BIOINFORMATICS ORIGINAL PAPER

Vol. 21 no. 11 2005, pages 2739–2747
doi:10.1093/bioinformatics/bti406

Systems biology

Modularized learning of genetic interaction networks from biological annotations and mRNA expression data

Phil Hyoun Lee[1] and Doheon Lee[2,*]

[1]School of Computing, Queen's University, Canada and [2]Department of BioSystems, KAIST, Korea

Received on November 12, 2004; revised on March 2, 2005; accepted on March 22, 2005
Advance Access publication March 29, 2005

ABSTRACT

Motivation: Inferring the genetic interaction mechanism using Bayesian networks has recently drawn increasing attention due to its well-established theoretical foundation and statistical robustness. However, the relative insufficiency of experiments with respect to the number of genes leads to many false positive inferences.

Results: We propose a novel method to infer genetic networks by alleviating the shortage of available mRNA expression data with prior knowledge. We call the proposed method 'modularized network learning' (MONET). Firstly, the proposed method divides a whole gene set to overlapped modules considering biological annotations and expression data together. Secondly, it infers a Bayesian network for each module, and integrates the learned subnetworks to a global network. An algorithm that measures a similarity between genes based on hierarchy, specificity and multiplicity of biological annotations is presented. The proposed method draws a global picture of inter-module relationships as well as a detailed look of intra-module interactions. We applied the proposed method to analyze *Saccharomyces cerevisiae* stress data, and found several hypotheses to suggest putative functions of unclassified genes. We also compared the proposed method with a whole-set-based approach and two expression-based clustering approaches.

Availability: JAVA programs for the MONET framework are available from the corresponding author upon request. Web supplementary data is accessible at http://biosoft.kaist.ac.kr/∼dhlee/monet/index.html

Contact: doheon@kaist.ac.kr

1 INTRODUCTION

Recently, learning genetic interaction networks from mRNA expression data has successfully shown its potential to uncover cellular mechanisms in a cell (Liang *et al*., 1998; Friedman *et al*., 2000; Akutsu *et al*., 2000; Tamada *et al*., 2003; Segal *et al*., 2003c). Among several computational formalisms, such as Boolean networks and qualitative networks, Bayesian networks (Neapolitan, 2004) have drawn increasing attention due to their well-established theoretical foundation and statistical robustness (Friedman *et al*., 2000; Peer *et al*., 2001; Yoo *et al*., 2002; Hartemink *et al*., 2002; Tamada *et al*., 2003).

Learning Bayesian networks can be regarded as an inference of relationships between nodes (i.e. genes) from observational mRNA expression data. It is known that sufficiently large amounts of expression profiles are required to infer statistically reliable relationships among nodes (Neapolitan, 2004). However, it is hard or nearly impossible to secure such sufficient amounts of expression profiles when hundreds or thousands of genes are considered. This shortage of observation data leads to many false positive edges; a significant portion of inferred relationships is not consistent with known biological knowledge. To alleviate this problem, several techniques incorporating statistical biases and prior biological knowledge have been proposed.

Friedman *et al*. (2000) have introduced two statistical techniques, sparse candidates (Friedman *et al*., 1999) and model averaging. The former restricts the maximum number of affecting genes for each target gene so that the search space is reduced. The latter generates multiple networks from different initial conditions, and extracts commonly inferred edges. Other groups have incorporated prior biological knowledge to refine network structures. Hartemink *et al*. (2002) have applied the chromatin immuno-precipitation (CHIP) assay and Tamada *et al*. (2003) incorporated promoter sequence motif information as prior knowledge. They both assumed that relationships between transcription factor genes and their target genes should be supported by other biological clues. Recently, modularization approaches have been introduced by several groups (Fashing *et al*., 2002; Segal *et al*., 2003a,b). They used clustering methods to divide a gene set into smaller groups, and applied network learning over each module.

In this paper, we propose a new method for inferring modularized gene networks by utilizing two complementary sources of information: biological annotations and gene expression. First, *seed genes*, which respond very distinctively in a specific experimental condition, are identified. Secondly, the closely related genes with the *seed genes* based on biological annotations and expression data are grouped into overlapped modules. After the identification of modules, the proposed method infers a Bayesian network for each module and integrates them through common *intermediary genes*. The outline of the proposed method is depicted in Figure 1. Our method is based on the assumption that a cellular system is composed of locally interacting biological modules; most of the genes are likely to be related to the genes in the same biological modules rather than the genes in different modules (Calabretta *et al*., 1998; Hallinan, 2004). Therefore, a divide-and-conquer approach not only enables independent construction of subnetworks, but also improves learning due to the increased ratio of the number of experiments to the number of genes. The proposed method also assumes that many genetic relationships will share the same biological annotations or show related mRNA expression patterns. Recently, Tong *et al*. (2004) reported that genetic relationships frequently coincide with known functional

[*]To whom correspondence should be addressed.

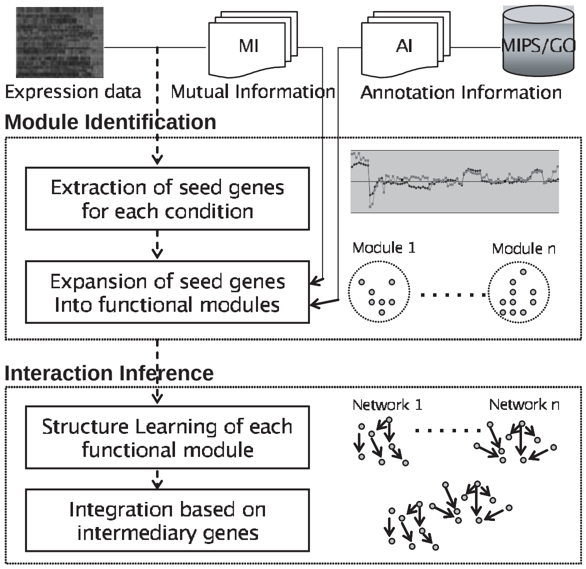
Expression data
MI
Mutual Information
AI
MIPS/GO
Annotation Information
Module Identification
Extraction of seed genes for each condition
Module 1
Module n
Expansion of seed genes Into functional modules
Interaction Inference
Network 1
Network n
Structure Learning of each functional module
Integration based on intermediary genes

Fig. 1. Overview of modularized network learning. The procedure is composed of two main parts: (1) The module identification part decomposes a whole gene set into overlapped modules of genes. (2) The interaction inference part infers relationships between genes using a Bayesian network algorithm, and integrates the learned networks.

relationships between gene pairs; over 12% of genetic interactions are comprised of genes with an identical GO (GO Consortium, 2001) annotation (12 times more than expected by chance), and over 27% of genetic interactions are between genes with similar or identical GO annotations in a very conservative estimate.

Basically, the proposed method concurs with other modularization approaches (Segal *et al*., 2003a,b; Fashing *et al*., 2002) in that explicit modularization would reduce the incorrect dependencies caused by the high dimensionality of data. However, our method has several unique aspects. First, we adopt overlapped modularization rather than partitioning since there are genes which participate in multiple cellular processes, or function as an inter-connector between different processes. These overlapped genes are called *intermediary genes*, and they function as integrators for combining separately learned subnetworks from each module into global networks. Secondly, by identifying related modules via *intermediary genes*, the proposed method presents genome-wide inter-module relationships as well as detailed intra-module relationships. Thirdly, it complements the limitation of expression data by incorporating biological annotations, which are less noisy and more reliable than high-throughput data. Nevertheless, biological annotations are only used for identification of modules but not for inference of interactions in a network. Therefore, their use reduces the chance of fallacious inferences but does not interfere in the learning process itself. Lastly, the proposed method identifies modules using two independent sources of information (i.e. biological annotations and expression data) in a union-set way. Even though established knowledge does not support the relationship between genes, they can be grouped together if patterns of gene expression strongly indicate it, and vice versa. This union-set way rather than a joint-set way complements the limitation of either information by the other, and opens the chance to discover yet-unknown but highly plausible hypotheses for further research.

We applied the proposed method to analyze *Saccharomyces cerevisiae* stress data (Gash *et al*., 2000). It has been shown that our method not only infers interactions in accordance to established biological knowledge, but also presents interesting new hypotheses about the function of currently unclassified genes. In addition, we compared our method with a whole-set-based approach (i.e. inference of Bayesian networks over a set of genes as a whole) and two expression-based clustering approaches to evaluate the advantage of the proposed method.

2 METHODS AND ALGORITHM

2.1 Module identification

The first part of the proposed method is module identification; it identifies overlapped gene modules consistent with localized cellular processes. Beginning with active *seed genes*, related genes based on two independent sources of information, biological annotations and mRNA expression data are grouped together.

2.1.1 Extraction of seed genes Since expression data measures the changed pattern of mRNA abundance responding to experimental conditions, cellular activities closely related to a given external stimulus tend to show a wide sphere of action. Here, we concentrate on reconstructing those active biological processes during the given experimental conditions. First, we define *seed genes* as a set of genes which show significantly higher or lower expression levels in one condition than in all the others. For example, *S.cerevisiae* stress data from Gash *et al*. (2000) consist of 173 experiments consecutively measured in 16 different stress conditions; every stress condition consists of several experiments (refer Fig. 4 for details). *Distinctiveness D* of a gene i in one condition c is based on Sharmir's measure (Shamir, 2002) and defined as follows:

$$D(gene_i, condition_c) = \frac{|\mu_{ci} - \mu_{\neg ci}|}{\sigma_{ci} + \sigma_{\neg ci}}$$

μ_{ci} is the mean expression value of gene *i* during experiments belonging to the same condition *c*, while $\mu_{\neg ci}$ is the mean expression value of gene *i* during experiments not belonging to a condition c. σ_{ci} and $\sigma_{\neg ci}$ are the standard deviations corresponding to the former and the latter cases, respectively. Intuitively, a large difference between μ_{ci} and $\mu_{\neg ci}$ indicates that gene i shows a distinctive expression pattern in a condition c compared to all the other conditions. The smaller σ_{ci} and $\sigma_{\neg ci}$ are, the more consistent the expression pattern of a gene i in both cases. Those genes whose *Distinctiveness D* is greater than a threshold are extracted as *seed genes*. Here, we use the relative threshold value based on the distribution of *Distinctiveness D* of all genes in a dataset: $\mu_D + 3 * \sigma_D$. This was empirically chosen to restrict the number of *seed genes* to ~5% of the total genes considered.

2.1.2 Utilization of functional annotations To identify genes involved in the same cellular processes as *seed genes*, we utilize biological annotations such as MIPS (Mewes *et al*., 1997) or GO (GO Consortium, 2001). This prior knowledge provides us with reliable explanations about the biological roles of genes, but they have unique characteristics which should be reflected properly. First, biological annotations have a hierarchical structure. Even though two annotations are different, they can be closely related via common ancestors. Secondly, multiple annotations are allowed for a single gene. Therefore, we have to consider not only whether two genes share the same annotation, but also how many annotations they share. Lastly, biological annotations have different specificities. For example, while a GO term, GO0006414 (translational elongation), annotates 309 yeast genes, another GO term, GO0006448 (regulation of translational elongation), annotates only three yeast genes. Therefore, in the context of biological annotations, the degree of gene similarities not only depends on the number of shared annotations, but also depends on the specificity of them. Lord *et al*. (2003) showed that a semantic tree can enable us to calculate the similarity of two biological

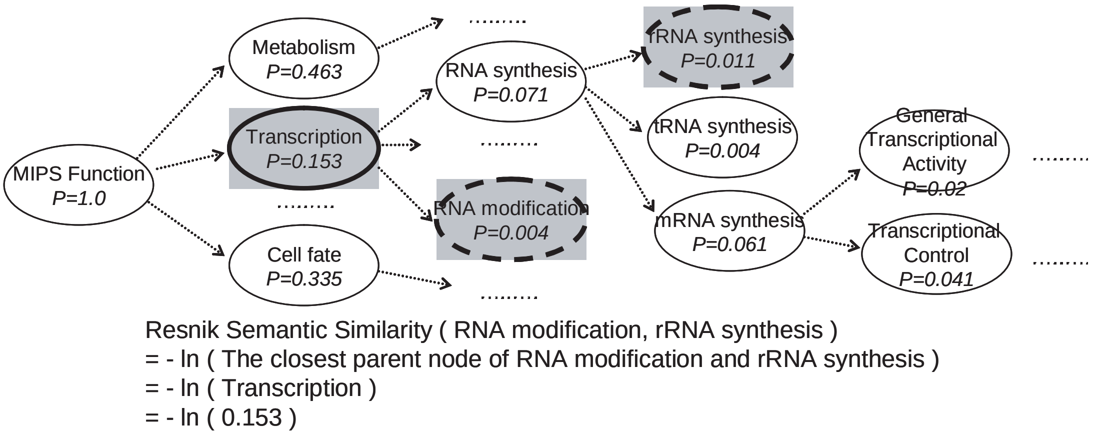

Fig. 2. A semantic tree is built from the annotation terms in MIPS (Mewes *et al*., 1997). Each node in a tree represents an annotation term in MIPS, and the P value in it indicates the *Information Content* of the annotation, which calculates how many genes each node, or any of its children, annotate as a percentage. To measure the semantic similarity between two annotation terms, the Resnik similarity measure is used. As an example, calculation of the semantic similarity of two terms, RNA modification and rRNA synthesis, which are designated with a dotted line with a gray color is shown in the figure. The *Information Content* of the closest common parent of two terms, *Transcription*, which is marked with a solid line with a gray color is used in the equation.

annotations based on their hierarchy and specificity. We adopt this concept to identify the similarity of two genes.

The *Annotation Information* (*AI*) score of two genes is defined as a similarity measure of them in the context of biological annotations[1]. First, we build a semantic tree K from biological annotations (Lord *et al*., 2003). Each node in a semantic tree corresponds to an annotation term in source biological annotations, and it contains an *Information Content* value P, which indicates how many genes each node, or any of its children, annotates as a percentage. The *Similarity* score S of two annotation terms f_i and f_j in a semantic tree K is calculated by Resnik Measure (Resnik, 1999) as follows:

$$S(f_i, f_j) = -\log(\textit{Information Content}\, P \text{ of the closest parent of } f_i \text{ and } f_j \text{ in a semantic tree } K)$$

Figure 2 describes an example of the *Similarity* score calculation, and its detailed algorithm can be found in (Lord *et al*., 2003). The *AI* score of two genes g_i and g_j is defined based on the *Similarity* score *S* of their annotation terms:

$$\begin{aligned} & AI(g_i, g_j) \\ & \quad = \sum_{f_k \in (AT(g_i) \bigcap AT(g_j))} S(f_k, f_k) \\ & \quad + \max_{(f_i \in (AT(g_i) \bigcap AT^c(g_j))) \bigcap (f_j \in (AT^c(g_i) \bigcap AT(g_j)))} S(f_i, f_j) \end{aligned}$$

$AT(g_i)$: a set of annotation terms for a gene i

$AT(g_j)$: a set of annotation terms for a gene j

Annotation terms in $AT(g_i)$ and $AT(g_j)$ can be divided into two categories: terms present in both sets or not. If two genes share the same annotation terms, the *Similarity* score of those terms are accumulated. This is based on the assumption that if two genes share multiple annotations, they are considered more similar than a pair of genes which share a smaller part of those annotations. For the annotation terms belonging to only one set, the maximum *Similarity S* of all combination of annotation pairs is added to the *AI* score. This is to prevent the *AI* score from being increased due to a large number of annotation terms some genes have, not due to their real similarity. Current *AI* scores based on the MIPS (Mewes *et al*., 1997) semantic tree have a maximum value of 31.55 and minimum value of 4.9e−324.

2.1.3 Utilization of the mRNA expression data

To find genes that participate in the same cellular activities as *seed genes* but not annotated yet, we use *Mutual Information* (Kohane *et al*., 2003). *(MI)* of mRNA expression data. *Mutual information* indicates how much information one random variable tells about another. Therefore, the *MI* score of two expression profiles represents the degree of dependency between two genes based on their mRNA expression patterns. In the extreme case, if expression patterns of two genes are completely independent, their *MI* score will be zero. An *MI* score of two genes, g_i and g_j is defined as follows.

$$MI(g_i, g_j) = \sum_{x_i} \sum_{x_j} p(x_i, x_j) \log \frac{p(x_i, x_j)}{p(x_i)p(x_j)}$$

x_i : a discretized expression value of a gene g_i

x_j : a discretized expression value of a gene g_j

2.1.4 Expansion of seed genes into modules

Selected *seed genes* are expanded into modules by including closely associated genes based on *AI* and *MI* scores. Basically, one *seed gene* is an initiating point to grow into a single module. However, if more than one seed gene are close enough to each other based on the *AI* and *MI* threshold values, they are merged into a single module to avoid having multiple modules with almost the same members in them. The *AI* and *MI* threshold values for picking up closely associated genes were determined empirically: $\mu_{AI} + x_{AI} \cdot \sigma_{AI}$ and $\mu_{MI} + x_{MI} \cdot \sigma_{MI}$. ($\mu_{AI}$ is the mean value of all *AI* scores of the yeast gene pairs downloaded from SGD (Cherry *et al*., 1998) and σ_{AI} is the standard deviation of them. μ_{MI} is the mean value of all *MI* scores of the yeast gene pairs in a dataset after preprocessing. σ_{MI} corresponds to the standard deviation of them.) Since each dataset has different target genes after preprocessing, the distribution of total *MI* scores will depend on the gene list in the dataset. For each possible value of x_{AI} and x_{MI} (here, integers from 3 to 5), we examined the resulting modules based on three factors, total number of modules, the average module size and coverage. Coverage is defined as the total number of genes in all identified modules. We prefer threshold values which lead to modules with high coverage and small average module sizes.

$$\textit{Modularization score} = \frac{\textit{Coverage}}{\textit{Average Module Size}}$$

[1] We used MIPS (Mewes *et al*., 1997) as source biological annotations for module identification, and used GO (GO Consortium, 2001) to validate the consistency of the final networks.

Table 1. Comparison of the Interaction Inference results from Modularization with different thresholds

Case number	Threshold (x_{MI}, x_{AI})	Modularization score	Gene number	Edge number	Overlapped/non-overlapped edge number	Consistent/inconsistent edge number
I	(3,4)	21.21	1612	328	229 (69.8%)/99 (30.2%)	71 (21.6%)/257 (69.8%)
II	(4,4)	18.31	879	338	240 (71.0%)/98 (29.0%)	82 (24.3%)/235 (75.7%)
III	(5,5)	12.61	429	147	64 (43.5%)/83 (56.5%)	35 (23.8%)/103 (70.1%)
IV	(5,3)	12.02	1290	480	90 (18.8%)/390 (81.2%)	81 (16.9%)/325 (67.7%)

For an *S.cerevisiae* stress dataset (Gash *et al*., 2000), threshold values, $\mu_{AI} + 4 \cdot \sigma_{AI}$ and $\mu_{MI} + 3 \cdot \sigma_{MI}$ were chosen to have the highest *Modularization score*. Table 1 shows the effect of different *AI* and *MI* threshold values, and the Results section discusses its implication. Detailed module information including *seed genes* and module members can be found in the web supplementary data.

2.2 Network learning

2.2.1 Learning of subnetworks for individual modules To learn Bayesian networks for individual modules, we apply a Bayesian network learning technique as in Friedman *et al*. (2000) and Hartemink *et al*. (2002), which is based on hill climbing, sparse candidates (Friedman *et al*., 1999) and model averaging. Beginning with randomly generated initial networks, a hill climbing algorithm with random restart is used to search the best matching network structures for a given data. We use the MDL (minimum description length) (Lam and Bacchus, 1994) score as an evaluation function for a network structure. With *N* (here 100) best candidate networks, a final network is built by selecting confident edges based on a ratio of occurrences and a score of a network. The *Confidence* score of an edge ($edge_i$) in *N* candidate networks is defined as below:

$$Confidence\ (edge_i) = \frac{\sum_{n_k \in S \wedge edge_i \in n_k} Score(n_k)}{\sum_{n_j \in S} Score(n_j)}$$

$$S = \text{ a set of } N \text{ best networks}$$

Edges whose confidence is >0.75 compose a base framework of the final network and edges whose confidence is between 0.75 and 0.5 are appended to it if either end of them is already residing in a base framework. Final networks learned from each module are called *subnetworks* since they become parts of global networks after being integrated with other *subnetworks*.

2.2.2 Integration of subnetworks via intermediaries Integration of *subnetworks* is done by combining *subnetworks* which share common genes between them. Recall that genes can belong to multiple modules (i.e. overlapped modularization) if they show acceptable *AI* and *MI* scores with respect to the *seed genes* in different modules. We call those genes belonging to multiple modules as *intermediary genes*. These genes play a role of intermediaries among *subnetworks* in the sense that they may intermediate different cellular processes or suggest related modules. The integration algorithm is described in Figure 3.

```
mark all edges in subnetworks as non-connected;
network_index=-1;

do {
      org_edge = the first non-connected edge;
      mark an org_edge as connected;
      network_index++;
      put an org_edge in a global_network[network_index];

       do {
         for( each subnetwork ) {
             for( each non-connected edge ) {
                 if( edge is connected to edges
                          in a global_network[network_index]) {
                    mark it as connected;
                     put it in a global_network[network_index];
                  }
             }
      } while(no more edge is connected);

} while( no edge is non-connected );
```

Fig. 3. The integration algorithm of subnetworks via intermediary genes.

3 RESULTS

3.1 Data

The proposed method was applied to *S.cerevisiae* stress data (Gash *et al*., 2000). In this dataset, a total of 173 expression values of each gene are measured on 16 different stress conditions in a time-series manner. After preprocessing,[2] 4931 genes remain to be studied. Among 16 experimental conditions, we focus on 12 well-studied stress conditions including heat shock stress, oxidative stress, reductive stress, osmotic stress, starvation and diauxic shift.

3.2 Biological analysis

3.2.1 Extracted seed genes and expanded modules Since external stress causes cellular perturbations, we expected that the identified *seed genes* and their expanded modules would include the cellular processes mainly related to protecting and maintaining the internal balance of a cell under the corresponding stress condition. Characteristics of *seed genes* and their corresponding stress conditions are summarized in Figure 4. It is clear that many annotations of *seed genes* imply the biological processes closely related to the corresponding stress conditions. Annotations involving in C-compound, transported routes and cell death appear in several places regardless of the stress type. We assume that those cellular processes function ubiquitously during various stress conditions.

The genes expanded from *seed genes* based on their *MI* score give us clues about the function of unannotated genes. For example, YLR217W was extracted as a *seed gene* during a heat shock condition type-I. Its expanded genes include CPR6, HSP10, HSP60, HSP82, HSC82 and STI1. All of them carry out very similar biological functions, i.e. chaperone. Even though a *seed gene*, YLR217W is a currently unannotated ORF; its related genes based on mRNA expression patterns and its distinctive stress condition

[2] Imputing missing data using Norm2.02 (Schafer, 1997), 2-fold variation filtering, and smoothing with adjusting window size as 3 (Kwon *et al*., 2003) is done.

MIPS Annotation Level I	MIPS Annotation Level II	Heat Shock		Oxidative Stress			Reductive Stress		Osmotic Stress		Starvation		Diauxic shift
		1	2	3	4	5	6	7	8	9	10	11	12
Metabolism	Amino acid			■	■▨						■▨		
	Nitrogen, sulfur										■▨	■	
	Nucleotide										■	■▨	
	C-compound			■▨		■	■▨	■	■	■▨	■▨		■▨
	Lipid, fatty acid			■▨	■▨			■	■	■▨			
	Vitamins,			■▨									
Energy	Fermentation			■▨		■▨							
Cell cycle	Cell cycle		■▨		■▨	■▨	▨		■				▨
	DNA process			■▨		■▨			■	■▨			
Transcription	RNA synthesis			▨	■▨		▨		■▨	■	■	■	▨
	RNA process						■▨	▨	■▨	■▨			
Protein fate	Degradation					■			■▨				
	Folding	■			▨								▨
	Modification						■▨	■	■				
	Targeting							■	■▨				
Synthesis	Ribosome				■▨								
Cellular transport	Route		■		■▨	■	■	■▨	■▨	■	■▨	■▨	■▨
	Facilitation						▨						
	Compounds		■		■▨					■▨	■▨	■▨	■▨
Cell rescue	Stress Response	■		■	■▨	■			■				
	Detoxification			■	■▨	■							■▨
Interaction	Homeostasis				■▨					■	■		
	Sensing				▨				■	■▨			
Cell fate	Growth				▨				■				▨
	Differentiation				▨				■	■	■▨	■▨	
	Death			■▨	■▨	■▨	■▨	■▨	■▨	■	■	■	▨
Component	Mitochondria			■	■▨								
Differentiation	Fungal				▨		■▨		■				▨

Fig. 4. Biological categories of the extracted seed genes. Biological annotations which coincide with the identified seed genes more than three times are summarized. If a seed gene shows the reduced expression in the identified condition compared with all the other conditions, it is marked with a gray color. The opposite case is marked with a black color. Each stress condition [heat shock *stress*$_1$(experiments 1–9), heat shock *stress*$_2$ (10–15), hydrogen peroxide stress (36–45), superoxide generating drug menadione stress (46–54), diamide stress (70–77), dithiothreitol *stress*$_1$(55–62), dithiothreitol *stress*$_2$ (63–69), hyper-osmotic stress (78–84), hypo-osmotic stress (78–90), amino acid starvation stress (91–95), nitrogen depletion stress (96–105) and diauxic shift (106–112)] are numbered from 1 to 12 consecutively.

(i.e. heat shock) suggest that its function may be related to a chaperone activity. We found additional evidence which supports this hypothesis. In the experiment done by Travers *et al*. (2000), YLR217W is constantly over-expressed during the unfolded protein response, which implies its increased necessity for the unfolded protein response. Moreover, among 14 genes which show similar expression with YLR217W, about half of the genes with known functions are related to ‘chaperone’ based on GO (GO Consortium, 2001) annotation: APJ1, STI1, TFS1. In addition, we have found protein interaction data which shows YLR217W interacts indirectly with co-chaperone activator (YOR349W) via a domain protein YMR294W (http://biodata.mshri.on.ca/yeast_grid/).

Some expanded ORFs are revealed to be located next to the corresponding *seed genes* on the chromosome: CPR6(YLR216C) and YLR217W, YOL150C and GRE2(YOL151W) and YOR225W and ISU2(YOR226C). Since the expression profiles of two genes in each pair are very similar to each other, we presume that they were originally a single gene or constitute a protein complex.

3.2.2 Learned networks The Interaction Inference step resulted in one big connected network and many small networks. Among them, the largest connected network is depicted in Figure 5. Several distinctive cellular activities are identified, and a few of them are summarized below.

Networks directly protecting internal systems responding to external stresses are mainly reconstructed. A representative case is an oxidative stress. Genes related to the peroxisome organization and biogenesis (OAF2, CAT3, CIT2, IDP3, AAT2), exporting

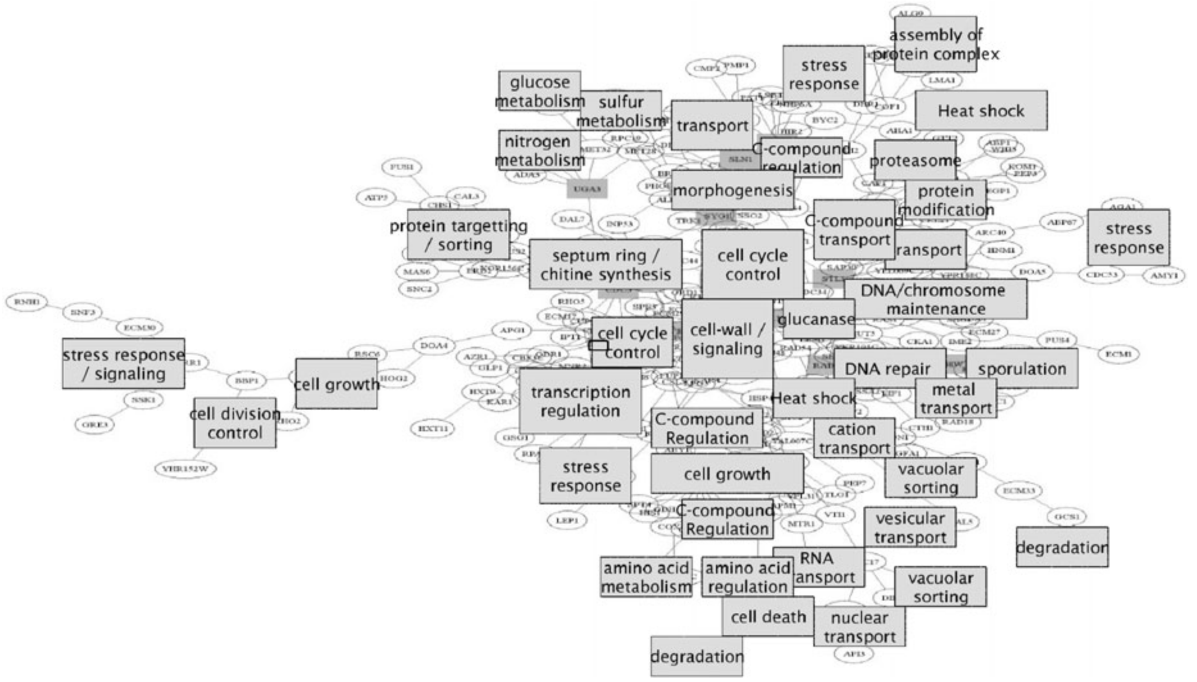

Fig. 5. The largest connected network by integrating subnetworks via common intermediaries. Characteristic nodes are marked with the corresponding biological annotations.

metabolites of the peroxisome (CAT2), a regulator of cell redox homeostasis (cTPxII), glutathione transferase (GTT2), superxoide dismutase copper chaperone (CCS) and sulfite reductase (ECM17) are found in this part. Another representative example is a heat shock stress. Subnetworks of protein folding and degradation include: chaperone genes (STI1, EGD1, EGD2, STT10, BIP, CPR7, ATX1), chaperone regulator (HSP40), chaperone activator(AHA1), protein folding and stabilization genes (HSP30, ABP1, ACT2, WHI4, SAP30, PPH3, EXG2, SSA2, BMH2, CLC1, LPI6, CTH1, BYC2, SWH48, CMP1, APR6, CMK1, YGL190C, TPM1), protein fate genes (CLS8, CCS), and ubiquitin protein complexes and their targeting genes (UBC4, DOA3, DOA4, DOA5, SCD2, CBF3D, CDC53, MDP1, RAD18).

Networks common in various stress conditions are also found. The typical examples include C-compound and carbohydrate utilization, respiration, transport and cell wall organization. Energy sources such as glycogen and trehalose are well known to play a critical role in response to various stress conditions (Francois *et al*., 2001; Hohmann and Mager, 2003). Since stress defense mechanisms consume a significant amount of ATPs, respiration components(ATP5, AEP2, IMG1) and aerobic respiration genes(SMP2, COX11, YDR115W) are also induced in various stresses. In stressful conditions, many genes should be transcribed and moved to the cytosol to respond to rapidly changing external conditions. Therefore, *subnetworks* involved in RNA transcription, splicing, import and export are found in many places as well. Also, stressful conditions demand the cell wall and cytoskeleton structures in a cell to reorganize in order to adapt in a changed environment.

By examining edges in final networks, new hypotheses or complementary evidences about the functions of currently unclassified genes were presented. YBL010C is a currently un-annotated ORF, but its connected pairs in a network, ECM10, PIM1, PUP1 suggest that it may be involved in protein stabilization or degradation under a heat shock stress. We have found two other evidences to support this hypothesis. First, Middendorf *et al*. (2004) presented the decision rule learning to predict gene regulatory responses using motifs and expression levels. Under USV1 knockout (heat shock and osmolarity stress), YBL010C is selected as a state-changing target gene, which implies its involvement in the given stress condition. Secondly, the protein product of YBL010C is known to interact with SPP382p, a suppressor of the temperature-sensitive growth defect (Hazbun *et al*., 2003).

YBL055C is also an un-annotated ORF. Its inferred pair, NUP157, is involved in nuclear transporter activity. YBL055C has a physical interaction with a nucleotide exchange factor, PRP20 (Bader *et al*., 2003). These suggest YBL055C also has a function in transporter activity. Another un-annotated ORF, YHR207C, is assumed to be involved in drug responses based on its inferred pair, FLR1, a multidrug transporter. Physical interaction data of YHR207C with RIO2, a nucleocytoplasmic transporter, indirectly supports this hypothesis (http://contact14.ics.uci.edu/pgo/). In other research (Mendizabal *et al*., 1998), RIO2 is identified as one of the halo tolerance genes and it contains two C2H2 zinc finger motifs. Based on the other transcription factors with same zinc finger motifs, the authors suggested the function of RIO2 as a transcriptional controller of genes responsive to a drug stress. YKL061W is also currently un-annotated. Its pair, NTC20, in the learned network performs nuclear mRNA splicing. It is also known to physically interact with NUP57, NUP82, NUP1 and NUP116, which constitute a nuclear pore complex. It also interacts with UTP5, which performs snoRNA binding. This suggests that YKL061W has functions of binding or transporting activity in the nucleus. Another example is YPL251W. Its learned pair, BUD31, is

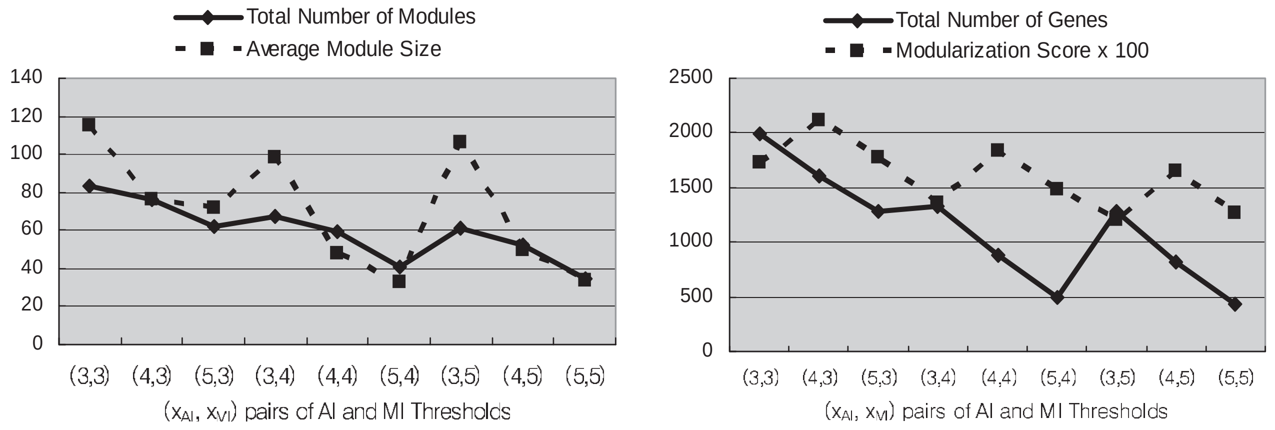

Fig. 6. Modularization results based on three factors, the total number of modules, the average module size and coverage.

involved in bud-site selection, and we have found that it is known to interact with BUD9, another protein involved in bud-site selection.

There are cases which confirm highly plausible, but currently unsettled, functional activities of genes. The representative example is YML131W. YML131W is known to be involved in the process of lipid, fatty acid and isoprenoid biosynthesis (Mewes *et al*., 1997), but its explicit biological role has not been clarified yet (GO Consortium, 2001). In our experiment, YML131W was selected as a *seed gene* during an oxidative stress. It turned out that YML131W and the only other member in a same module, YNL134C share a zinc-containing alcohol dehydrogenase superfamily domain (IPR002085). The function of YML131W as an oxidoreductase, based on its domain information, is strongly supported by its inferred interaction and the identified stress condition using the proposed method. This procedure shows a typical *in silico* process of how learned edges can be used for generating or substantiating the biological hypotheses.

3.2.3 Computational analysis Firstly, we examined how robust the proposed method is to the change of the AI and MI threshold values. The characteristics of the resulting modules are compared based on three factors: the total number of modules, the average module size and coverage (i.e. the total number of genes in identified modules). The results are illustrated in Figure 6. As thresholds increase, the total number of modules, the average module size and coverage decrease. The average module size and coverage seem to be affected more by the *AI* threshold value than by the *MI* threshold. We counted how many edges in final global networks from different threshold settings coincide, in other words, overlapped, with each other. As representative cases, we compared the results from four different modularization: modules with the two highest *Modularization Scores* and modules with the two lowest *Modularization Scores*. The Results are summarized in Table 1 and the rows in the table are ordered by their *Modularization Scores*. Even though the total number of genes among four cases is quite different, the ratio of overlapped edges among them is relatively high. Additionally, we counted how many edges (i.e. pairs of genes) in the final global networks share the same annotation in GO (GO Consortium, 2001). If two nodes of an edge share the same process or functional GO term, we counted it as a consistent edge with established biological knowledge. Otherwise, it was counted as an inconsistent one except the cases including unclassified ORFs. Component terms in GO were not used since they tend to distort the consistency rate due to the low specificity of terms such as ‘cytosol’. Since only identical terms are counted, the measured estimate is a rather conservative one. Generally, modularization results with higher *Modularization scores* are more reliable than those with lower ones based on their overlapping and consistency ratio.

Secondly, we compared the proposed method with two different alternatives: a whole-set-based approach (i.e. inference of Bayesian networks over a set of genes as a whole) and expression-based clustering approaches (i.e. inference of Bayesian networks over each gene cluster based on only mRNA expression data). For expression-based clustering, two common clustering algorithms were used to avoid an algorithm-specific bias: SOM toolbox for MATLAB (http://www.cis.hut.fi/projects/somtoolbox/) and K-means clustering. The number of clusters was automatically decided by the SOM toolbox software and this was also used for K-means clustering. Two extreme cases, Case I and Case IV in Table 1, were tested, and the result[3] is summarized in Table 2.

The whole-set-based approach performs worst for Case IV. Note that we did not apply any extra screening information such as transcription factors or gene perturbation to the whole-set-based approach in order to examine the pure effect of modularization. It is interesting that the proposed method comes up with not only a significantly higher ratio of consistent edges but also much abundant inferred edges than the whole-set-based one. However, the average number of edges in candidate networks from the whole-set-based approach was much greater than that of the proposed method. This suggests that the networks from the whole-set-based learning contain a much smaller ratio of consistent edges than that of the proposed method. Two expression-based clustering algorithms perform better than the whole-set-based approach, but worse than the proposed method. The performance gain of the proposed method is from 32 to 116%. Also, it is worth noting that the absolute number of consistent edges from our method far exceeds that from the expression-based methods. Detailed information about inferred networks such as directions and confidence scores of edges can be found in supplementary material.

[3] The whole-set-based learning result of Case I is not available due to the limited capacity of our software.

Table 2. Comparison of the proposed method with a whole-set approach and two expression-based clustering approaches

Method	Module number	Final edge number	Consistent edge number	Inconsistent edge number
Case I (1612 genes)				
The proposed method	76	328	71 (21.6%)	257 (69.8%)
Whole-set-based learning	1	N/A	N/A	N/A
SOM-cluster-based learning	60	94	10 (10.6%)	73 (77.7%)
K-means-cluster-based learning	60	113	10 (10.0%)	87 (77.0%)
Case IV (1290 genes)				
The proposed method	61	480	81 (16.9%)	325 (67.7%)
Whole-set-based learning	1	92	2 (2.2%)	77 (83.7%)
SOM-cluster-based learning	60	180	23 (12.8%)	128 (71.1%)
K-means-cluster-based learning	60	148	19 (12.8%)	107 (72.3%)

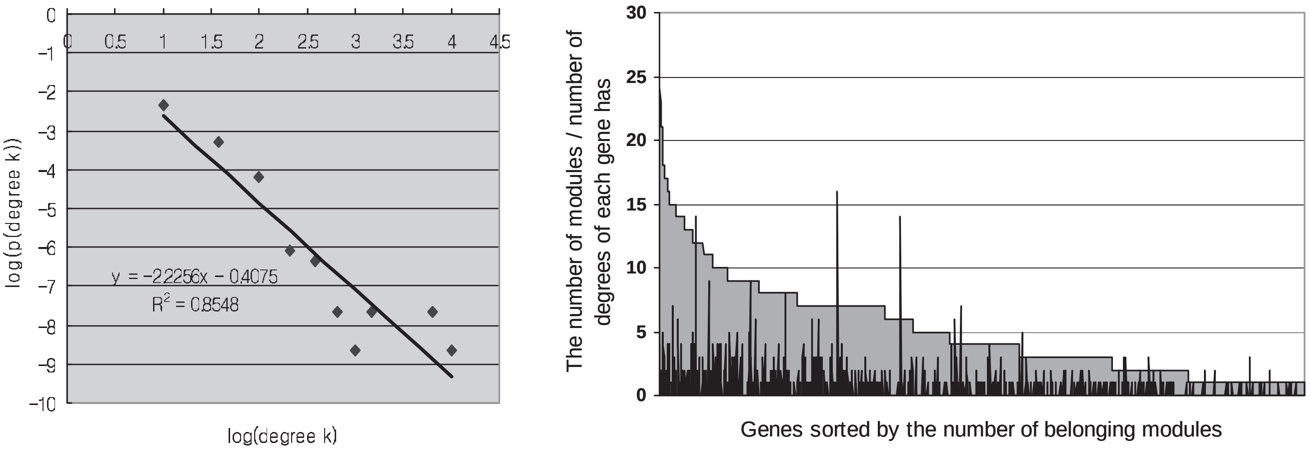

Fig. 7. (**a**) Scale-free characteristics of learned networks. When the degree, k, of the nodes of a scale-free network is plotted against the probability of occurrence of that degree, $P(k)$, on a log–log scale, the data form a straight line, the slope of which is $\gamma = -2.2256$ in this graph (Hallinan, 2004). (**b**) Dependency between the degree of a gene and the number of modules it belongs to.

Recent researches (Jeong *et al*., 2001; Hallinan, 2004) have shown that cellular networks show the characteristic of scale-free networks common in many natural networks. Here, we examined the scale-free characteristic of the final networks. Figure 7a shows that the probability $P(k)$ of finding a node with degree k in final networks follows a power law: $P(k) \propto k^{-\gamma}$ (here, $\gamma = -2.2256$). To confirm that this scale-free characteristic of final networks does not come from the overlapped *intermediary genes* in multiple modules, we examined the correlation between the number of modules each gene belongs to and the number of degrees each gene has (Fig. 7b). We use the Spearman's rank correlation coefficient and the result verifies that two factors show very low dependency: $R = 0.4347$, $N = 1290$, $p \leq 6.777\text{e}{-55}$ $(Z = 15.6079)$.

4 CONCLUDING REMARKS

To infer genetic interaction mechanisms in a cell, we have proposed a new method called MONET, which stands for modularized network learning using biological annotations and mRNA expression data. The proposed method presents a global picture of actively responding biological processes as well as a detailed look of relationships among genes. The whole procedure is composed of two main parts: Module Identification and Interaction Inference. In the Module Identification step, it identifies *seed genes* that show distinctive expression patterns in a specific experimental condition. Beginning with those *seed genes*, functionally related genes are grouped into different modules based on prior biological knowledge and expression data in terms of the *Annotation Information* (*AI*) and *Mutual Information* (*MI*) scores. In the Interaction Inference step, an existing Bayesian network learning algorithm is applied to each module to infer detailed interactions among genes. These separately inferred *subnetworks* over each module are integrated into final global networks through common *intermediary genes*.

The proposed method has several advantages over previous approaches. Firstly, since identified modules contain those genes whose biological annotations or mRNA expression patterns are tightly related, we expect that they would be more consistent with localized cellular processes than clustering-based modules, which are based on only expression patterns. Even though mRNA expression patterns can show close relationships between genes, there are many cases which conventional clustering methods cannot detect (Fashing *et al*., 2002). Moreover, overlapped modularization of genes rather than partitioning of them reflects the multiple activities of genes in a cell more realistically. Secondly, the proposed method can reduce false positive inferences among genes. Since Bayesian network learning is applied to a smaller number of genes given the same number of expression profiles, we can achieve a better ratio between the number of variables and observations for a learning procedure. Thirdly, common genes in multiple modules play intermediaries among modules so that our method can also infer inter-relationships of modules. This provides us an overall picture of cellular processes as well as detailed relationships between genes. Lastly, it can utilize the existing sophisticated learning techniques incorporating designed gene perturbation or transcription factor binding information, since those enhancements can be applied to the learning of intra-module networks.

We have analyzed the expression profiles of yeast stress data (Gash *et al.*, 2000) using the proposed method. The result was well in accordance with established biological knowledge besides suggesting some putative hypotheses and complementary evidences about the functions of currently unclassified genes. Currently, we are going to incorporate recent improvements of network learning procedures, such as utilization of gene perturbation and transcription factor binding information, into our framework.

ACKNOWLEDGEMENTS

We thank the anonymous referees for their valuable comments. This work was supported by the Korean Systems Biology Research Grant (2005-00343) from the Ministry of Science and Technology. We would also like to thank CHUNG Moon Soul Center for BioInformation and BioElectronics and the IBM SUR program for providing research and computing facilities.

REFERENCES

Akutsu,T., Miyano,S. and Kuhara,S. (2000) Algorithms for inferring qualitative models of biological networks. In *Proc. of Pacific Symposium on Biocomputing*. pp. 290–301.

Bader,G.D. *et al*. (2003) Bind: the biomolecular interaction network database. *Nucleic Acids Res.*, **31**, 248–250.

Calabretta, *et al*. (1998) A case study of the evolution of modularity: towards a bridge between evolutionary biology, artificial life, neuro- and cognitive science. In *Proc. of the Sixth International Conference on Artificial Life*. pp. 275–284.

Cherry,J.M. *et al*. (1998) SGD: *saccharomyces* genome database. *Nucleic Acid Res.*, **26**, 73–79.

Fashing,M. *et al*. (2002) A clustering algorithm explicitly designed to produce priors for Bayesian network discovery from whole-genome expression level data.

Francois,J. and Parrou,J.L. (2001) Reserve carbohydrate metabolism in the yeast *Saccharomyces cerevisiae. Fems. Microbiol. Rev.*, **25**, 125–145.

Friedman,N. *et al*. (1999) Learning Bayesian network structure from massive datasets: the 'sparse candidate' algorithm. In *Proc. of Fifteenth Conference on Uncertainty in Artificial Intelligence*. Association of Uncertainty in Artificial Intelligence, pp. 206–210.

Friedman,N. *et al*. (2000) Using Bayesian networks to analyze expression data. *J. Comput. Biol.*, **7**, 601–620.

Gash *et al*. (2000) Genomic expression program in the response of yeast cells to environmental changes. *Mol. Biol. Cell*, **11**, 4241–4257.

GO Consortium (2001) Creating the gene ontology resource: design and implementation. *Genome Res.*, **11**, 1425–1433.

Hallinan,J. (2004) Gene duplication and hierarchical modularity in intracellular interaction networks. *BioSystems*, **74**, 51–62.

Hartemink,A. *et al*. (2002) Combining location and expression data for principled discovery of genetic regulatory network models. In *Proc. of Pacific Symposium on Biocomputing*, 437–449.

Hazbun,T.R. *et al*. (2003) Assigning function to yeast proteins by integration of technologies. *Mol. Cell*, **12**, 1353-1365.

Hohmann,S. and Mager,W. ed (2003) *Yeast Stress Responses*, Topics in Current Genetics. Springer.

Jeong,H. *et al*. (2001) Lethality and centrality in protein networks. *Nature*, **411**.

Kohane,I., Kho,A. and Butte,A. (2003) *Microarrays for an Integrative Genomics*. MIT press.

Kwon,A. *et al*. (2003) Inference of transcriptional regulation relationships from gene expression data. *Bioinformatics*, **19**, 905–912.

Lam,W. and Bacchus,F. (1994) Learning Bayesian belief networks: an approach based on the mdl principle. *Comput. Intell.*, **10**, 269–293.

Liang,S., Fuhrman,S. and Somogyi,R. (1998) Reveal, a general reverse engineering algorithm for inference of genetic network architectures. In *Proc. of Pacific Symposium on Biocomputing*. pp. 18–29.

Lord,P. *et al*. (2003) Investigating semantic similarity measures across the gene ontology: the relationship between sequence and annotation. *Bioinformatics*, **19**, 1275–1283.

Mendizabal,I. *et al*. (1998) Yeast putative transcription factors involved in salt tolerance. *Fed. Eur. Biochem. Soc. Lett.*, **425**, 323-328.

Mewes,H. *et al*. (1997) MIPS: a database for protein sequences, homology data and yeast genome information. *Nucleic Acid Res.*, **25**, 28–30.

Middendorf,M. *et al*. (2004) Predicting genetic regulatory response using classification. *Bioinformatics*, **20**, i232–i240.

Neapolitan,R. (2004) *Learning Bayesian Networks*. Prentice Hall.

Peer,D. *et al*. (2001) Inferring subnetworks from perturbed expression profiles. *Bioinformatics*, **17**(S1), S215–S224.

Resnik,P. (1999) Semantic similarity in a taxonomy: an information-based measure and its application to problems of ambiguity in natural language. *J. Artif. Intell. Res.*, **11**, 95–130.

Schafer,J.L. (1997) Analysis of incomplete multivariate data. Chapman and Hall.

Segal,E. *et al*. (2003a) Discovering molecular pathways from protein interaction and gene expression data. *Bioinformatics*, **19** (Suppl.1), i264–i272.

Segal,E. *et al*. (2003b) Genome-wide discovery of transcriptional modules from DNA sequence and gene expression. *Bioinformatics*, **19**(Suppl. 1), i273–i282.

Segal,E. *et al*. (2003c) Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nat. Genet.*, **34**, 166–176.

Shamir,R. (2002) *Lecture note: analysis of gene expression data*. Tel Aviv University.

Tamada,Y. *et al*. (2003) Estimating gene networks from gene expression data by combining Bayesian network model with promoter element detection. *Bioinformatics*, **19**, ii227–ii236.

Tong,A.H.Y. *et al*. (2004) Global mapping of the yeast genetic interaction network. *Science*, **303**, 808–813.

Travers,K.J. *et al*. (2000) Functional and genomic analyses reveal an essential coordination between the unfolded protein response and er-associated degradation. *Cell*, **101**, 249–258.

Yoo,C., Thorsson,V. and Cooper,G. (2002) Discovery of causal relationships in a gene regulation pathway from a mixture of experimental and observational DNA microarray data. In *Proc. of Pacific Symposium on Biocomputing*, pp. 498–509.