Learning module networks from genome-wide location and expression data

Xiaojiang Xu[1], Lianshui Wang[1], Dafu Ding*

Key Laboratory of Proteomics, Institute of Biochemistry and Cell Biology, Shanghai Institutes for Biological Sciences, Chinese Academy of Sciences, Graduate School of the Chinese Academy of Sciences, Shanghai 200031, PR China

Received 1 November 2004; accepted 10 November 2004

Available online 23 November 2004

Edited by Robert B. Russell

Abstract We develop a systematic algorithm for discovering network of regulatory modules, which identifies regulatory modules and their regulation program by integrating genome-wide location and expression data. Unlike previous approaches [Eisen, M.B., Spellman, P.T., Brown, P.O. and Botstein, D. (1998) Proc. Natl. Acad. Sci. USA 95, 14863–14868; Tavazoie, S., Hughes, J.D., Campbell, M.J., Cho, R.J. and Church, G.M. (1999) Nat. Genet. 22, 281–285; Ihmels, J., Friedlander, G., Bergmann, S., Sarig, O., Ziv, Y. and Barkai, N. (2002) Nat. Genet. 31, 370–377; Segal, E., Shapira, M., Regev, A., Pe'er, D., Botstein, D., Koller, D. and Friedman, N. (2003) Nat. Genet. 34, 166–176] that relied primarily on gene expression data, our algorithm regards the regulator binding data as prior knowledge that provide direct evidence of physical regulatory interactions. We applied the method to a *Saccharomyces cerevisiae* genome-wide location data [Lee, T.I., Rinaldi, N.J., Robert, F., Odom, D.T., Bar-Joseph, Z., Gerber, G.K., Hannett, N.M., Harbison, C.T., Thompson, C.M., Simon, I., Zeitlinger, J., Jennings, E.G., Murray, H.L. Gordon, D.B., Ren, B., Wyrick, J.J., Tagne, J.B., Volkert, T.L., Fraenkel, E., Gifford, D.K. and Young, R.A. (2002) Science 298, 799–804] for 106 DNA-binding transcription factors and 250 gene expression experiments under the conditions from the cell cycle to responses to various stress conditions. The results show that our method is able to identify functionally coherent modules and their proper regulators. Supplementary materials are available at http://compbio.sibsnet.org/projects/module-network/.

Keywords: Module network; Regulatory program; Location probability

1. Introduction

The complex functions of a living cell are carried out through networks of interacting biochemical components. The key for biochemical networks is the proper context-dependent expression of genes. This activity is often coordinated by the organization of the genome into regulatory modules, i.e., sets of co-regulated genes that involve in a common function [6]. To achieve this, the cell has evolved a highly interconnected transcriptional networks composed of signaling molecules, transcription factors, and their DNA targets [7]. Genome-wide expression profiles provide important information about these cellular processes. Current approaches to analyzing gene expression data can successfully identify groups of co-expressed genes [1–3]. Friedman and his co-workers constructed a probabilistic model that use expression data to link genes with their regulators [4,8,9]. Their method assumes that the expression levels of the regulated genes are controlled by the expression level of their regulators. This assumption holds only when the expression level changes of the regulators, not the others (e.g., post-transcription modification of the regulators), are the regulatory signals. Some other approaches have combined expression data with additional information, such as shared DNA-binding motifs [10–12]. But these additional data sources provide essentially only indirect evidence of genetic regulatory interactions.

Large-scale, genome-wide transcription factor binding analysis, which identifies physical interactions between regulators and the regulatory DNA regions they bind to, provides direct evidence of regulatory relationships [5,13]. Although helpful, the validity of binding information is also limited, as the binding between the regulator and a certain regulatory region indicates only binding but not always functioning. The regulator acting positively, negatively or not at all depends on many conditions. Because expression data and location data provide complementary information, we commit to develop an efficient computational method for integrating them together. Such an algorithm could assign genes to modules and modules to regulators more accurately than the other methods based on one single data source alone.

In this paper, we report a computational approach based on a Bayesian probabilistic framework for inferring regulatory networks of gene modules from genome-wide location data and expression data. We begin with clustering genes into modules, using hierarchical clustering algorithm [1]. Then, for every module we perform an efficient exhaustive search over all possible transcriptional regulators by computing location probability from location data and measuring mutual information [14] from gene expression data, respectively. Once sets of strong candidate regulators were found, given these as inputs, we use iterative procedure (see Supplementary Methods online) to search for regulation programs of modules

*Corresponding author. Fax: +86 21 54921011. *E-mail addresses:* xjxu@sibs.ac.cn (X. Xu), dingdafu@server.shcnc.ac.cn (D. Ding).

[1] These authors contributed equally to this work.

Abbreviations: CPD, condition probability distribution; EM, expectation maximization; GO, gene ontology; TF, transcriptional factor

0014-5793/$22.00 © 2004 Federation of European Biochemical Societies. Published by Elsevier B.V. All rights reserved.
doi:10.1016/j.febslet.2004.11.019

(see Fig. 3) and re-assign genes into modules simultaneously. Every regulation program is organized as a regression tree [15] in which groups of co-regulated genes, their regulators, and the behavior of the module are specified as a function of the regulators' expression and the conditions under which regulation takes place. Finally, the procedure outputs a list of modules and corresponding regulation programs. (See Section 2 for a complete description of the algorithm.)

We applied our method to the study of gene regulatory modules in yeast. We considered genome-wide location data for 106 transcription factors [5] together with gene expression data on 250 conditions from different experiments. The results were compared with previous work [2,4,16] and showed that our algorithm could discover biologically meaningful, functionally coherent modules and their proper regulators. Therefore, our method is useful for studying transcriptional regulatory networks by integrating genome-wide location data and gene expression data.

2. Materials and methods

2.1. Candidate regulators

We compiled a set of 472 candidate regulators (see Supplementary Table 1 online), containing both transcription factors and signaling proteins that may have transcriptional effect [17]. The transcriptional factors (TFs) that were used in the genome-wide location analysis are all included.

2.2. DNA microarray data set

We used an *Saccharomyces cerevisiae* gene expression data set (details are available in online Supplementary data) that measures 250 transcription levels for each gene under various conditions from the cell cycle to responses to various stress conditions [18,19]. We chose a subset of 3224 genes with significant changes of expression level under the different conditions. Our gene set also included all genes chosen as candidate regulators.

2.3. Location data set

We used genome-wide location data for 106 transcription factors [5], which identified physical interactions between regulators and DNA regions they bind to.

2.4. Learning regulation programs and modules network

Below we describe the algorithm, with some details omitted owing to space constraints. See "Supplementary Methods" online for complete information.

We iteratively search for regulation programs (regulation trees) for each module and re-assign each gene to these regulatory modules by maximizing the Bayesian score that our modules network is correct. Then, we use expectation maximization (EM) algorithm [20] to search for modules network with the highest score. Each of the iteration consists of two steps: an E-step and an M-step (Fig. 1).

The M-step can be viewed as partitioning the genes into modules and learning the optimal regression tree for each module. Finding structure of Bayesian network that maximizes Bayesian score is often cast as an optimization problem [21,22]. For computational efficiency, at first we restrict the potential parents in each regulatory module to a small subset of candidates. We search for strong candidate regulators by measuring mutual information between the regulator and target gene expression profiles and computing location probability from location P-value. For a pair of gene i and regulator k, given location P-value, the location probability is

$$P_{ik} = \frac{e^{-w\rho_{ik}} - e^{-w}}{1 - e^{-w}} \tag{1}$$

where ρ_{ik} is a P-value in genome-wide location analysis and $w = 20$ [11] is weight of exponential distribution. A small location P-value suggests high probability for the binding of a regulator in the regulatory regions of a gene. We choose 25 sub-candidate regulators set with high mutual information and 25 sub-candidate regulators set with high location probability. Then, we combine these two sub-candidate regulators sets as strong candidate regulators. Secondly, we organize strong candidate regulators and genes in the regulatory module into regression tree, and then create a smaller regression tree that is pruned to the estimated best size. Given modules and their regulators in the regression trees, we consider them as prior structure of modules network. Our algorithm is based on the classical Bayesian network [23], which describes relationships of probabilistic dependency between variables (e.g., genes). We require that the genes in the same module have the same parents (regulators) and the same conditional probability distribution (CPD). We calculate location probability for every pair of genes and regulators and regard them as structure prior to the Bayesian score. Next, we compute the Bayesian score for this modules network (see Supplementary Methods online).

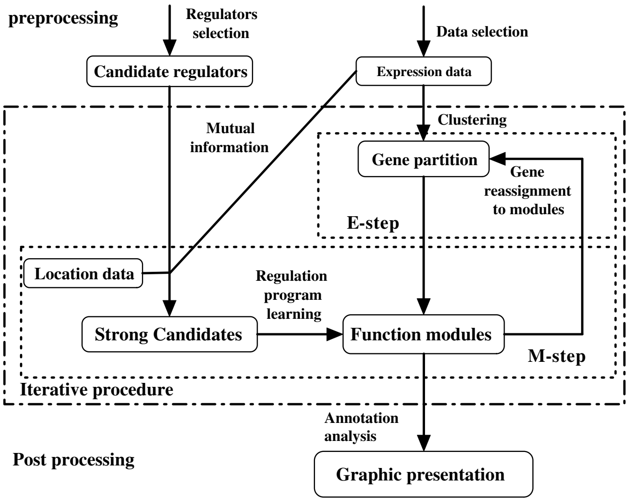

Fig. 1. Overview of our automated approach. The procedure takes as input a data set of gene expression profiles, a large precompiled set of candidate regulators genes and location analysis data. The core EM algorithm (dotted line box) is an iterative procedure, including two steps: an E-step procedure which partition genes to modules; and M-step which learns regulatory program for each module. In a post-processing phase, we evaluate the validity of each module by testing the enrichment of the genes from the same category.

In the E-step, given the inferred regulation tree, we re-assign each gene to the module whose program optimally predicts the gene's behavior. We compute the CPD for every gene with their inferred regulation tree and pick up the gene in every regulatory module whose behavior is worst predicted by the regression tree and put them into a pool. Then, we compute the CPD for every gene in the pool with every inferred regulation tree. Every gene is assigned into the regulatory module with the highest CPD. We avoid assigning a regulator gene to a module in which it is also a regulatory input.

We initialized our iterative algorithm with 50 clusters by using standard clustering procedure [1] and creating one module from each of the resulting clusters. The EM algorithm is applied to refining both the gene partition and the regulatory tree. These two steps were iterated until convergence is reached.

2.5. Evaluating statistical significance for functional category enrichment of modules

The hypergeometric distribution was used to determine statistical significance for the biological relevance of a module. We discarded all annotations associated with less than five genes in our gene set and got a list of 404 gene ontology (GO) [24] categories and 296 MIPS [25] categories. For each module, we computed the fraction of genes in the module associated with each category and used the hypergeometric distribution to calculate a corresponding P-value. We carried out a Bonferroni correction for multiple independent hypotheses and took P-value below $0.05/n$ ($n = 404$ and 296 for GO, MIPS annotations, respectively) as being significant.

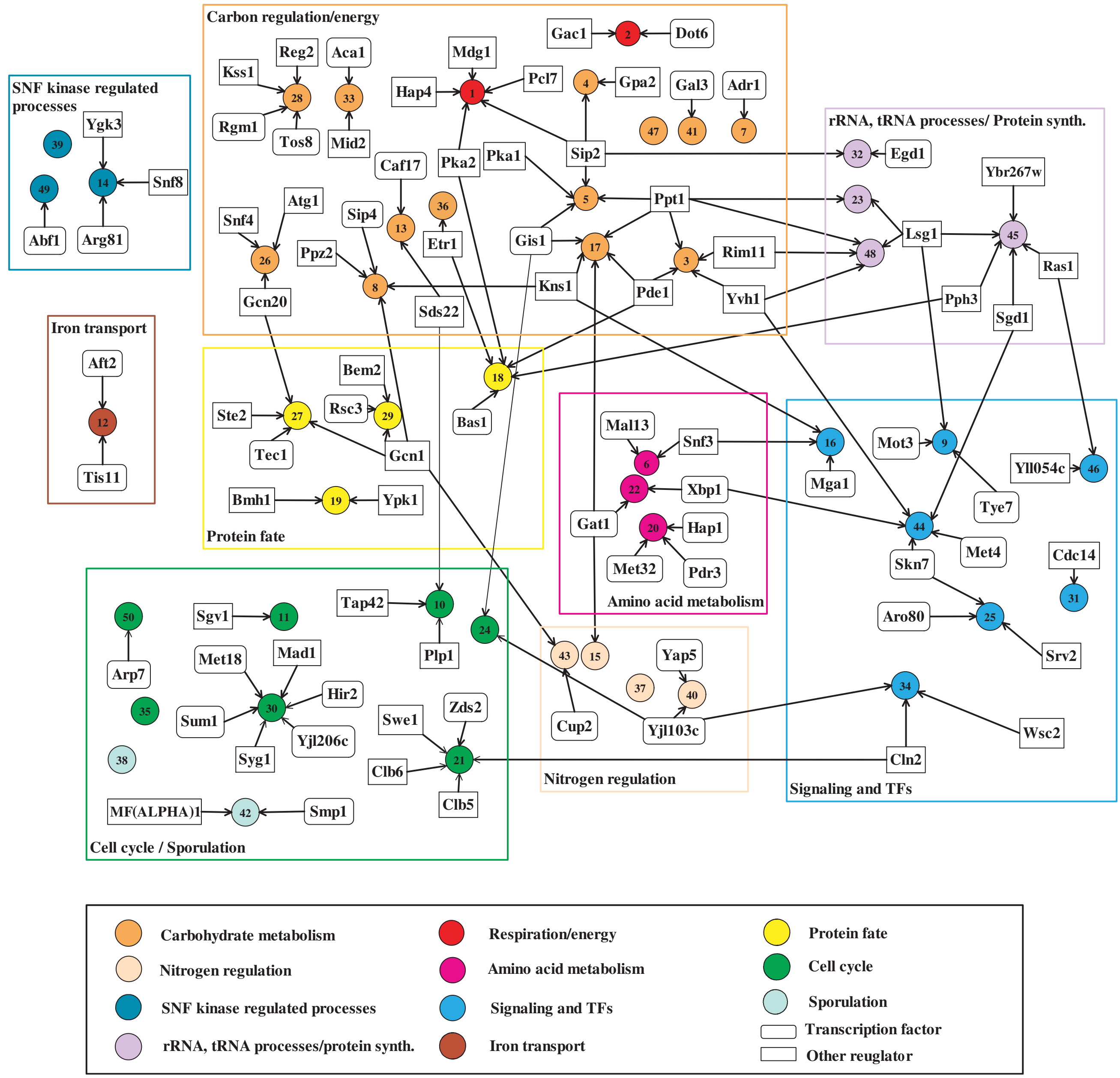

Fig. 2. Global view of modules network. The modules network is visualized as a directed graph with directed edges between regulators and regulatory modules. The modules network consists of 50 regulatory modules and 86 regulators. In most cases, one module is controlled by two or more regulators, which show combinatorial interactions. Regulatory modules are colored according to the GO annotation and MIPS category to which a significant number of genes from the same category belong ($P < 0.01$).

3. Results and discussion

We compiled a list of 472 candidate regulators and applied the procedure described above to the 250 arrays of the yeast stress and cell-cycle data set and to genome-wide location data for 106 transcription factors. We identified 50 modules, containing 3224 distinct genes and regulated by 86 of the regulators. The inferred regulatory modules spanned a wide range of biological processes, including metabolic and energy pathway, various stress responses, cell cycle-related process, molecular functions (e.g., protein folding) and signal transduction (e.g., Snf1 kinase-regulated processes). Fig. 2 presents a global view of these results as a graph with edges linked between regulatory modules and their regulators.

3.1. Modules and their regulation program

We found 15 cohesive modules that participated in the process of respiration and energy metabolism (see Table 1). The respiration and energy I module (see Fig. 3) is a clear example of predicted module. It consists mainly of genes encoding energy synthesis proteins (10 of 19) and respiration protein (5 of 19). The inferred regulation program specifies the Pka2 protein, a catalytic subunit of the cAMP-dependent protein kinase (PKA), as the module's top regulator. This prediction is supported by a recent study [26] showing that the expression of several genes in the module (for example, Atp2) is PKA-dependent. The Hap4 transcription factor is induced when Pka2 is activated, primarily under stationary phase (a growth phase in which nutrients, mainly glucose, are depleted). The predic-

Table 1
Summary of module analysis and validation

#	Module[a]	# G[b]	C (%)[c]	Reg.[d]	L	Reg.[d]	L	Reg.[d]	L	Reg.[d]	L	Reg.[d]	L	Reg.[d]	L
1	Respiration and energy I	19	79	PKA2	Y	HAP4	Y	SIP2	Y	MDG1	N	PCL7	Y		
2	Respiration and energy II	51	69	DOT6	N	GAC1	Y								
3	Carbohydrate metabolism	276	62	YVH1	Y	PDE1	Y	PPT1	Y	RIM11	Y				
4	Carbon regulation and protein catabolism	231	50	SIP2	Y	GPA2	Y								
5	Carbon regulation and signaling	23	65	SIP2	Y	PPT1	Y	GIS1	Y	PKA1	Y				
6	Amino acid metabolism, gluconeogenesis and hexose transport	141	57	SNF3	Y	MAL13	Y								
7	Glucose repression	12	58	ADR1	Y										
8	Gluconeogenesis and TFs	199	57	PPZ2	Y	SIP4	Y	KNS1	N	GCN1	Y				
9	TFs	17	35	MOT3	Y	TYE7	Y	LSG1	N						
10	Cell cycle and general TFS	20	85	SDS22	Y	TAP42	N	PLP1	N						
11	Cell cycle and budding	6	83	SGV1	Y										
12	Iron transport	13	38	TIS11	Y	AFT2	Y								
13	Aldehyde metabolism	30	60	CAF17	Y	SDS22	Y								
14	SNF kinase regulated processes	19	84	YGK3	Y	ARG81	Y	SNF8	Y						
15	Nitrogen transport and metabolism	38	63	GAT1	Y										
16	Signaling and transport	59	56	SNF3	Y	MGA1	Y	KNS1	Y						
17	Carbon regulation and gluconeogenesis	29	55	PPT1	Y	PDE1	Y	KNS1	Y	GIS1	Y	GAT1	Y		
18	Glycogen metabolism, protein fate and cAMP signaling	117	56	PDE1	Y	BAS1	Y	PKA2	Y	ETR1	Y	PPH3	Y		
19	Protein fate and carbohydrate metabolism	29	45	BMH1	Y	YPK1	Y								
20	Sulfate amino acid and purine metabolism and Ty ORFs	36	83	PDR3	N	MET32	Y	HAP1	Y						
21	Cell cycle (G1/S) and DNA replication	40	72	CLN2	Y	CLB5	Y	ZDS2	Y	SWE1	Y	CLB6	Y		
22	Amino acid and purine metabolism	29	79	XBP1	Y	GAT1	Y								
23	Ribosome and protein biosynthesis	349	66	LSG1	Y	PPT1	Y								
24	Cell cycle (G1/S) and signaling I	22	77	YJL103C	N	GIS1	Y								
25	Osmosensory signaling pathway	9	78	SRV2	Y	ARO80	N	SKN7	Y						
26	Glycolysis,gluconeogenesis and proteasome	68	51	SNF4	Y	GCN20	Y	ATG1	Y						
27	Protein folding and signaling	12	75	GCN20	Y	TEC1	Y	GCN1	Y	STE2	Y				
28	Regulation of redox homeostasis	12	50	REG2	N	KSS1	N	RGM1	N	TOS8	N				
29	Nuclear pore transport and signaling	39	41	BEM2	Y	RSC3	Y	GCN1	N						
30	Cell cycle (G1/S) and signaling II	12	42	MET18	Y	MAD1	Y	HIR2	Y	YJL206C	N	SYG1	Y	SUM1	Y
31	ER and carbon regulation	77	60	CDC14	N										
32	Protein biosynthesis and ER	243	52	EGD1	Y	SIP2	Y								
33	Carbon regulation and cell wall	27	67	ACA1	Y	MID2	Y								
34	Cell wall protein modification and signaling	15	67	YJL103C	N	WSC2	Y	CLN2	Y						
35	Cell cycle	21	81												
36	Carbon regulation	41	56	ETR1	Y										
37	Nitrogen regulation	23	57												
38	Sporulation I	11	55												

39	SWI/SNF protein kinase regulated procession I	29	66										
40	Nitrogen regulation and transport	18	56	YJL103C	N	YAP5	N						
41	Galactose metabolism and cell wall	30	47	GAL3	Y								
42	Sporulation II	66	27	MF(ALPHA)1	Y	SMP1	Y						
43	Nitrogen regulation and TFs	26	46	GCN1	Y	CUP2	N						
44	General TFs	26	54	SKN7	Y	MET4	Y	XBP1	Y	YVH1	Y	SGD1	Y
45	rRNA and tRNA processing	150	41	SGD1	Y	YBR267W	N	PPH3	Y	LSG1	Y	RAS1	Y
46	Signaling	8	100	YLL054C	N	RAS1	Y						
47	Chromatin metabolism and carbon regulation	58	53										
48	rRNA and tRNA transcription and processing	268	33	LSG1	Y	YVH1	Y	PPT1	Y	RIM11	Y		
49	SWI/SNF protein kinase procession II	56	50	ABF1	Y								
50	Cell cycle and SWI/SNF protein kinase processes	68	54	ARP7	Y								

[a] Each module was assigned a name based on the largest one or two categories of genes in the module (combining significant gene annotations and the literature).These concise names are used to facilitate the presentation and may not cover the full content of some of the more heterogeneous modules or modules with lots of unknown genes.

[b] Number of genes in the module.

[c] Functional/biological coherence of each module, measured as the percentage of genes in module covered by significant gene annotations ($P < 0.01$).

[d] Regulators predicted to regulate each module. Some modules (35,37–39,47) did not have regulators, as no good regulatory tree could predict the expression profile of their gene members. L, literature evidence showing that the respective regulator regulates at least one of the genes, or a process significantly overrepresented in the module. Y, literature evidence present (see Supplementary Table 2 for the references); N, literature evidence absent.

tion is consistent with the known role of Hap4 in the process of the activation of respiration [27]. Our results suggest that these changes are regulated by combinatorial interaction of the transcription factor and the protein kinase. When Pka2 and Hap4 are not induced, this module could either activate more mildly or repressed. It implies that other regulators, such as the Snf1 kinase subunits Sip2, regulate those changes of genes expression. This prediction is supported by a recent study [28,29] showing that Glucose depletion regulates gene expression via Snf1 (Sip2) and cAMP-dependent protein kinase (Pka2) pathway.

We also found seven modules (see Table 1) involved in the cell-cycle regulatory processes. They are significantly enriched for functional categories, as most of the modules (5/7) had a coherence level above 72%. For example, genes in the module 21 (see Supplementary regulation programs online) take part in DNA replication and repair (19 of 40) and cell-cycle regulation (12 of 40). Cln2 is suggested as the top regulator of the module, supported by a study [30] showing that cyclin Cln2 associated with Cdc28 control G1/S. Moreover, cyclin Clb5 and Clb6 are involved in DNA replication [31] and regulators Swi1 and Zds2 involved in G1/S [31,32].

Modules relating to rRNA and tRNA processing and protein biosynthesis (Table 1) contain lots of significantly enriched genes. The numbers of genes in most of these modules are more than 150. This illustrates our method's capability to identify big expression signatures. For example, module 48 (see Supplementary regulation programs online) consists of 268 genes, in which 106 are in rRNA transcription and processing category, 35 are in tRNA transcription and processing category, 22 are in nucleus transport category, and 62 have unknown function. Lsg1 is suggested as the key regulator of this module, which is required for ribosomal subunit biogenesis and location [33]. Protein phosphatases Yvh1 and Ppt1 and phosphorylase Rim11 involve in regulation of the cAMP-dependent protein kinase cascade [34,35].

We also found nitrogen regulation and amino acid metabolism modules (see Table 1). For example, the nitrogen transport and metabolism module (see Supplementary regulation programs online) show the ability of our method to capture an entire cellular response, whose genes participate in diverse nitrogen metabolic pathway and cellular roles (6 of 38 in allantoin and urea metabolism, 18 of 38 in amino acid metabolism and transport and 8 of 38 in sulfur or methionine metabolism). Our results suggest that Gat1 is the regulator of the module, and this prediction is supported by a recent study [36] showing that transcription of nitrogen-catabolic genes is activated by Gat1. However, it is known that nitrogen metabolism is regulated by many other transcription factors including Gln3, Dal80, and Dal81. We failed to find regulation relationship between these additional regulators and their target genes because of limited expression data. Although these additional regulators had high transcription levels on the condition of nitrogen depletion, only 10 time points were experimentally measured, which is still insufficient for our method to identify all the regulators.

3.2. Evaluation of modules and their regulation program

We analyzed all the 50 resultant modules to test whether the proteins encoded by genes in the same module had related functions. We computed the functional/biological coherence of each module (see Table 1) based on the percentage of the

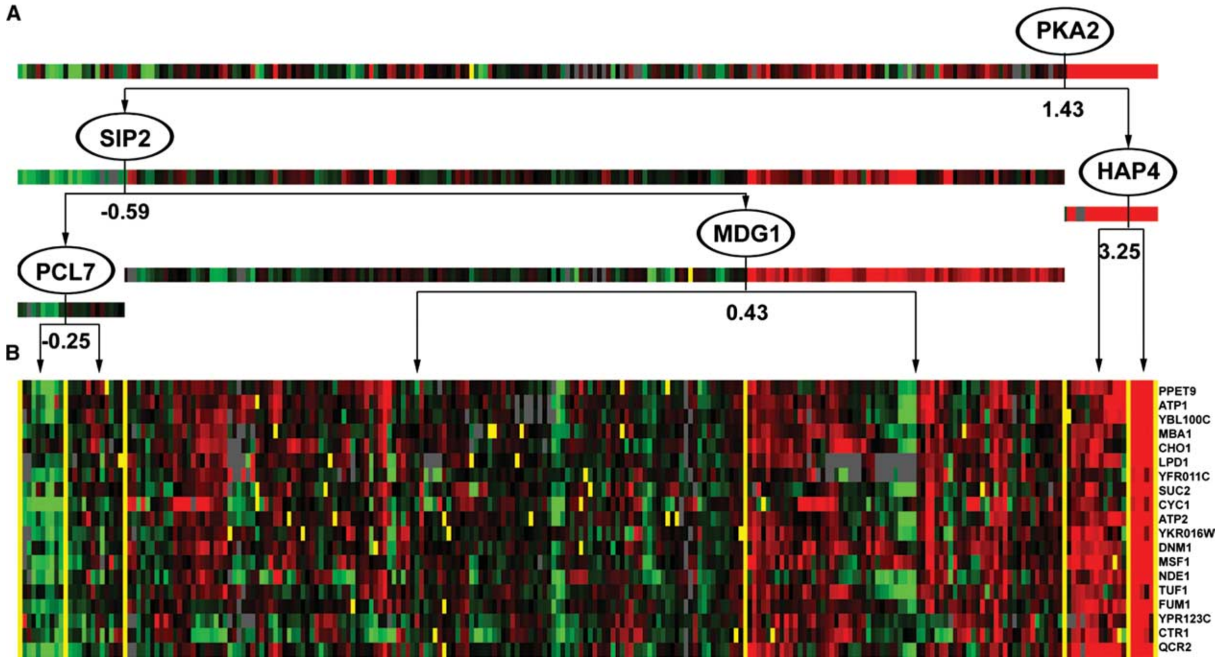

Fig. 3. The regulation program of the respiration and energy I module (19 genes). (A) Regulation tree. Each node in the tree represents a regulator (for example, PKA2) and a qualitative value, which trigger a query "if the expression of regulatory gene is bigger than the qualitative value?" Right branches represent the expression conditions, under which the answer to the query in the node is TRUE; left branches represent the expression conditions under which the answer is FALSE. The expression levels of the regulators themselves are shown below their corresponding nodes. (B) Gene expression profiles. Rows represent different genes and columns represent different arrays. Arrays are arranged according to the regulation tree. For example, the rightmost leaf includes the arrays in which PKA2's expression is greater than 1.43 and HAP4's expression is greater than 3.25.

genes covered by annotations significantly enriched in the module. For example, in the respiration and energy I module (see Table 1) the functional coherence is 79%. Here, we define functional homogeneous modules as modules with functional coherence percentage greater than 50%; we define functional heterogeneous modules as modules with functional coherence percentage less than 30%.

Previous work identified regulatory modules from genome-wide expression data by clustering [2] and the model of module networks [4]. Clustering algorithm grouped genes with highly correlated expression profile. Modules network organized genes into modules according to Bayesian posterior probability. Because genome-wide expression data provide essentially only indirect evidence of genetic regulatory interactions, these methods cannot reliably distinguish among genes that have similar expression patterns but are under the control of various regulatory networks. Clustering method [2] clustered 3000 genes into 30 modules. However, only five of them (17%) were functional homogeneous modules, other 23 of the 30 clusters (77%) had no significantly common biological function (see Table 2). The model of module networks [4] clustered 2355 genes into 50 modules and 31 of them (62%) were functional homogeneous modules (see Table 2). Despite the fact that gene expression data are useful for deriving regulatory modules, our algorithm can complement the limitation of using expression data alone by integrating location data with gene expression data. Our systematic algorithm organized 3225 genes into 50 modules. Overall, most regulatory modules (40 of 50) were functional homogeneous modules and only 1 of 50 (2%) had no significantly common biological function (see Tables 1 and 2). This indicates that our algorithm is capable of identifying highly biologically relevant modules.

Although the location P-value data alone are potentially useful for linking a set of regulators with a set of genes to which the regulators bind, our algorithm can compensate the limitation of these data alone by integrating expression data. To determine regulatory relationship between genes from location data, previous work used a statistical model and chose a relatively stringent P-value threshold (<0.001) with the intention of reducing false positives at the expense of false negatives [5]. Our algorithm presents a useful alternative to such single P-value threshold to predict binding events, because our method not only uses location probability to search for strong candidate regulators, but also regards it as structure prior of modules network to compute Bayesian score and to evaluate regulatory relationship between regulators and modules. For

Table 2
A comparison of the regulatory modules and their regulators identified by different methods

Method	Total genes	Total modules	Functional homogeneous modules	Functional heterogeneous modules	Total regulators
Our method	3224	50	40	1	86
Clustering [2]	3000	30	5	23	
Modules network [4]	2355	50	31	4	80
GRAM [16]	655	106	62	32	68

Functional homogeneous modules are modules whose functional coherence level is above 50%; Functional heterogeneous modules are modules whose functional coherence level is below 30%.

example, Hap4 is a well-characterized regulator of genes involved in carbohydrate metabolism and respiration [27]. The Hap4 module contains 19 genes that are involved in respiration and show a high function coherence level (79%) (Table 1). Eight of these genes (PET9, ATP1, ATP2, CYC1, MBA1, NDE1, FUM1 and QCR2) would not have been identified as Hap4 targets using the stringent 0.001 *P*-value threshold.

GRAM algorithm was designed to infer transcriptional regulatory networks through the combination of genome-wide location and expression data too. The GRAM algorithm [16] clustered 655 genes into 106 modules and linked transcriptional regulators with sets of genes by combining location data with expression data. In their study, 62 of the 106 modules (58.5%) were functional modules, but 32 of the 106 modules (30%) had no significantly common biological function (see Table 2). The method considered only 106 candidate transcriptional regulators, but Genetic networks in a living cell include at least 450 candidate regulators, which are far more than 106 transcription factors that are used in the location experiment. Alternatively, we compiled a set of 472 candidate regulators (see Supplementary Table 1 online), including 106 transcription factors used in the genome-wide location analysis. So, our algorithm can exhaustively search for strong candidate regulators over much more possible transcriptional regulators. As a result, we identified 86 of the regulators for 42 modules (see Tables 1 and 2), but previous work [16] identified only 68 of the transcriptional regulators for 106 modules.

We organized the regulation program as a regression tree, which specified the expression behavior of the module as a function of regulators' expression and the conditions under which regulation took place. For example, in the respiration and energy I module, five regulators (Pka2, Hap4, Sip2, Mdg1 and Pcl7) constructed a regulation tree (see Fig. 3). We found that the regulation programs generally assigned regulators accurately to regulatory modules, whose functions were consistent with the regulator's known role. We compared the known function of the inferred regulators with the method's predictions, where the known function is based on a compiled list of literature references (see Supplementary Table 2 online), in which direct experimental evidence exists for the role of the predicted regulators. Most of the modules (42 of 50) included genes known to be regulated by at least one of the module's predicted regulators (see Table 1 and Supplementary Table 2 online).

4. Conclusion

In this study, we have shown that our algorithm could identify biologically relevant regulatory modules and accurately assign regulators to modules whose functions were consistent with the regulator's known roles by integrating expression data and location data. In identifying regulatory modules, our algorithm is more powerful than clustering and other methods on the basis of correlated expression. In discovering regulatory relationship between regulators and genes in modules, our method is more useful than previous method [5] that choose a single *P*-value threshold to predict binding events. On the one hand, we refined candidate regulators from a set of 472 candidate regulators by computing location probability and mutual information between regulators and genes of modules. On the other hand, our iterative algorithm used location probability as a structure prior of Bayesian score and computing likelihood of Bayesian score from expression data. In each iteration, not only in the step of assigning genes to modules but also in the step of searching for regulators for each regulatory module, our algorithm considered both genome-wide location data and expression data.

Additionally, our algorithm could identify both regulatory modules and their control programs, which suggest concrete regulators for each module, their combinatorial interactions and the experimental conditions under which they are activated. Maybe a prominent feature of our method is its ability to generate detailed testable hypotheses concerning the role of specific regulators and the conditions under which this regulation takes place (see Fig. 3). We have validated the predicted results by using experimental evidence that is showed on a compiled list of literature references (see Supplementary Table 2). Regulatory roles of many genes related to signal transduction have been identified, which have post-transcriptional changes (see Table 1). As demonstrated, the algorithm can integrate sources of genome-wide location and expression data to compensate for technical limitations in location experiment and expression data.

Despite the successes described above, our method failed to identify certain regulatory relations, especially in serial regulation of transcription regulators in cell cycle. In some cases, we notice that the change of the regulator's expression level is not so significant that experiment can detect; in other cases, it is due to that we do not have the complete location analysis data over all the regulators (only 106 have known experiment data) yet. When more diverse gene expression data and location data of regulators become available in the future, we believe that important new insights in understanding the complex networks of biological regulation will be gained.

Acknowledgments: We thank Dr. Haixu Tang (Indiana University, Bloomington) for his useful comments on this manuscript and Dr. Eran Segal (University of Stanford, Stanford, CA) for critical discussions. This work was supported by the National High Technology Research and Development Program of China (863 Program) (No. 2002AA234021) and the Knowledge Innovation Program of the Chinese Academy of Sciences (KSCX2-2-07, KJCX1-08).

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version at doi:10.1016/j.febslet.2004.11.019.

References

[1] Eisen, M.B., Spellman, P.T., Brown, P.O. and Botstein, D. (1998) Proc. Natl. Acad. Sci. USA 95, 14863–14868.

[2] Tavazoie, S., Hughes, J.D., Campbell, M.J., Cho, R.J. and Church, G.M. (1999) Nat. Genet. 22, 281–285.

[3] Ihmels, J., Friedlander, G., Bergmann, S., Sarig, O., Ziv, Y. and Barkai, N. (2002) Nat. Genet. 31, 370–377.

[4] Segal, E., Shapira, M., Regev, A., Pe'er, D., Botstein, D., Koller, D. and Friedman, N. (2003) Nat. Genet. 34, 166–176.

[5] Lee, T.I., Rinaldi, N.J., Robert, F., Odom, D.T., Bar-Joseph, Z., Gerber, G.K., Hannett, N.M., Harbison, C.T., Thompson, C.M., Simon, I., Zeitlinger, J., Jennings, E.G., Murray, H.L., Gordon,

D.B., Ren, B., Wyrick, J.J., Tagne, J.B., Volkert, T.L., Fraenkel, E., Gifford, D.K. and Young, R.A. (1996) Science 298, 799–804.

[6] Werner, T. (2001) Pharmacogenomics 2, 25–36.

[7] Levine, M. and Tjian, R. (2003) Nature 424, 147–151.

[8] Friedman, N., Linial, M., Nachman, I. and Pe'er, D. (2000) J. Comput. Biol. 7, 601–620.

[9] Pe'er, D., Regev, A., Elidan, G. and Friedman, N. (2001) Bioinformatics 17 (Suppl. 1), S215–224.

[10] Bussemaker, H.J., Li, H. and Siggia, E.D. (2001) Nat. Genet. 27, 167–171.

[11] Segal, E., Barash, Y., Simon, I., Friedman, N. and Koller, D. (2002). In: Proc. 6th Inter. Conf. on Research in Computational Molecular Biology (RECOMB), pp. 263–272.

[12] Wang, W., Cherry, J.M., Botstein, D. and Li, H. (2002) Proc. Natl. Acad. Sci. USA 99, 16893–16898.

[13] Ren, B., Robert, F., Wyrick, J.J., Aparicio, O., Jennings, E.G., Simon, I., Zeitlinger, J., Schreiber, J., Hannett, N., Kanin, E., Volkert, T.L., Wilson, C.J., Bell, S.P. and Young, R.A. (2000) Science 290, 2306–2309.

[14] Friedman, N., Nachman, I. and Pe'er, D. (1999) In: Proc. Fifthteenth Conference on Uncertainty in Artificial Intelligence (UAI'99), pp. 196–205.

[15] Breiman, L., Friedman, J., Olshen, R. and Stone, C.J. (1984) Classification and Regression Trees, Wadsworth, Belmont, CA.

[16] Bar-Joseph, Z., Gerber, G.K., Lee, T.I., Rinaldi, N.J., Yoo, J.Y., Robert, F., Gordon, D.B., Fraenkel, E., Jaakkola, T.S., Young, R.A. and Gifford, D.K. (2003) Nat. Biotechnol. 21, 1337–1342.

[17] Pe'er, D., Regev, A. and Tanay, A. (2002) Bioinformatics 18 (Suppl. 1), S258–267.

[18] Spellman, P.T., Sherlock, G., Zhang, M.Q., Iyer, V.R., Anders, K., Eisen, M.B., Brown, P.O., Botstein, D. and Futcher, B. (1998) Mol. Biol. Cell 9, 3273–3297.

[19] Gasch, A.P., Spellman, P.T., Kao, C.M., Carmel-Harel, O., Eisen, M.B., Storz, G., Botstein, D. and Brown, P.O. (2000) Mol. Biol. Cell 11, 4241–4257.

[20] Friedman, N. (1998) In: Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI), pp. 129–138.

[21] Cooper, G.F. and Herskovits, E. (1992) Mach. Learning 9, 309–347.

[22] Heckerman, D., Geiger, D. and Chickering, D.M. (1995) Mach. Learning 20, 197–243.

[23] Pearl, J. (1988) Probabilistic Reasoning in Intelligent Systems. Networks of Plausible Interference, Morgan Kaufmann, San Fransisco, CA.

[24] Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T., Harris, M.A., Hill, D.P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J.C., Richardson, J.E., Ringwald, M., Rubin, G.M. and Sherlock, G. (2000) Nat. Genet. 25, 25–29.

[25] Mewes, H.W., Frishman, D., Guldener, U., Mannhaupt, G., Mayer, K., Mokrejs, M., Morgenstern, B., Munsterkotter, M., Rudd, S. and Weil, B. (2002) Nucleic Acids Res. 30, 31–34.

[26] Norbeck, J. and Blomberg, A. (2000) Yeast 16, 121–137.

[27] Gancedo, J.M. (1998) Microbiol. Mol. Biol. Rev. 62, 334–361.

[28] Ashe, M.P., De Long, S.K. and Sachs, A.B. (2000) Mol. Biol. Cell 11, 833–848.

[29] Schuller, H.J. (2003) Curr. Genet. 43, 139–160.

[30] Simon, I., Barnett, J., Hannett, N., Harbison, C.T., Rinaldi, N.J., Volkert, T.L., Wyrick, J.J., Zeitlinger, J., Gifford, D.K., Jaakkola, T.S. and Young, R.A. (2001) Cell 106, 697–708.

[31] Schwob, E. and Nasmyth, K. (1993) Genes Dev. 7, 1160–1175.

[32] Hsu, C.L., Chen, Y.S., Tsai, S.Y., Tu, P.J., Wang, M.J. and Lin, J.J. (2004) Nucleic Acids Res. 32, 511–521.

[33] Kallstrom, G., Hedges, J. and Johnson, A. (2003) Mol. Cell. Biol. 23, 4344–4355.

[34] Beeser, A.E. and Cooper, T.G. (2000) J. Bacteriol. 182, 3517–3528.

[35] Griffioen, G., Swinnen, S. and Thevelein, J.M. (2003) J. Biol. Chem. 278, 23460–23471.

[36] Coffman, J.A., Rai, R., Cunningham, T., Svetlov, V. and Cooper, T.G. (1996) Mol. Cell. Biol. 16, 847–858.