Hierarchical Modular Structure in Gene Coexpression Networks

Shuqin Zhang
Center for Computational Systems Biology
School of Mathematical Sciences
Fudan University
Shanghai, 200433, China
Email: zhangs@fudan.edu.cn

***Abstract*—Network module (community) structure has been a hot research topic in recent years. Many methods have been proposed for module detection and identification. Hierarchical structure of modules is shown to exist in different kinds of biological networks. Compared to the module identification methods, less research is done on the hierarchial structure of modules. In this paper, we propose a method for constructing the hierarchical modular structure in networks based on the extended random graph model. Statistical tests are applied to test the hierarchial relations between different modules. We give both artificial networks and real data examples to illustrate the performance of our approach. Application of the proposed method to yeast gene co-expression network shows that it does have a hierarchical modular structure with the modules on different levels corresponding to different gene functions.**

I. Introduction

Network has been widely applied for modeling complex systems, including biological systems, social organizations, World-Wide-Web, and so on. In a network, the nodes (vertices) represent the members in the system, while the edges represent the interactions among the members. If two nodes have interactions in a network, there will be an edge connecting them. With such a representation, the complex systems can be analyzed by computational methods.

Module (community) structure is a common property of many different types of networks. Modules are the dense subgroups of a network, where the nodes in the same module are more likely to connect each other than the other nodes. In general, the members in the same module share some common properties or play similar roles. For example, in a gene co-expression network, the genes in the same module may belong to the same functional category such as lipid metabolism and acute-phase response [1]. Since the paper published by [2], module detection and identification becomes a hot research topic in several different areas such as computer science, physics, and statistics. A large number of related works have been published with the physicists making the most contributions [3], [4], [5], [6], [7], [8], [9], [10], [11], [12]. Among these methods, modularity optimization has attracted much attention [7], [8], [14], [13]. In the paper by Newman [7], [8], modularity measures the difference between the number of edges within groups in the network and the expected number of such edges in an equivalent network where the edges are placed at random. By optimizing modularity, the partitioning of the network into modules is obtained. One important problem with this modularity is on the resolution limit [6]. When the size of the module is smaller than certain value, it cannot be resolved, which depends on the size of the subnetworks to be divided and the interconnectedness of the subgroups. Several modifications are presented to improve this modularity later on [19], [20], [21], [22], [11]. Some statistical property of this modularity is analyzed in [13]. There the author shows that this modularity cannot identify the modules consistently. At the same time, a novel modularity is proposed in the paper, which can consistently recover the modules in dense networks. However, the computation of maximizing this modularity is very time-consuming. Li *et al.* proposed another modularity in [11]. This modularity is shown to perform better than the modularity proposed by Newman and it can improve the resolution. However, resolution limit is still a problem with their proposed criterion of choosing the number of modules. Besides these methods based on modularity, some other proposed methods also give good identification of modules. For example, [12] gives an information-theoretic framework for module identification, and the method works well. Several recent review papers provide details and comparison of the module identification methods [16], [6], [9]. [16] compares the performance of several existing methods for both computation time and output. [6] is a thorough, more recent discussion. [9] contrasts different perspectives of the methods and sheds light on some important similarities of several methods.

Although so many related works are published, how to choose an appropriate number of modules keeps being an open problem. Different methods output different solutions of the number of modules when they are applied to the same network. In reality, all of the different choices may be reasonable since different choices of this number may correspond to the modules on different levels. As explained in [17], some modular networks may have hierarchical structure. For example, in a friendship network, on the large scale, the modules may correspond to people from different countries. If we look at the modules on the smaller scales, people in the same module may graduate from the same university, grow up in the same community, or even be born in the same family. Such hierarchical modular structure appears in different

978-1-4673-4398-5/12/$31.00 ©2012 IEEE

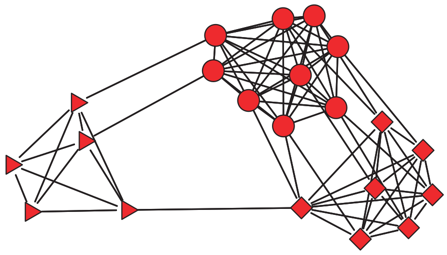


Fig. 1. Example of hierarchial modular network structure.

kinds of networks. For example, Meunier and colleagues gave an example on hierarchical modular structures in human brains [23]. Fig. 1 shows an example of hierarchical modular network. There are two levels of the modules. We can identify three modules corresponding to different shapes of nodes on the lowest level or two modules with nodes represented by cubes and circles being combined together on the higher level.

Compared to the module identification in a partitional way, there are much fewer works on computational methods for hierarchical modular structure analysis [24], [25], [26]. Although these papers present some methods to construct the hierarchial modular structure, they do not give a clear picture on how these modules are organized, what the relationship among the modules is. In this paper, we mainly consider the problem of hierarchical modular structure in unweighted networks. We give our proposed method in section II, where we give the methods on how to find all the possible modules in a network and how to construct the hierarchical structure from these modules. Numerical experiments for both simulated networks and real data networks are presented to show the performance of our proposed method in Section III. The application of the proposed method to yeast gene co-expression network shows that it does have a hierarchical structure, which corresponds to the different levels of gene functions. Conclusion remarks are given finally.

II. Methodology

Before going to the details on how to construct the hierarchical structure, we give its definition first. We consider a network $G(V, E)$ with n nodes, where V denotes the set of nodes and E denotes the set of edges. The adjacency matrix is denoted as A with each entry being 0 or 1. The hierarchical structure of a network is defined based on the random modular network model, which is a direct extension of the Erdös-R$\acute{e}$nyi random graph model [27]. A random graph is obtained by starting with a set of n nodes and adding edges between them in a probabilistic fashion. The presence of an edge between any two nodes is a Bernoulli event where the probability may be vertex-pair dependent. Suppose any node has a probability μ_i to be in the module M_i, where $\mu = (\mu_1, \mu_2, \cdots, \mu_K)$ satisfies $\sum_{i=1}^{K} \mu_i = 1$. Then any two nodes $u, v \in V$ and $u \in M_i, v \in M_j$ are connected with probability $P_{i,j}$ depending on M_i, M_j, and P is symmetric. If there is the modular structure in the network, then $P_{i,j} < \min\{P_i, P_j\}$. For a network composed of three modules M_i, M_j, and M_k, if $P_{i,j} > \max\{P_{i,k}, P_{j,k}\}$, then we say there is hierarchical structure among these three modules. If there are K modules in the network, the hierarchical structure can be defined recursively.

To construct the hierarchical structure, we look at the partitional case first, that is all the modules are on the same level. We suppose the number of modules K is given and we aim to find all the possible modules. We let N_k denote the number of nodes in subnetwork V_k, L_{kk} denote twice the total number of edges in subnetwork V_k, and L_{kl} denote the total number of connections between the subnetworks V_k and V_l, where $k, l = 1, 2, \cdots, K$. The module identification problem is formulated as:

$$\max_{\mathbf{P}} \Phi(\mathbf{P}) = \Phi_1(\mathbf{P}) - \Phi_2(\mathbf{P}),$$

where

$$\Phi_1(\mathbf{P}) = \sum_{k=1}^{K} \frac{L_{kk}}{N_k}, \Phi_2(\mathbf{P}) = \sum_{k=1}^{K} \sum_{l \neq k} \frac{L_{kl}}{N_k}.$$

Here $\mathbf{P}$ is a partition of the network. This metric has been presented in [11].

In matrix form, if we let

$$S_{ik} = \begin{cases} 1, & \text{if node } i \in V_k \\ 0, & \text{otherwise} \end{cases} \quad i = 1, 2, \cdots, n.$$

Then, the problem is formulated as:

$$\begin{aligned} \max \quad & \Psi(S) = \sum_{k=1}^{K} \frac{S^T_{.,k} A S_{.,k}}{S^T_{.,k} S_{.,k}} - \sum_{k=1}^{K} \sum_{l \neq k} \frac{S^T_{.,k} A S_{.,l}}{S^T_{.,k} S_{.,k}} \\ \text{s.t.}: \quad & S_{i,j} \in \{0, 1\} \text{ for } i, j = 1, 2, \cdots, K, \\ & \sum_{k=1}^{K} S_{.,k} = \mathbf{1}. \end{aligned} \tag{1}$$

Here $\mathbf{1}$ is a vector with all elements being 1.

The function $\Phi_1(\mathbf{P})$ defines the sum of the average degrees in each subnetwork and $\Phi_2(\mathbf{P})$ defines the sum of the average number of connections between one subnetwork and other subnetworks. The objective function aims to both maximize Φ_1 and minimize Φ_2 since Φ_1 and Φ_2 may lead to inconsistent results when applied separately. By maximizing $\Phi(\mathbf{P})$, we expect to achieve a good balance and make correct inference on the modules.

To solve the problem (1), an approximate method is applied. Let $\tilde{S}_{.,k} = \frac{S_{.,k}}{\|S_{.,k}\|_2}$, the function $\Psi(S)$ can be approximated as $\text{Tr}(\tilde{S}^T A \tilde{S}) - \text{Tr}(\tilde{S}^T L \tilde{S}) = \text{Tr}(\tilde{S}^T (2A - D) \tilde{S})$, and we aim to solve the optimization problem:

$$\begin{aligned} \max \quad & \tilde{\Psi}(\tilde{S}) = \text{Tr}(\tilde{S}^T (2A - D) \tilde{S}) \\ \text{subject to}: \quad & \tilde{S}^T \tilde{S} = I. \end{aligned}$$

978-1-4673-4398-5/12/$31.00 ©2012 IEEE

The problem of maximizing $\tilde{\Psi}(\tilde{S})$ is the standard form of a trace optimization problem. Its solution can be obtained from the Rayleign-Ritz theorem. The solution can be approximated by the eigenvectors corresponding to the K largest eigenvalues of the matrix $2A - D$. To obtain a binary matrix S, which defines the partition of the network, the K eigenvectors are applied to do the k-means clustering for module assignments. By adding maximization of the sum of the average degrees in each subnetwork, the network can be divided into comparatively large modules. The algorithm is summarized in the following:

Algorithm:

Input: Adjacency matrix $A_{n\times n}$, and K, which is the number of modules.

```
1) Compute the matrix $2A - D$;
2) Compute the last $K$ eigenvectors $\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_K$ of matrix $2A - D$;
3) Construct a new matrix $T \in R^{n\times K}$, with columns $\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_K$;
4) Cluster the points constructed from each row of matrix $T$ with $k$-means clustering into modules $M_1, M_2, \cdots, M_K$;
```

Output: Index of nodes in each module.

With the above algorithm, we can get a partition of the network into modules. Now, we discuss how to determine the lowest level of all the possible modules. For any node $i \in V$, the degree can be written as:

$$d_i = \sum_{k=1}^{K} d_i(V_k),$$

where

$$d_i(V_k) = \sum_{j\in V_k} A_{ij},$$

which defines the connections that node i has in the subnetwork V_k. To determine the number of possible modules, we consider the average connectivity within a subnetwork and that between it and any other subnetwork. If the average connectivity within a subnetwork is greater than that between subnetworks, we take it as a module, that is:

$$\frac{\sum_{i\in V_k} d_i(V_k)}{N_k} > \frac{\sum_{i\in V_k} d_i(V_l)}{N_k}, l \neq k. \qquad (2)$$

Alternatively, it can also be written as:

$$L_{kk} > L_{kl},$$

if we multiply both sides with $N(V_k)$.

We do the clustering for K increasing from two until the condition (2) does not hold. Now we get all the possible modules. The efficiency of the above algorithm for identifying partitional modules can be seen in [31]. The details of the above method can be found in [31].

Based on the above results, we construct the hierarchial structure in an agglomerative way (bottom-to-up). The distance between any two modules is defined as one minus their connection probability, which is computed from the clustering results through maximum likelihood estimation. This connection probability matrix is denoted as $\hat{P}^0$. First the maximum connection probability between different modules is found, and we assume it is $\hat{P}^0_{i_0,j_0}$. The corresponding two modules i_0, j_0 are recorded. The second largest connection probability for these two modules i_0, j_0 are also found, and we assume they are $\hat{P}^0_{i_0,k_0}$, and $\hat{P}^0_{j_0,l_0}$. The corresponding modules k_0, l_0 are also recorded. To test whether there is a hierarchial structure for these modules, we use Fisher exact test to see whether the connection probability $\hat{P}^0_{i_0,k_0}$, and $\hat{P}^0_{j_0,l_0}$ are the same as $\hat{P}^0_{i_0,j_0}$. That is, we need to test $\hat{P}^0_{i_0,j_0} = \hat{P}^0_{i_0,k_0}$ and $\hat{P}^0_{i_0,j_0} = \hat{P}^0_{j_0,l_0}$. Here we take a p-value threshold to be 0.05. Three different cases may occur. (1) Both of these two null hypotheses are rejected. Now we say there is hierarchical structure and the modules i_0, j_0 are on the lower level than k_0 and l_0. We combine the two modules i_0, j_0 and take them as one module. (2) Only one of $\hat{P}^0_{i_0,j_0} = \hat{P}^0_{i_0,k_0}$ and $\hat{P}^0_{i_0,j_0} = \hat{P}^0_{j_0,l_0}$ is accepted. Now the corresponding modules having the same connection probability are combined together. We look for the next large connection probability for these three modules, and test the relationship again. If the null hypothesis is accepted, the corresponding module is enrolled into this group, and the same step is implemented again. Otherwise, we combine the modules having the same connection probability together. (3) These modules are taken as on the same level. In this case, we search the next large connection probability to these four modules and do the statistical test until the hierarchical structure occurs or all the modules are combined together. After the above steps are finished, the connection probability between different modules is recalculated and recorded as $\hat{P}^1$. The above search and test steps are repeated for $\hat{P}^1$. Such steps are implemented recursively until all the modules are combined into one big module/network. For the statistical tests, we can also use t-test to test the relations between the connection probabilities if the distribution of the connections between different modules can be approximated by normal distribution.

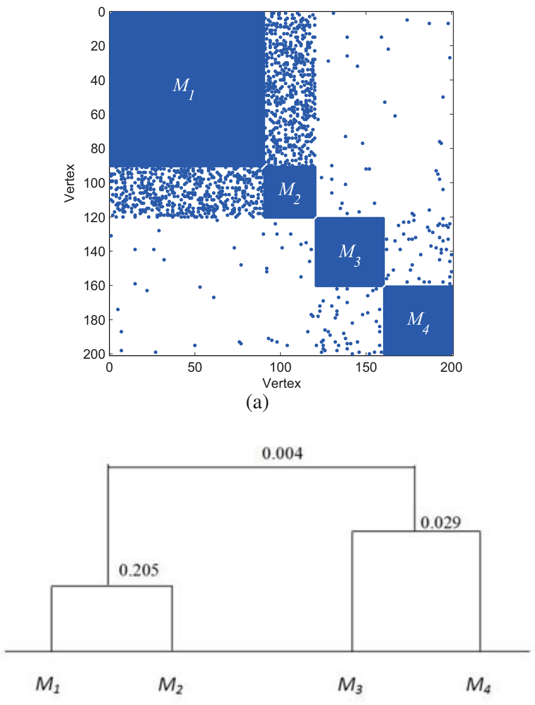
III. Numerical Experiments

In this section, we evaluate the performance of our proposed method through its application to several examples. We first start with two artificial networks having comparatively clear module structures. We then apply our method to two real networks to evaluate its performance. The first real network is the well-known karate club network and the second one is a yeast gene co-expression network.

A. Artificial Networks

1) A network composed of cliques: We consider a network with 200 nodes, which is composed of 4 cliques. The sizes of the cliques are 90, 30, 40, and 40. The connections between different cliques are randomly generated with the following

978-1-4673-4398-5/12/$31.00 ©2012 IEEE



(a)

(b)

Fig. 2. Example of hierarchial modular network structure.(a) Pattern of the adjacency matrix, (b) The hierarchical structure of the network

probability:

$$P = \begin{pmatrix} 1.000 & 0.200 & 0.002 & 0.003 \\ 0.200 & 1.000 & 0.005 & 0.010 \\ 0.002 & 0.005 & 1.000 & 0.030 \\ 0.003 & 0.010 & 0.030 & 1.000 \end{pmatrix}$$

The pattern of the adjacency matrix is shown in Fig. 2(a). From upper-left to lower-right, we denote the four modules as M_1, M_2, M_3, and M_4, which correspond to the position in the connection probability matrix. We can see the hierarchical structure of the network from the adjacency matrix. We apply our proposed method to this network. The condition (2) is satisfied until $K = 4$. The estimated connection probability matrix is:

$$\hat{P} = \begin{pmatrix} 1.000 & 0.205 & 0.003 & 0.003 \\ 0.205 & 1.000 & 0.006 & 0.009 \\ 0.003 & 0.006 & 1.000 & 0.029 \\ 0.003 & 0.009 & 0.029 & 1.000 \end{pmatrix}$$

We apply statistical tests to the corresponding modules, and finally we get the hierarchical structure as shown in Fig.2(b). The values on the hierarchial tree is the estimated connection probability of the corresponding modules. On the lowest level, there are four modules. If the tree is cut between 0.205 and 0.029, there are three module while if the cutoff is greater than 0.029, there are only two modules. These results are consistent with the network generation strategy.

2) A randomly generated network: In this example, we also consider a network with 200 nodes and 4 modules. The size of each module is 10, 45, 45, and 100. We set the degree of each node within its module to be 6, 15, 15, and 30. Then the connections between different nodes are randomly generated. We keep all the edges generated for each node. So finally the average degree within each module is greater than the pre-specified number. The connection probability between different modules is 0.002. The pattern of the adjacency matrix is shown in Fig. 3. From upper-left to lower-right, the four modules are M_1, M_2, M_3, and M_4, respectively. With our proposed method, the network is partitioned into four modules correctly on the lowest level and the estimated connection probability is:

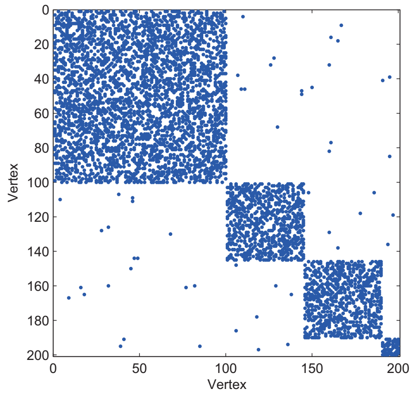


Fig. 3. Pattern of the adjacency matrix for the randomly generated network.

$$\hat{P} = \begin{pmatrix} 0.298 & 0.002 & 0.002 & 0.003 \\ 0.002 & 0.328 & 0.002 & 0.004 \\ 0.002 & 0.002 & 0.321 & 0.000 \\ 0.003 & 0.004 & 0.000 & 0.560 \end{pmatrix}$$

By using the statistical tests, these four modules are determined as parallel modules, which is consistent with our network generation strategy.

B. *Karate Club Network*

We consider the Zachary's network of karate club members [18] in this example. There are 34 nodes in this network corresponding to the members in a karate club. This dataset has been applied as a benchmark to test many module identification algorithms since the true modules are known in this network. The people in the club were observed for a period of three years. The edges represent connections of the individuals outside the activities of the club. At some point, the administrator and the instructor of the club broke up due to a conflict between them. The club was separated into two groups supporting the administrator and the instructor. The question is whether it is possible to infer the composition of the two groups from the original network structure recorded during the three years. Fig.4 shows the network. Originally, there are two modules, which have 16 nodes (squares and pentagons in the figure) and 18 nodes (circles and triangles in the figure), respectively.

We apply our proposed method to this network. The criterion (2) is satisfied until $K = 4$. The result is shown in Fig.4,

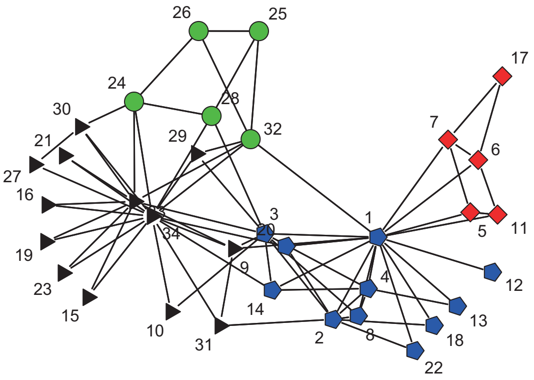


Fig. 4. Zacharys karate club network. Different shapes show the modules. M_1: pentagon, M_2: square, M_3: triangle, M_4: circle.

with different shapes of the nodes denoting different modules. The estimated connection probability matrix is:

$$\hat{P} = \begin{pmatrix} 0.364 & 0.073 & 0.056 & 0.036 \\ 0.073 & 0.480 & 0.000 & 0.000 \\ 0.056 & 0.000 & 0.237 & 0.108 \\ 0.036 & 0.000 & 0.108 & 0.480 \end{pmatrix}$$

From this matrix, it is easy to see that M_3 and M_4 are more likely to connect each other. With statistical tests, we can get that the connection probability among M_3, M_4, and M_1 is the same. Although M_2 has no connections to M_3 and M_4, it has a larger connection probability to M_1 than $M3, M_4$ to M_1. Thus these four modules are on the same level. In [25], the authors considered constructing the hierarchical modular structure of this network too. At first, they also found four modules on the lowest level. Then they found that this network has two modules with some nodes (3, 9, 10, 14, 31) belonging to both of them. We did not consider the overlapping nodes in this article. However, we can see that because these overlapping nodes belong to both M_1 and M_3, and they connect both parts closely, our method detect M_1 and M_3, M_3 and M_4 having the same connectivity.

C. Hierarchical Modular Structure in Yeast Gene Co-expression Network

In this section, we apply our proposed approach to analyze a gene co-expression network of yeast. The data set we use was generated by Brem and colleagues from a cross between two distinct isogenic strains BY and RM [28]. As described in [28], a total of 5740 ORFs were obtained after data preprocessing. In our analysis, we only use the 1,800 most differentially expressed genes as input to construct co-expression network and derive modules. When constructing the adjacency matrix of the network, we use the hard thresholding, that is: if the coefficient between two genes is greater than some given value, we assign an edge between them; otherwise, there is no edge. We compute the linear regression coefficient between the $\log 10$ transformed degree d ($\log 10(d)$) and the frequency of d ($\log 10(f(d))$), and choose the threshold that leads to approximately scale free property of the network as described in [30]. Finally, the threshold is set to be 0.705, $\hat{R}$ is about 0.75. By such a setting, this gene co-expression network is divided into 690 unconnected parts with the largest part having size 788. Here, we only analyze the hierarchical modular structure of the largest connected network.

Starting from $K = 2$, we apply our proposed method to this network, and the condition (2) holds until $K = 10$. To make the solution of partitioning the network into 10 modules more accurate, starting from the solution of our proposed approximation method, we do a global maximization changing the module index of boundary nodes. Since the approximate solution is already good, this step is very fast. The structure of the network is shown in 5(a), with different colors and shapes denoting different modules as described in Table I. Then we construct the hierarchical modular structure as shown in Fig. 5(b). On the lowest level, there are ten modules, while on the highest level, there are four modules.

Since co-expressed genes tend to be co-regulated and possibly have similar functions, genes in the same module are expected to be enriched for some function categories. In order to understand the biological basis of the network modules, we consider each identified module for enrichment of annotations from gene ontology (GO) [29]. In our analysis, the enrichment analysis was performed by GOstats from Bioconductor. For each module, the statistically most significant GO categories are analyzed. Table I shows the enrichment results for the ten modules. 'M-size' and 'G-size' are the size of both the modules and the GO categories, respectively. 'Overlap' is the overlap size of the module and the GO category. Table II shows the enrichment results for the modules on different levels. From the tables, it is easy to see that different gene function categories are enriched most on different levels. For example, module M_2 enriches the GO category "Translation" most significantly, while the combined module M_2, M_8 enriches "Ribonucleoprotein complex biogenesis" most significantly, with M_2 containing 42 genes having this function. The combined module M_2, M_8, M_4 and M_1 also enriches this function, while M_4 itself enriches "Cellular respiration" significantly. On the uppermost level, the module composed of M_2, M_8, M_1, M_4, M_3, and M_7 enriches four GO function categories most significantly, and all the genes are overlapped. Three ("Cellular component biogenesis", "Cellular component biogenesis at cellular level", and "Ribosome biogenesis") of them are different from the most enriched gene functions for each of these six modules. All these results indicate that hierarchial modular structure do exist in gene co-expression networks and different gene functions are enriched most on different levels.

We use the software REViGO to check the hierarchical structure of the enriched GO categories [32]. We considered the enriched GO categories in Table. I and Table. II except the category "Regulation of translational termination" because its G-size is very small and the p-value is comparatively large. Fig. 6 shows the tree map of the most enriched GO categories. Here the modules M_6, M_9 and other modules are parallel to each other, which is consistent with our results. M_3 and M_7

978-1-4673-4398-5/12/$31.00 ©2012 IEEE

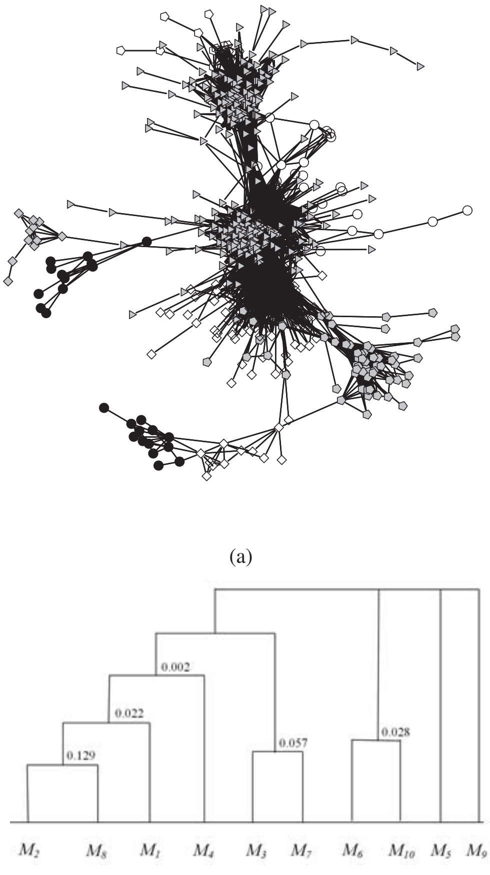


Fig. 5. Yeast gene co-expression network.(a) The network structure, (b) The hierarchical structure.

belong to a large category, which is "Branched chain family amino acid metabolic process". This large category is different from the most enriched category for the combined module M_3 and M_7. This may come from the fact that since M_7 is very small, it does not cover a large part of its enriched category. M_1 and M_4 are parallel to each other which is also consistent with our analysis. All these results show that our proposed method can explain some of the hierarchical structure of the GO categories. Due to the network size, we did not handle all the genes of yeast. This may be a reason why some of our computational results are not consistent with the GO function tree map.

IV. Conclusion

Module identification problem has attracted much attention from different fields and it continues being a hot research topic. How to determine the number of modules in a modular network has been an open problem during the study of module identification methods. Different identification methods may give different numbers. This problem may come from the hierarchical structure of modular networks. These different numbers correspond to the different levels of the hierarchial structure and they may be all reasonable. In this paper, we proposed a method for constructing the hierarchical modular structure of networks. With statistical tests, we can identify both the parallel modules and the hierarchical structure. According to different cutoffs of the hierarchical tree, different number of modules can be identified. This may solve the problem of the number of network modules to some extent. Several examples are given to demonstrate the efficiency of our method. Application of this method to gene co-expression networks shows that there are hierarchical modules in yeast gene co-expression network. On different levels of such networks, the genes in the module belong to different gene functions most. Thus studying the gene function through constructing the hierarchical modular structure instead of specifying the number of modules should perform better. Application of such algorithms to other kinds of networks may also contribute to other research fields.

Acknowledgment

This work was supported in part by NSFC grants 10901042, 10971075, and 91130032.

References

[1] R. Guimerà and L. A. N. Amaral, "Functional cartography of complex metabolic networks", Nature, 433, 895-900, 2005.

[2] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks", Proc. Natl. Acad. Sci. USA, 99, 7821-7826, 2002.

[3] A. Arenas, J. Borge-Holthoefer, S. Gómez, and G. Zamora, "Optimal map of the modular structure of complex networks", New J. Phys., 12, 053009, 2010.

[4] J. Dong and S. Horvath, "Understanding network concepts in modules", BMC Systems Biology, 1, 2007.

[5] E. Estrada and N. Hatano, "Communicability in complex networks", Physical Review E. 77, 036111, 2008.

[6] S. Fortunato, Community detection in graphs, Physics Reports, 486, 75-174, 2010.

[7] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices", Physical Review E, 74, 036104, 2006.

[8] M. E. J. Newman, "Modularity and community structure in networks", Proc. Natl. Acad. Sci. USA, 103, 8577-8582, 2006.

[9] M. A. Porter, *et al.*, "Communities in networks", Notices of the AMS 56, 1082-1102, 2010.

[10] F. Radicchi, *et al.*, "Defining and identifying communities in networks", Proc. Natl. Acad. Sci. USA, 101, 2658-2663, 2004.

[11] Z. Li, S. Zhang, R. S. Wang, X. S. Zhang, and L. Chen, "Quantitative function for community detection", Physical Review E, 77, 036109, 2008.

[12] M. Rosvall and C. T. Bergstrom, "An information-theoretic framework for resolving community structure in complex networks", Proc. Natl. Acad. Sci. USA, 104, 7327-7331, 2007.

[13] P. Bickel and A. Chen, "A Nonparametric View of Network Models and Newman-Girvan and Other Modularities", Proc. Natl. Acad. Sci. USA, 106, 21068-21073, 2009.

[14] M. E. J. Newman, "Fast algorithm for detecting community structure in networks", Physical Review E, 69, 066133, 2004.

[15] S. Fortunato and M. Barthélemy, "Resolution limit in community detection", Proc. Natl. Ac. Sci. USA, 104, 36-41, 2007.

[16] L. Danon, *et al.*, "Comparing community structure identification", Journal of Statistical Mechanics: Theory and Experiment, P09008, 2005.

[17] A. Arenas, A. Díaz-Guilera, and C. J. Pérez-Vicente, "Synchronization reveals topological scales in complex networks", Phys. Rev. Lett., 96, 114102, 2006.

[18] W. W. Zachary, "An information flow model for conflict and fission in small groups", J. Anthropol. Res., 33, 452-473, 1977.

[19] B. H. Good, Y.-A. de Montjoye, and A. Clauset, "The performance of modularity maximization in practical contexts", Physical Review E, 81, 046106, 2010.

978-1-4673-4398-5/12/$31.00 ©2012 IEEE

TABLE I
GO ENRICHMENT ANALYSIS RESULTS OF THE GENE MODULES ON THE LOWEST LEVEL

Module	Color, shape	M-size	Enriched GO category	p-value	G-size	Overlap
M_1	white, square	190	Cellular carbohydrate metabolic process	3.23×10^{-9}	60	35
M_2	white, circle	126	Translation	4.70×10^{-59}	101	80
M_3	grey, triangle	135	Organic acid biosynthetic process	5.41×10^{-35}	89	64
M_4	grey, pentagon	62	Cellular respiration	4.13×10^{-27}	36	28
M_5	black, circle	12	Amino acid catabolic process to alcohol via Ehrlich pathway	1.76×10^{-7}	5	4
M_6	black, circle	13	Steroid biosynthetic process	2.20×10^{-15}	13	9
M_7	white, pentagon	19	Branched chain family amino acid metabolic process	4.37×10^{-8}	11	6
M_8	grey, triangle	209	Ribonucleoprotein complex biogenesis	5.94×10^{-39}	149	106
M_9	grey, square	11	Protein targeting to membrane	8.91×10^{-6}	4	3
M_{10}	white, square	11	Regulation of translational termination	1.55×10^{-4}	2	2

TABLE II
GO ENRICHMENT ANALYSIS RESULTS OF GENE MODULES ON THE UPPER LEVEL

Module	M-size	Enriched GO category	p-value	G-size	Overlap
M_2, M_8	335	Ribonucleoprotein complex biogenesis	4.02×10^{-66}	149	148
M_2, M_8, M_1	525	Ribonucleoprotein complex biogenesis	1.33×10^{-29}	149	148
M_2, M_8, M_1, M_4	587	Ribonucleoprotein complex biogenesis	6.04×10^{-23}	149	149
M_3, M_7	154	Organic acid biosynthetic process	9.22×10^{-40}	89	71
$M_2, M_8, M_1, M_4, M_3, M_7$	741	Cellular component biogenesis	4.01×10^{-6}	175	175
		Cellular component biogenesis at cellular level	1.84×10^{-5}	156	156
		Ribonucleoprotein complex biogenesis	3.19×10^{-5}	149	149
		Ribosome biogenesis	3.44×10^{-5}	148	148
M_6, M_{10}	24	Steroid biosynthetic process	2.36×10^{-19}	13	12

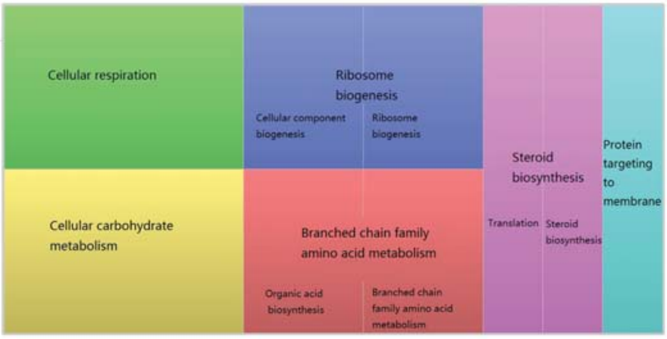


Fig. 6. Tree map of the enriched GO categories in yeast gene coexpression network.

[20] A. Khadivi, A. A. Rad, and M. Hasler, "Network community-detection enhancement by proper weighting", Physical Review E, 83, 046104, 2011.

[21] T. Richardson, P. J. Mucha, and M. A. Porter, "Spectral tripartitioning of network", Physical Review E, 80, 036111, 2009.

[22] J. Ruan and W. Zhang, "Identifying network communities with a high resolution", Physical Review E, 77, 016104, 2008.

[23] D. Meunier, R. Lambiotte, and E. T. Bullmore3, "Modular and hierarchically modular organization of brain networks", Front. Neurosci., 4(200), doi: 10.3389/fnins.2010.00200, 2010.

[24] H. Shen, X. Cheng, K. Cai, and M. Hua, "Detect overlapping and hierarchical community structure in networks", Physica A, 388, 1706-1712, 2009.

[25] A. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the overlapping and hierarchical community structure in complex networks", New Journal of Physics, 11, 033015, 2009.

[26] E. Ravasz, "Detecting Hierarchical Modularity in Biological Networks", Computational Systems Biology, 54, 145-160, 2009.

[27] P. Erdős, and A. Rényi, "On Random Graphs. I.", Publicationes Mathematicae, 6, 290-297, 1959.

[28] R. B. Brem and L. Kruglyak, "The landscape of genetic complexity across 5,700 gene expression traits in yeast", Proc. Natl. Acad. Sci. USA, 102, 1572-1577, 2005.

[29] M. Ashburner, *et al.*, "Gene ontology: tool for unification of biology, the gene ontology consortium", Nature Genetics, 25, 25-29, 2000.

[30] B. Zhang and S. Horvath, "A general framework for weighted gene co-expression network analysis", Stat. Appl. Gen. Mol. Biol., 4(1), Article 17, 2005.

[31] S. Zhang and H. Zhao, "Community identification in networks with unbalanced community structure", Physical Review E, 85, 066114, 2012.

[32] F. SupekF, M. Bošnjak, N. Škunca, T. Šmuc, "REVIGO Summarizes and Visualizes Long Lists of Gene Ontology Terms", PLoS ONE, 6(7): e21800. doi:10.1371/journal.pone.0021800, 2011.

978-1-4673-4398-5/12/$31.00 ©2012 IEEE