$. We define $N(S) = \cup_{v \in S} N(v)$ where $S \subseteq V$. Similarly we can define $N_H(S)$. We note $diam(K)$ as diameter of subgraph K of G . A graph is 2-connected if there does not exist any node whose removal disconnects the graph.

2.2 Algorithm for Finding Near Complete Subgraph

In this section a new recursive algorithm is presented to identify clusters of bounded diameters within a protein interaction network. The Bounded Diameter Clustering algorithm starts at single point in the network as seed node. This algorithm recursively calls a subroutine to compute a maximal bounded diameter subgraph of the network. The theoretical foundations for this algorithm are as follow.

Lemma 1. Let $H = (V_H, E_H)$ be a subgraph of $G = (V, E)$ of diameter d . Suppose u is a node in $V - V_H$ such that diameter of $H + u$ increases to $d + 1$. Let v be another node in $V - V_H$. $diam(H + u + v) = diam(H + u)$ if $\emptyset \neq N_H(v) \subseteq N_H(u)$.

Proof. It can be easily observed that the length of the shortest path from each node in H to v is at most $d + 1$. Since $N_H(v) \subseteq N_H(u)$, $dist(u, v) \leq 2$. Hence $diam(H + u + v) = diam(H + u)$. \square

Lemma 2. Let H be a maximal subgraph of G of diameter d . Let $\{v_1, \dots, v_l\}$ be a set of nodes in $G - H$. If $\cap_{i=1}^l N_H(v_i) \neq \emptyset$ then $diam(H + \{v_1, \dots, v_l\}) = d + 1$.

Proof. H is a maximal subgraph of G of diameter d . Adding any node to H will increase the diameter of the

new graph. Let $v \in \cap_{i=1}^l N_H(v_i)$. For any node u in H , $d(u, v_i) \leq d(u, v) + d(v, v_i) \leq d + 1$, $i = 1, \dots, l$. \square

Algorithm 1 Bounded Diameter maximal Subgraph(H,G)

Input: graph $G = (V, E)$ and a subgraph $H = (V_H, E_H)$ of diameter d

Output: a maximal subgraph of G containing H and of diameter $d + 1$

```

1:  $S = N(V_H)$ ,  $T = V_H$ ,  $P = V_H$ 
2: while  $S$  and  $P$  are nonempty do
3:   let  $v$  be a node with maximum  $deg_T$ 
4:   if  $deg_S(v) \neq 0$  then
5:      $T = T \cup \{v\}$ ,  $P = P \cap N_H(v)$ ,  $S = S \cap N_S(v)$ 
6:   else  $P = P - \{v\}$ 
7:   end if
8: end while
9: Return  $G[T]$ 

```

The input of the algorithm 1 is a maximal subgraph of diameter d and the output is a maximal subgraph of diameter $d + 1$. The algorithm starts with the set of all neighbors of all nodes in the input graph, S . T is the set of all nodes in the output graph. $T = V_H$ grants that the output graph contains the input graph as a subgraph. The first node that is selected to be in T is a node of highest degree in S . A node could be a candidate node for the next step selection, if and only if adding it to the graph found in previous step does not increase the diameter. The set of candidate nodes, in initial step, is the same as set S . After each node selection, set P will be updated to $P \cap N_H(V)$. This satisfies the conditions of Lemma 2. Therefore, the diameter of the output graph will not exceed $d + 1$. The algorithm stops when there are no more nodes that can be added to the output graph. This happens either if there are no more nodes left in S or adding any other node will increase the diameter of the graph to more than $d + 1$. This algorithm is a subroutine of algorithm 2.

Algorithm 2 Bounded Diameter Clustering Algorithm(d)

Input: seen node v of graph $G(V, E)$ and diameter d

Output: a maximal subgraph of G of diameter d containing the seed node and

```

1:  $pivot = v$ 
2: for  $i = 1$  to  $d$  do
3:    $pivot = \text{Bounded - Diameter - Maximal -}$   

    $\text{Subgraph}(pivot, G)$ 
4: end for
5: Return  $pivot$ 

```

The running time of algorithm 2 is $O(dn)$, where n is the number of nodes in the graph G . The algorithm starts at a seed node, and returns a maximal subgraph of diameter

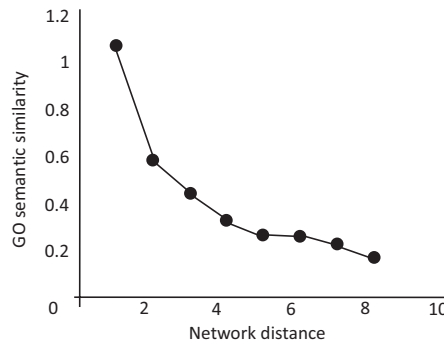


Figure 1: Correlation between protein functional distance and network distance. x -axis: distance in network. y -axis: average functional similarity of protein pairs that lie at the specified distance. The functionality of two proteins is measured using the semantic similarity of their gene ontology categories. (This figure is from a paper by Sharan, Ulitsky and Shamir ¹⁷)

d . Line 3 of the algorithm recursively calls the algorithm 1 as a subroutine. The diameter of the output graph, d , is determined by the user. However, Figure 1 suggests that the best value is 2. Later, we will test BD algorithm for $d = 2$.

Theorem 1. *The output of algorithm 2 is a maximal subgraph of diameter d and containing the input graph.*

Algorithm 2 is very fast in theory, and has several advantages over all other existing algorithms. Most other fast and efficient algorithms need also several extra steps including, preprocessing of inputs and postprocessing of the results. For example, MCODE post processes the results to filter complexes that do not contain at least a graph of minimum degree 2. In this algorithm a new node is added to a cluster if some predefined parameters are above the threshold.

As we stated before, dense subgraphs of a PPI network are usually carrying important information regarding common functionalities and complexes. The density factor of a subgraph is a measure to show how close the subgraph is to the complete graph. Here we will show that the density of subgraphs found by algorithm 2 are always larger than a threshold, which is a function of the diameter and the number of nodes in the subgraph. Belotserkosky [3] showed that there is a relation between the number of edges in a 2-connected subgraph and its diameter.

Theorem 2. *Let $G = (V, E)$ be a 2-connected graph of diameter d . If $|V|$ is sufficiently large compared to d , then $|E| \geq \lceil (dn - 2d - 1)/(d - 1) \rceil$, where $|V| = n$.*

In this paper we are just considering subgraphs of diameter at most 2. The reason is that the average functional similarity will decrease to less than half for PPI subgraphs of diameter more than 2. The output of algorithm 2 may not be a 2-connected graph. In fact, the graph may contain several leaf nodes. However, based on the way that

the algorithm is choosing nodes, all leaf nodes have the same parent in the subgraph. Hence, the result has a 2-connected block, and a set of leaf nodes connected to one of the nodes in the 2-connected block.

Theorem 3. *Suppose $G = (V, E)$ be an output subgraph of algorithm 2 of diameter 2. The density factor of G is at least $\frac{4}{n-1}$.*

Proof. Let V_1 and V_2 be the set of nodes in a 2-connected block and leaf nodes, respectively. Let E_1 be the set of edges in the induced subgraph $G[V_1]$ and E_2 be the rest of edges. Then $|E| = |E_1| + |E_2|$. Lemma 2 implies that $|E_1| \geq 2|V_1| - 5$. Hence, $|E| \geq 2|V_1| + |V_2| - 5 = |V| + |V_1| - 5$. The density of the graph G can be computed as follows,

$$\begin{aligned} \frac{2|E|}{|V|(|V| - 1)} &\geq \frac{2(|V| + |V_1| - 5)}{|V|(|V| - 1)} \\ &\geq \frac{2}{|V| - 1} + \frac{2|V_1|}{|V_1|(|V| - 1)} \geq \frac{4}{|V| - 1}. \end{aligned}$$

□

3 Results and Discussions

Identifying hidden topological structures of protein interaction networks often unveil biologically relevant functional groups and structural complexes. We have developed an efficient algorithm for finding bounded diameter subgraphs in protein interaction networks. Here, we compare the BD algorithm with two algorithms commonly utilized for extracting functional modules from PPI graphs, MCODE and MCL. A recent study [4] compared these algorithms (among others) showed that the MCL algorithm, in particular, was very efficient in identifying protein complexes from protein interaction networks.

3.1 Comparison of BD and MCODE

We tested the algorithm on a dataset with 14131 interactions among 4623 yeast proteins, which is the combined data of Ito et al. [13], and Uets et al. [24]. In this section we compare the results of our algorithm and MCODE. We analyze the highest ranked clusters by MCODE and the corresponding BD algorithm clusters using the *Cellular Component* ontology to compare the effectiveness of these algorithms, in terms of identifying protein complexes. The first best scoring cluster in MCODE is composed of 26 proteins that all belong to the known complex, anaphase promoting complex (GO:31145). BD algorithm did capture the same complex with exact same set of proteins as one of its clusters.

We use gene ontology, GO, level as a measure of cluster function. The gene ontology is structured as directed acyclic graphs with exactly one source node, *all*. We can define the GO level of a gene as the length of the longest path from a specific gene to *all*. If a gene is connected to *all* via a relatively short path, the gene is common among a relatively large class of proteins. Hence we can consider it as less specific. On the other hand, if the length of the path is relatively large, the gene is listed in a smaller number of proteins. Figure 2(left) illustrates the GO level of protein clusters found by MCODE and BD. As this figure is suggesting clusters found by BD represent more specific information regarding the gene function of the cluster.

The p-values give a good indication about the prominence of a given functional category. Figure 2(right) shows that the p-value of both algorithms are in the same range. Another factor we tested was the similarity factor of each cluster. The similarity factor of each cluster can be defined as This factor shows how similar the nodes in a cluster are, in term of functionality. Obviously, the desire is to capture protein clusters with high similarity factors. Figure 3(a) shows that the similarity factor of clusters founded by MCODE and BD are very close. However, some of the clusters that are found by MCODE are cliques in the PPI network, which are supposed to show a higher rate of similarity among their proteins. Figure 3(b) is showing the relation of similarity factor of non-clique clusters found by MCODE verses the similarity factor of clusters founded by BD. As this figure illustrates, BD algorithm achieves a better result in terms of similarity.

3.2 Comparison of BD and MCL

Next, we compared the clusters obtained by the MCL algorithm with the ones from BD. The two-hybrid interactions of the budding yeast (*Saccharomyces cerevisiae*) [13], [14] is being used for this comparison. This dataset contains 4549 interactions among 3282 proteins. This dataset contains extra information such as *literature shar-*

ing score, which has been used as the edge weight for MCL algorithm. Like previous section we compare the clusters based on p-value distribution, GO level and similarity factor.

4 Conclusion

Identifying protein clusters within a biological systems is essential for understanding of the high-level organization of the cell. In this study, we implied the fact that proteins with small hop distance in PPI network usually have similar functionalities [22]. A new recursive algorithm was designed to capture clusters of bounded diameters from protein interaction networks. Our new approach is based on the topological characteristics of the network. In the proposed algorithm, first a maximal clique is identified around the seed node in PPI network. Then by adding nodes progressively the clique is extended to maximal subgraph of diameter d , where d is a user defined value. We compared BD with MCODE and MCL. Computer experiments showed that BD has a superior precision in complex prediction.

References

- [1] A. Asur, S. Parthasarathy and D. Ucar, "An ensemble framework for clustering protein-protein interaction networks," *Bioinformatics*, vol. 23, no. 13, pp. 29-40, 2007.
- [2] G. D. Bader and C. W. Hogue, "An automated method for finding molecular complexes in large protein interaction networks," *BMC Bioinformatics*, vol. 4, pp. 2, 2003.
- [3] D. Belotserkovsky, "The smallest number of edges in a 2-connected graph with specific diameter," *Discrete Mathematics*, vol. 307, pp. 2376-2384, 2007.
- [4] S. Borhe and J. van Helden, "Evaluation of clustering algorithms for protein-protein interaction networks," *BMC Bioinformatics*, vol. 7, no. 488, 2006.
- [5] D. Bu, Y. Zhao, L. Cai and et al., "Topological structure analysis of the protein-protein interaction network in budding yeast," *Nucleic Acids Research*, vol. 31, no. 4, pp. 2443-2450, 2003.
- [6] M. Charikar, "Greedy approximation algorithms for finding dense components in a graph," *Lecture notes in computer science*, vol. 1913, pp. 84-95, 2000.
- [7] G. Cui, Y. Chen, D. Huang and K. Han, "An algorithm for finding functional modules and protein complexes in protein-protein interaction network," *Journal of Biomedicine and Biotechnology*, vol. 2008, Article ID 860270, 10 pages, 2008. doi:10.1155/2008/860270

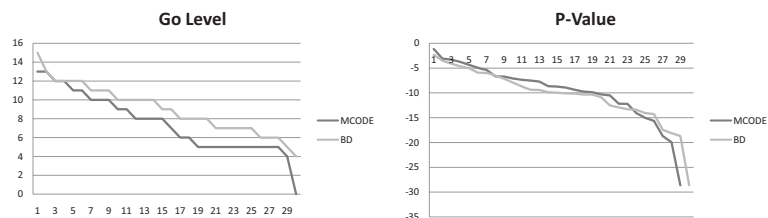


Figure 2: (left) The Gene Ontology level of most significant gene in each cluster found by MCODE and BD. (right) The lg(p-value) of most significant gene in each cluster found by two algorithms MCODE and BD.

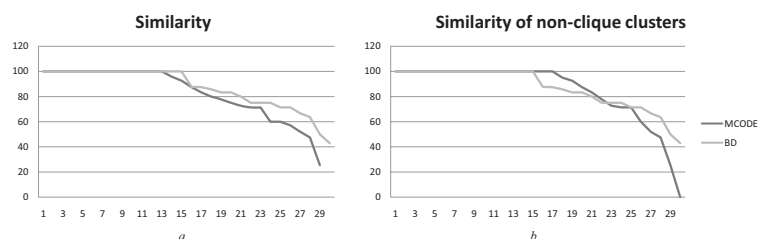


Figure 3: (a) the similarity factor of clusters found by MCODE and BD. Some of clusters found by MCODE are cliques. (b) similarity of none-clique clusters of MCODE and BD.

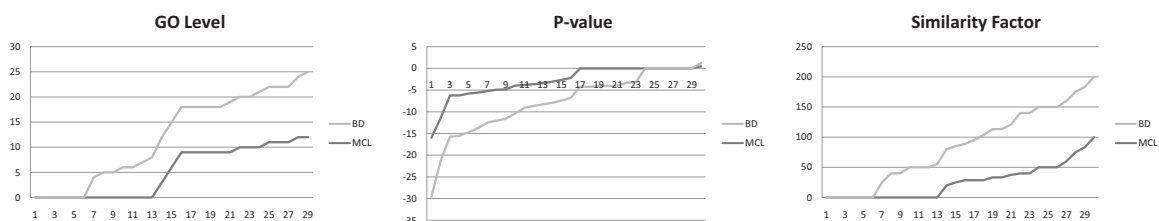


Figure 4: (left) The Gene Ontology level of most significant gene in each cluster found by MCL and BD. (middle) The lg(p-value) of most significant gene in each cluster found by two algorithms MCL and BD. (right) The similarity factor of clusters found by MCL and BD.

- [8] D. Dong, Z. Bing, and J. J. Han, "Comparing the biological coherence of network clusters identified by different detection algorithms," *Chinese Science Bulletin*, vol. 52, no. 21, pp. 2938-2944, 2007.
- [9] S. van Dongen, "Graph Clustering by flow simulation," Ph.D. Dissertation, University of Utrecht, 2000.
- [10] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315(5814), pp. 972-976, 2007.
- [11] L. Giot and et al, "A protein interaction map of *Drosophila melanogaster*," *Science*, vol. 302, pp. 1727-1736, 2003.
- [12] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceeding of National Academic of Science*, vol. 99, pp. 7821-7826, 2002.
- [13] T. Ito, T. Chiba, R. Ozawa, M. Yoshida, M. Hattori and Y. Sakaki, "A comprehensive two-hybrid analysis to explore the yeast protein interactome," *Proc. Natl. Acad. Sci.*, vol. 98, pp. 4569-4574, 2001.
- [14] T. Ito, K. Tashiro, S. Muta, R. Ozawa, T. Chiba, M. Nishizawa, K. Yamamoto, S. Kuhara and Y. Sakaki, "Toward a protein-protein interaction map of the budding yeast: a comprehensive system to examine two-hybrid interactions in all possible combinations between the yeast proteins," *Proc. Natl. Acad. Sci.*, vol. 97, pp. 1143-1147, 2000.
- [15] H. Jansen, S. P. Mason, A. -L. Barabasi and Z. N. Oltvai, "Lethality and centrality in protein networks," *Nature*, vol. 411, pp. 41-42, 2001.
- [16] S. Li and et al., "A map of the interactome network of the metazoa," *Science*, vol. 303, pp. 540-543, 2004.
- [17] C. Lin, Y. Cho, W. Hwang, P. Pie and A. Zhang, "Knowledge Discovery in Bioinformatics: Techniques, Methods and Applications," John Wiley & Sons, Inc. 2006.
- [18] P. W. Lord, R. D. Stevens, A. Brass and C. A. Goble, "Investigating semantic similarity measures across the gene ontology: the relationship between sequences and annotation," *Bioinformatics*, vol. 19, pp. 1275-1283, 2003.
- [19] F. Luo, Y. Yang, C. Chen, R. Chang, J. Zhou and R. H. Scheuermann, "Community structure in social and biological networks," *Proceeding of National Academic of Science*, vol. 99, pp. 7821-7826, 2002.
- [20] S. Maslov and K. Sneppen, "Specificity and stability in topology of protein networks," *Science*, vol. 296, pp. 910-913, 2002.
- [21] M. P. Samanta and S. Liang, "Redundancies in large scale protein interaction networks," *Proceeding of national academic of science*, vol.100, pp.12579-12583, 2003.
- [22] R. Sharan, I. Ulitsky and R. Shamir, "Network-based prediction of protein function," *Molecular Systems Biology*, 3:88, 2007.
- [23] V. Spirin and L. A. Mirny, "Protein complexes and functional modules in molecular networks," *Proceeding of national academic of science*, vol. 100, pp. 12123-12128, 2003.
- [24] P. Uetz et al., "A comprehensive analysis of protein-protein interactions in *saccharomyces cerevisiae*," *Nature*, 403:623, 2000.
- [25] J. Vlasblom and S. J. Wodak, "Markov clustering versus affinity propagation for the partitioning of protein interaction graphs," *BMC Bioinformatics*, 10:99, 2009.
- [26] A. Wagner, "The yeast protein interaction network evolves rapidly and contains a few redundant duplicate genes," *Molecular Biology Evolution*, vol. 18, pp. 1283-1292, 2001.
- [27] K. Yu, S. Yu and V. Tresp, "Soft Clustering on Graphs," *Advanced in Neural Information Processing Systems*, vol. 19, 2005.
- [28] Carpenter, R.H.S., *Movements of the Eyes*, 2nd Edition, Pion Publishing, 1988.
- [29] Franklin, G.F., Powel, J.D., Workman, M.L., *Digital Control of Dynamic Systems*, Second Edition, Addison-Wesley, 1990.
- [30] Oh, P.Y., Allen, P.K., "Design a Partitioned Visual Feedback Controller," *IEEE Int Conf Robotics & Automation*, Leuven, Belgium, pp. 1360-1365 5/98
- [31] Oh, P.Y., Allen, P.K., "Visual Servoing by Partitioning Degrees of Freedom," *IEEE Trans on Robotics & Automation*, V17, N1, pp. 1-17, 2/01