Parallel Bayesian Network Learning

For Inferring Gene Regulatory Networks

YoungHoon Kim, Doheon Lee
Department of BioSystems, Korea Advanced Institute of Science and Technology
373-1 Guseong-dong, Yuseong-gu, Daejeon 305-701, Republic of Korea
Email : yhkim@biosoft.kaist.ac.kr, dhlee@biosoft.kaist.ac.kr

ABSTRACT: Cell phenotypes are determined by the concerted activity of thousands of genes and their products. This activity is coordinated by a complex network that regulates the expression of genes. Understanding this organization is crucial to elucidate cellular activities, and many researches have tried to construct gene regulatory networks from mRNA expression data which are nowadays the most available and have a lot of information for cellular processes.

Several computational tools, such as Boolean network, Qualitative network, Bayesian network, and so on, have been applied to infer these networks. Among them, Bayesian networks that we chose as the inference tool have been often used in this field recently due to their 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. To alleviate this problem, we had developed the algorithm of MONET(MOdularized NETwork learning), which is a new method for inferring modularized gene networks by utilizing two complementary sources of information: biological annotations and gene expression. Afterward, we have packaged and improved MONET by combining dispersed functional blocks, extending species which can be inputted in this system, reducing the time complexities by improving algorithms, and simplifying input/output formats and parameters so that it can be utilized in actual fields. In this paper, we present the architecture of MONET system that we have improved.

1 INTRODUCION

Cell phenotypes are determined by the concerted activity of thousands of genes and their products. This activity is coordinated by a complex network that regulates the expression of genes. Understanding this organization is crucial to elucidate cellular activities, and many researches have tried to construct gene regulatory networks from mRNA expression data which are nowadays the most available and have a lot of information for cellular processes. Among several computational formalisms, such as Boolean networks and qualitative networks, Bayesian networks have drawn increasing attention due to well-established theoretical foundation and statistiçal robustness. 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. 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. have introduced two statistical techniques, Sparse Candidates 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. have applied the chromatin immuno-precipitation (CHIP) assay and Tamada et al. 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. They used clustering methods to divide a gene set into smaller groups, and applied network learning over each module.

To solve this problem, we had proposed a new method for inferring modularized gene networks by utilizing two complementary sources of information: biological annotations and gene expression.

Recently, many bioinformatics-related algorithms have been developed by various researchers, but there have been not many softwares that we can utilize in actual use, and that we can use easily with simple usage. Therefore, we have not only developed the proposed algorithm but also packaged and improved it so that it can be utilized in actual field. To do that, we have combined dispersed functional blocks, extended species which can be inputted in this system, reduced the time complexities by improving algorithms, and simplified input/output formats and parameters.

2 SYSTEM ARCHITECTURE

2.1 Overview of Modularized Network Learning

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.

2.2 Functional Description for Each Block

2.2.1 Preprocessing Block

a. Calculating Annotation Information (AI) for every gene pairs

To identify genes involved in the same cellular processes as seed genes, we utilize biological annotations such as MIPS or GO. This prior knowledge provides us with reliable explanations about 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. showed that a semantic tree can enable us to calculate the similarity of two biological 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 2. First, we build a semantic tree K from biological annotations. 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 as follows.

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

The Annotation Information(AI) score of two genes g_i and g_j is defined based on the Similarity score S of their annotation terms.

$$AI(g_i, g_j) = \sum_{f_k \in (AT(g_i) \cap AT(g_j))} S(f_k, f_k) + \max_{(f_i \in (AT(g_i) \cap AT^c(g_j))) \wedge (f_j \in (AT^c(g_i) \cap AT(g_j)))} S(f_i, f_j)$$

$$\mathrm{AT}(g_i) : \text{a set of annotation terms for a gene } i$$
$$\mathrm{AT}(g_j) : \text{a set of annotation terms for a gene } j$$

Annotation terms in AT(g_i) and AT(g_j) can be divided to two categories: terms in common 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 Annotation Information (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.

b. Calculating Mutual Information (MI) for every gene pairs

To find genes that participate in the same cellular activities as seed genes but not annotated yet, we use Mutual Information(MI) of mRNA expression data. Mutual

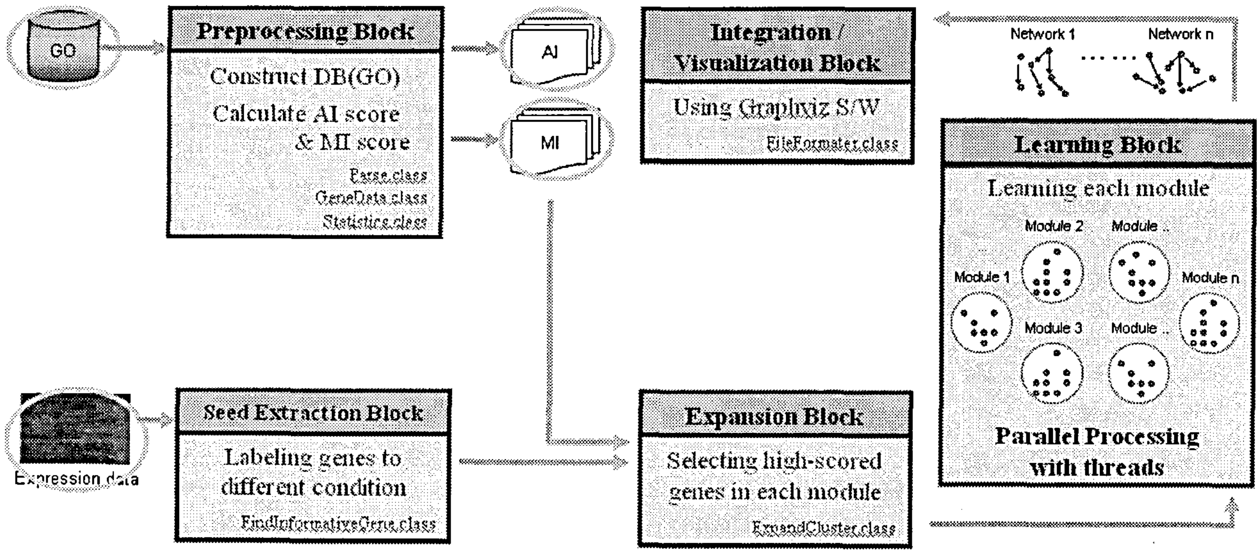

Figure 1: Overview of Modularized Network Learning

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. A Mutual Information (MI) score of two genes, g_i and g_j is defined as follows.

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

x_i : a discretized expression value of a gene g_i
x_j : a discretized expression value of a gene g_j

2.2.2 Seed Extraction Block

a. Processing Microarray Data as Input

Microarray-data that is plain-text file with 'tab' delimitation with rows of genes and columns of samples is accepted. Then, the data is treated by smoothing, imputation, and discretization.

b. Extracting Seed Gene by Distinctiveness

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 Gasch et al. consists of 173 experiments consecutively measured in 16 different stress conditions; every stress condition consists of several experiments. Distinctiveness D of a gene i in one condition c is based on Sharmir's measure and defined as follows:

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

m_{ci} is the mean expression value of gene i during experiments belonging to the same condition c, while $m_{\neg ci}$ is the mean expression value of gene i during experiments not belonging to a condition c. s_{ci} and $s_{\neg ci}$ are the standard deviations corresponding to the former and the latter cases, respectively. Intuitively, a large difference between mci and $m_{\neg ci}$ indicates that gene i shows a distinctive expression pattern in a condition c compared to all the other conditions. The smaller s_{ci} and $s_{\neg ci}$, 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.

2.2.3 Seed Expansion Block

a. Expanding Seed Gene by AI and MI

Selected seed genes are expanded into modules by including closely associated genes based on Annotation Information (AI) and Mutual Information (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 of the almost same members in them.

2.2.4 Learning Block

a. Learning of subnetworks for individual modules

To learn Bayesian networks for individual modules, we apply a Bayesian network learning technique, which is based on hill climbing, sparse candidates 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) 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:

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

S = a set of N best networks

2.2.5 Integrating and Visualization Block

a. 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). 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.

b. Visualization of whole network

Final result of the network is visualized by External software "GraphViz." Partial example of the network is shown in Figure 2.

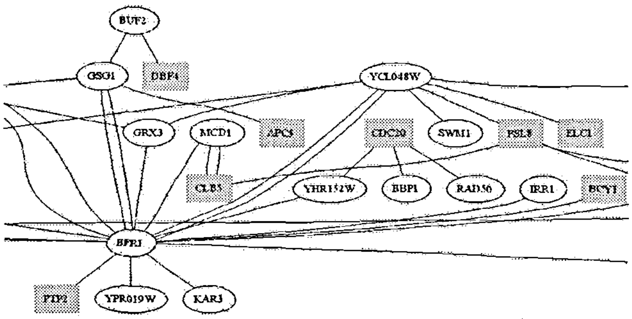

Figure 2: Example of Partial Graphs in S. Cerevisiae

3 CONCLUSION

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. Those separately inferred subnetworks over each module are integrated into final global networks through common intermediary genes.

We have packaged the algorithm of MONET into the system that can be utilized on actual, biological fields. This system can provide us with global picture of actively responding biological processes as well as detailed look of relationship among genes with reduced false positive even though the number of expression profiles is not enough relative to the number of genes. In the next, we need to search optimal parameters with respect to each species, and integrate visualization block.

REFERENCES

[1] S. Liang, S. Fuhrman, and R. Somogyi. Reveal, a general reverse engineering algorithm for inference of genetic network architectures. In Proc. of Pacific Symposium on Biocomputing, pages 18–29. PSB, 1998.

[2] N. Friedman et al. Using bayesian networks to analyze expression data. Journal of Computational Biology, 7:601–620, 2000.

[3] T. Akutsu, S. Miyano, and S. Kuhara. Algorithms for inferring qualitative models of biological networks. In Proc. of Pacific Symposium on Biocomputing, pages 290–301. PSB, 2000.

[4] Y. Tamada et al. Estimating gene networks from gene expression data by combining bayesian network model with promoter element detection. Bioinformatics, 19(2):ii227–ii236, 2003.

[5] E. Segal et al. Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. Nature Genetics, 34(2):166–176, 2003.

[6] R. Neapolitan. Learning Bayesian Networks. Prentice Hall, 2004.

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

[8] C. Yoo, V. Thorsson, and G. Cooper. 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, pages 498–509. PSB, 2002.

[9] A. Hartemink et al. Combining location and expression data for principled discovery of genetic regulatory network models. In Proc. of Pacific Symposium on Biocomputing. PSB, 2002.

[10] N. Friedman et al. Learning bayesian network structure from massive datasets: The "sparse candidate" algorithm. In Proc. of Fifteenth Conference on Uncertainty in Artificial Intelligence. Fifteenth Conference on Uncertainty in Artificial Intelligence, UAI, 1999.

[11] M. Fashing et al. A clustering algorithm explicitly designed to produce priors for bayesian network discovery from whole-genome expression level data. Spring 2002.

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

[13] E. Segal et al. Genome-wide discovery of transcriptional modules from dna sequence and gene expression. Bioinformatics, 19(Suppl.1):i273–i282, 2003.

[14] Calabretta et al. 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, pages 275–284, 1998.

[15] Jennifer Hallinan. Gene duplication and hierarchical modularity in intracellular interaction networks. BioSystems, 74:51–62, 2004.

[16] Amy Hin Yan Tong et al. Global mapping of the yeast genetic interaction network. Science, 303:808–813, 2004.

[17] The Gene Ontology Consortium. Creating the gene ontology resource: design and implementation. Genome Research, 11(8):1425–1433, 2001.

[18] Gash et al. Genomic expression program in the response of yeast cells to environmental changes. Molecular Biology of Cell, 11:4241–4257, 2000.

[19] R. Shamir. Lecture note: Analysis of gene expression data. Tel. Aviv. University, 2002.

[20] H. Mewes et al. Mips: A database for protein sequences, homology data and yeast genome information. Nucleic Acid Research, 25:28–30, 1997.

[21] P. Lord et al. Investigating semantic similarity measures across the gene ontology: the relationship between sequence and annotation. Bioinformatics, 19(10):1275–1283, 2003.

[22] P. Resnik. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. Journal of Arti-ficial Intelligence Research, 11:95–130, 1999.

[23] I. Kohane, A. Kho, and A. Butte. Microarrays for an integrative genomics. MIT press, 2003.

[24] J. Michael Cherry et al. Sgd: Saccharomyces genome database. Nucleic Acid Research, 26(1):73–79, 1998.

[25] W. Lam and F. Bacchus. Learning bayesian belief networks: an approach based on the mdl principle. Computational Intelligence, 10:269–293, July 1994.

[26] J.L. Schafer. Analysis of incomplete multivariate data, 1997.

[27] A. Kwon et al. Inference of transcriptional regulation relationships from gene expression data. Bioinformatics, 19(8):905–912, 2003.