BIOINFORMATICS ORIGINAL PAPER Vol. 22 no. 19 2006, pages 2413–2420
doi:10.1093/bioinformatics/btl396

Gene expression

Inferring gene regulatory networks from multiple microarray datasets

Yong Wang[1,2], Trupti Joshi[3], Xiang-Sun Zhang[2,*], Dong Xu[3,*] and Luonan Chen[1,4,*]

[1]Department of Electrical Engineering and Electronics, Osaka Sangyo University, Osaka 574-8530, Japan, [2]Academy of Mathematics and Systems Science, CAS, Beijing 100080, China, [3]Computer Science Department and Christopher S. Bond Life Sciences Center, University of Missouri, Columbia, MO 65211, USA and [4]Institute of Systems Biology, Shanghai University, Shanghai 200444, China

Received on March 5, 2006; revised on June 25, 2006; accepted on July 17, 2006
Advance Access publication July 24, 2006
Associate Editor: Jonathan Wren

ABSTRACT

Motivation: Microarray gene expression data has increasingly become the common data source that can provide insights into biological processes at a system-wide level. One of the major problems with microarrays is that a dataset consists of relatively few time points with respect to a large number of genes, which makes the problem of inferring gene regulatory network an ill-posed one. On the other hand, gene expression data generated by different groups worldwide are increasingly accumulated on many species and can be accessed from public databases or individual websites, although each experiment has only a limited number of time-points.

Results: This paper proposes a novel method to combine multiple time-course microarray datasets from different conditions for inferring gene regulatory networks. The proposed method is called GNR (Gene Network Reconstruction tool) which is based on linear programming and a decomposition procedure. The method theoretically ensures the derivation of the most consistent network structure with respect to all of the datasets, thereby not only significantly alleviating the problem of data scarcity but also remarkably improving the prediction reliability. We tested GNR using both simulated data and experimental data in yeast and *Arabidopsis*. The result demonstrates the effectiveness of GNR in terms of predicting new gene regulatory relationship in yeast and *Arabidopsis*.

Availability: The software is available from http://zhangorup.aporc.org/bioinfo/grninfer/, http://digbio.missouri.edu/grninfer/ and http://intelligent.eic.osaka-sandai.ac.jp or upon request from the authors.

Contact: chen@eic.osaka-sandai.ac.jp, xudong@missouri.edu, zxs@amt.ac.cn

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 INTRODUCTION

Microarray technologies have produced tremendous amounts of gene expression data (van Someren *et al*., 2001; Hughes *et al*., 2000). Mining these data to understand gene expression and regulation represents a major challenge for bioinformatics. A major focus on microarray data analysis is the reconstruction of gene regulatory network (GN), which aims to find the underlying network of gene-gene interactions from the measured dataset of gene expression (Hartemink, 2005; Basso *et al*., 2005; Levine and Davidson, 2005; Akutsu *et al*., 2000; H (2000)). A wide variety of approaches have been proposed to infer gene regulatory networks from time-course data (Holter *et al*., 2001; Tegner *et al*., 2003; Dewey and Galas, 2001), such as discrete models of Boolean networks and Bayesian networks (Husmeier, 2003; Rangel *et al*., 2004; Beal *et al*., 2005), and continuous models of neural networks, difference equations (van Someren *et al*., 2001) and differential equations (Chen and Aihara, 2001, 2002).

Since a typical gene expression dataset consists of relatively few time points (often <20) with respect to a large number of genes (generally in thousands), a major difficulty of GN inference for all methods is scarcity of time-course data or the so-called dimensionality problem (D'haeseleer *et al*., 2000; Zak *et al*., 2003; van Someren *et al*., 2001). In other words, the number of genes far exceeds the number of time points for which data are available, making the problem of determining GN structure an ill-posed one. Current methods generally use a single set of time-course data under a specific experimental condition, and hence often fail in using experimental data to construct GN accurately. On the other hand, gene expression data generated by different groups worldwide are increasingly accumulated on many species and can be accessed from public databases or individual websites, although each experiment has only a limited number of time-points. For example, in the GEO database (http://www.ncbi.nlm.nih.gov/geo/), currently there are 241 microarray datasets for human alone. If such large amounts of data from different experiments are combined and further exploited in an integrative and systematic manner, the scarcity of data can be greatly alleviated and a more accurate reconstruction of GN can be expected. It is worth mentioning that simply arranging multiple time-course datasets into a single time-course dataset is inappropriate for GN inference owing to data normalization issues and lack of temporal relationships among these datasets. Hence, current GN inference methods typically cannot handle multiple sets of data.

In addition to the dimensionality problem of data, another problem for the conventional approaches is that the derived gene networks often have densely connected gene regulatory relationships among nodes, which are not biologically plausible. A biological gene network is expected to be sparse (Gardner and Faith, 2005; Yeung *et al*., 2002), which should also be reflected in the procedure of the network reconstruction.

*To whom correspondence should be addressed.

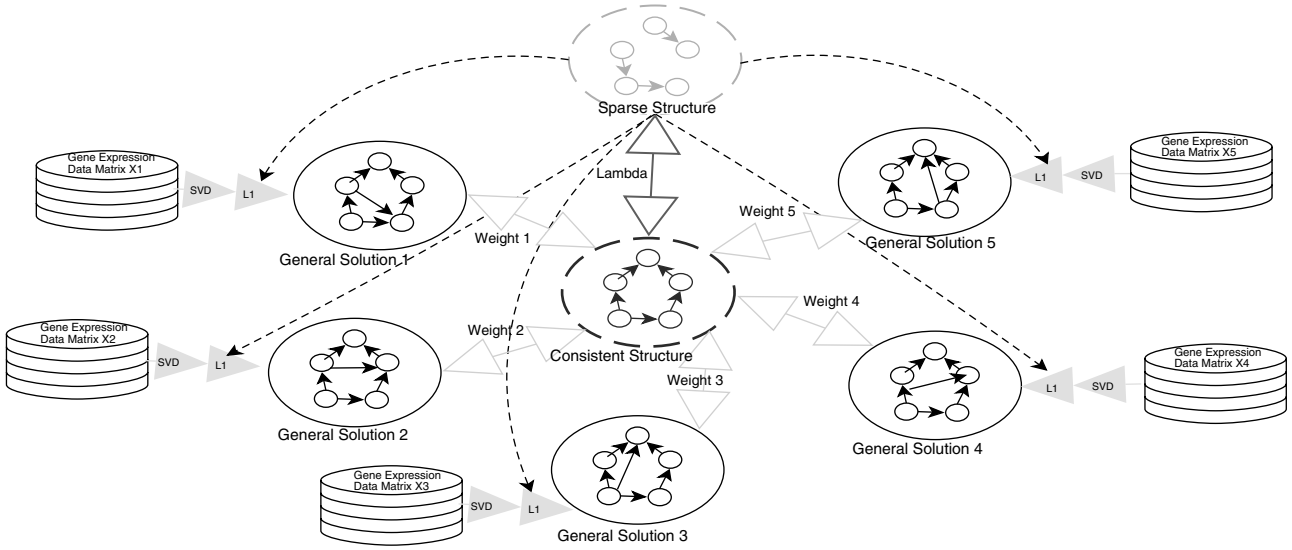

Fig. 1. Schematic of GNR.

This paper proposes a novel method to combine a wide variety of microarray datasets from different experiments (different environmental conditions or perturbations) for inferring GN with the consideration of sparsity of connections. GNR (Gene Network Reconstruction tool), based on LP (linear programming) and a decomposition procedure, is developed by exploiting the general solution form of arbitrary connectivity matrix for GN. The proposed method GNR theoretically ensures the derivation of the most consistent or invariant network structure with respect to all the used datasets, thereby not only significantly alleviating the problem of data scarcity but also remarkably improving the reliability. Specifically, inferring GN is formulated as an optimization problem with an objective function of forced matching and sparsity terms, so that a consistent and sparse structure that is also considered to be biologically plausible can be expected. An efficient algorithm has been developed to solve such a large-scale LP in an iterative manner. Both simulated examples and experimental data are used to demonstrate the effectiveness of GNR, which also leads to predictions of new gene regulation relationships for yeast and *Arabidopsis*. GNR is implemented in the Fortran programming language and the software is available from http://zhangorup.aporc.org/bioinfo/grninfer/, http://digbio.missouri.edu/grninfer/ and http://intelligent.eic.osaka-sandai.ac.jp or upon request from the authors.

2 METHODS

Figure 1 illustrates the schematic of the proposed method. In this section, we first describe a GN as a set of differential equations, and then derive a special solution of the GN based on singular value decomposition (SVD) for a single dataset (time-course data). By constructing the general solution of the GN for each single dataset, we formulate the GN reconstruction problem as an optimization problem which is to find the most consistent network structure with respect to all the used datasets. The optimal solution can be viewed as a special solution for the multiple datasets with the minimal connections or edges. We show that such an optimization problem is equivalent to a linear programming, and an efficient algorithm is developed to solve such an LP based on the decomposition technique.

2.1 Gene regulatory network

Generally, a genetic network can be expressed by a set of non-linear differential equations with each gene expression level as variables

$$\dot{x}(t) = f(x(t)), \tag{1}$$

where $x(t) = (x_1(t), \ldots, x_n(t))^{\mathrm{T}} \in R^n$, and $f = (f_1, \ldots, f_n)^{\mathrm{T}} : R^n \mapsto R^n$. $x_i(t)$ is the expression level (mRNA concentrations) of gene i at time instance t. Assume that there are a total of m time points for a given experimental condition from microarray, i.e. $t_1, \ldots, t_m$. f_i is a C^1 class non-linear function.

Although gene regulations are often non-linear, most of the existing approaches for GN inference use linear or additive models owing to unclear structures of biological systems and scarcity of data (D'Haeseleer *et al*., 1999; Gustafsson *et al*., 2005). From the viewpoint of dynamical systems, linear equations can at least capture the main features of the network or the function, in particular around a specific state of the system. The linear form of Equation (1) with appropriate normalization is

$$\dot{x}(t) = Jx(t) + b(t), \quad t = t_1, \ldots, t_m, \tag{2}$$

where $J = (J_{ij})_{n\times n} = \partial f(x)/\partial x$ is an $n \times n$ Jacobian matrix or connectivity matrix, and $b = (b_1, \ldots, b_n)^{\mathrm{T}} \in R^n$ is a vector representing the external stimuli or environment conditions, which is set to zero when there is no external input.

2.2 General solution for a single dataset

To overcome the difficulty because of scarce data, many techniques, such as clustering of genes, SVD, interpolation of data (van Someren *et al*., 2001) have been developed. We first adopt the SVD technique to derive a particular solution and further the general solution of Equation (2), using a single time-course dataset. By rewriting Equation (2), we have

$$\dot{X} = JX + B, \tag{3}$$

where $X = (x(t_1), \ldots, x(t_m))$, $B = (b(t_1), \ldots, b(t_m))$ and $\dot{X} = (\dot{x}(t_1), \ldots, \dot{x}(t_m))$ are all $n \times m$ matrices with $\dot{x}_i(t_j) = [x_i(t_{j+1}) - x_i(t_j)]/[t_{j+1} - t_j]$ for $i = 1, \ldots, n$; $j = 1, \ldots, m$. By adopting SVD, i.e. $(X^{\mathrm{T}})_{m\times n} = U_{m\times n} E_{n\times n} V^{\mathrm{T}}_{n\times n}$, where U is a unitary $m \times n$ matrix of left eigenvectors, $E = \mathrm{diag}(e_1, \ldots, e_n)$ is a diagonal $n \times n$ matrix containing the n eigenvalues and V^{T} is the transpose of a unitary $n \times n$ matrix of right eigenvectors. Without loss of generality, let all non-zero elements of e_k

be listed at the end, i.e. $e_1 = \cdots = e_l = 0$ and $e_{l+1}, \ldots, e_n \neq 0$. Then we can have a particular solution with the smallest L_2 norm for the connectivity matrix $\hat{\boldsymbol{J}} = (\hat{\boldsymbol{J}}_{ij})_{n\times n}$ as

$$\hat{\boldsymbol{J}} = (\dot{\boldsymbol{X}} - B)UE^{-1}V^{\mathrm{T}} \tag{4}$$

where $E^{-1} = \mathrm{diag}(1/e_i)$ and $1/e_i$ is set to be zero if $e_i = 0$. Thus, the network family, or the general solution of the connectivity matrix $J = (J_{ij})_{n\times n}$ is

$$J = (\dot{\boldsymbol{X}} - B)UE^{-1}V^{\mathrm{T}} + YV^{\mathrm{T}} = \hat{\boldsymbol{J}} + YV^{\mathrm{T}} \tag{5}$$

$Y = (y_{ij})$ is an $n \times n$ matrix, where y_{ij} is zero if $e_j \neq 0$ and is otherwise an arbitrary scalar coefficient. Solutions of (5) represent all of the possible networks that are consistent with the single microarray dataset, depending on arbitrary Y. Notice that $m + 1$ points are required in (5) owing to the estimation of $\dot{\boldsymbol{X}}$.

2.3 Special solution with minimal connections for multiple datasets

Assume that there are multiple microarray datasets for one organism, each of which corresponds to its own general solution of (5). Each time-course dataset may be measured under various environments or stimuli by different labs. Specifically, there are N datasets, and we can infer N networks respectively as

$$J^k = (\dot{\boldsymbol{X}}_k - B_k)U_kE_k^{-1}V_k^{\mathrm{T}} + Y^kV_{\mathrm{k}}^{\mathrm{T}} = \hat{\boldsymbol{J}}^k + Y^kV_k^{\mathrm{T}}, \tag{6}$$

where the subscript $k = 1, \ldots, N$ is the index of the dataset-k. Note that without normalization, J^k for each dataset is actually a normalized matrix even for different experiments with different time intervals due to the form of (4).

Next, we will find the most consistent network structure $J = (J_{ij})_{n\times n}$ for all $k = 1, \ldots, N$ of (6), with consideration of sparse structure, as illustrated in Figure 1. Mathematically, the problem is formulated as

$$\min_{Y,J} \sum_{k=1}^{N}\sum_{i=1}^{n}\sum_{j=1}^{n}[\boldsymbol{\omega}^k|J_{ij} - J_{ij}^k| + \lambda|J_{ij}|], \tag{7}$$

where J_{ij}^k is the function of Y^k according to (6), and $Y = (Y^1, \ldots, Y^N)$. The variables are Y and J. The first term is the matching term which forces the matching of J and J^k, whereas the second term is the sparsity term which forces J sparse owing to L_1 norm. λ is a positive parameter, which balances the matching and sparsity terms in the objective function. The variables in (7) are J_{ij} and all of non-zero y_{ij}^k. $\boldsymbol{\omega}^k$ is a positive weight coefficient for the dataset-k with $\sum_{k=1}^{N}\boldsymbol{\omega}^k = 1$. Since different datasets may have different qualities (e.g. different technologies, number of repeats in measurements, etc.), a weight coefficient is used to represent the reliability of each dataset. Assume that the number of the repeated experiments for the dataset-k is N_k by using the same type of microarray. Then $\boldsymbol{\omega}^k$ can be set as

$$\boldsymbol{\omega}^k = \frac{N_k}{\sum_{i=1}^{N}N_i}. \tag{8}$$

The optimization problem for (7) is a mathematical programming problem with positive combination of L_1 norm of variables, which can be transformed into a linear programming problem through a well-known procedure and solved by a simple iterative procedure. Owing to L_1 norm, generally the optimal solution of (7) has the property with the zeros for $|J_{ij} - J_{ij}^k|$ and $|J_{ij}|$ as many as possible, which exactly serves our purpose, i.e. consistent and sparse structure.

2.3.1 Decomposition and algorithm Clearly when J is fixed, the original problem of (7) can be divided into N independent subproblems. We decompose (7) into the following form.

$$\min_{J}\ \min_{Y}\sum_{k=1}^{N}\sum_{i=1}^{n}\sum_{j=1}^{n}[\boldsymbol{\omega}^k|J_{ij} - J_{ij}^k| + \lambda|J_{ij}|] \tag{9}$$

Since (9) is a large-scale linear programming (LP) problem owing to a large number of variables, we adopt an iterative technique to solve (9). Specifically, first we fix J to solve N small-size matching subproblems Y, and then update J based on the results of Y for N subproblems. Such iteration continues until converged.

Therefore, we have the following algorithm for deriving gene network.

- **STEP-0**: Initialization. Obtain all of the particular solution $\hat{\boldsymbol{J}}^k$ by SVD from (4), and $\boldsymbol{\omega}^k$ from (8). Set initial value $J_{ij}(0) = 0$, $Y_{ij}^k(0) = 0$ and $J_{ij}^k(0) = \hat{\boldsymbol{J}}^k$, and positive values λ, ε. Let iteration index be q and set $q = 1$.
- **STEP-1**: Set $J^k(q) = J^k(q-1) + Y^k(q)V_k^{\mathrm{T}}$ and solve $y_{ij}^k(q)$ at iteration q by LP for each subproblem from (9) with $J\,(q-1)$ fixed, i.e. solve $Y^k(q) = (y_{ij}^k(q))_{m\times m}$ of the following subproblem for $k = 1, \ldots, N$ with $J(q-1)$ given

$$\min_{Y^k(q)}\sum_{i=1}^{n}\sum_{j=1}^{n}|J_{ij}(q-1) - J_{ij}^k(q)| \tag{10}$$

  Note that $y_{ij}^k(q) = 0$ if $j > l_k$ according to (5).
- **STEP-2**: Solving $J_{ij}(q)$ at iteration q by LP with all of $y_{ij}^k(q)$ given, i.e. solve $J(q)$ of the following problem with all of $J^k(q)$ fixed.

$$\min_{J(q)}\sum_{k=1}^{N}\sum_{i=1}^{n}\sum_{j=1}^{n}[\boldsymbol{\omega}^k|J_{ij}(q) - J_{ij}^k(q)| + \lambda|J_{ij}(q)|] \tag{11}$$

  The detail procedures of solving (10) and (11) are described in Supporting Material.
- **STEP-3**: If J is converged, i.e. $||J(q) - J(q-1)|| < \varepsilon$, then terminate the computation. Otherwise, go to STEP-1 by $q \rightarrow q + 1$.

Although the solution may depend on λ, it is a single parameter which can be tuned in a relatively easy manner or be simply tested for a range of its value. A flowchart of the algorithm is illustrated in Supplementary Material. The non-linear network (e.g. with quadratic form) can also be derived with similar form of (7) in a self-consistent way.

2.3.2 Confidence evaluation Let the optimal solution of (7) be J^* and Y^{*k}. Then, the variances v_{ij} and deviation σ_{ij} of each element J_{ij} for J can be easily estimated by

$$v_{ij} = \frac{\sum_{k=1}^{N}\boldsymbol{\omega}^k[J_{ij}^* - J_{ij}^k(Y^{*k})]^2}{N} \tag{12}$$

$$\sigma_{ij} = \sqrt{v_{ij}} \tag{13}$$

By computing their average, we have

$$\bar{\sigma} = \sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\sigma_{ij}}{n^2} \tag{14}$$

In addition, the proposed approach can be further improved by combining with other methods, such as, the data expanding technique developed by van Someren *et al*. (2001).

3 RESULTS

In this section, we first report on several numerical tests that we have designed to benchmark GNR using multiple simulated datasets. Then we will describe the GN inference using yeast and *Arabidopsis* microarray gene expression data. As analysed in Methods, when a single time-course dataset is adopted, GNR is similar to the method of Yeung *et al*. (2002), which can recover the network connectivity from gene expression measurements in the presence of noise by SVD and regression. For a single time-course dataset, it is easy to show that the smallest number of time points needed is $O(\log n)$ to reconstruct the $n \times n$ connectivity matrix for an n-gene

network (Yeung *et al*., 2002). When adopting multiple datasets, we can further infer the most consistent network structure with respect to all the datasets in a more accurate and robust manner.

3.1 Simulated data

The first example is a small simulated network with five genes governed by

$$\begin{aligned}\dot{x}_1(t) &= -2x_2(t) + \xi_1(t),\\ \dot{x}_2(t) &= -x_3(t) + \xi_2(t),\\ \dot{x}_3(t) &= -3x_4(t) + \xi_3(t),\\ \dot{x}_4(t) &= -1.5x_5(t) + \xi_4(t),\\ \dot{x}_5(t) &= 2x_1(t) + \xi_5(t),\end{aligned}$$

where x_i reflects the expression level of the gene-i and $\xi_i(t)$ represents noise for $i = 1, 2, 3, 4, 5$. Clearly, the system is a negative gene regulation loop with genes 2, 3, 4, 5 repressing genes 1, 2, 3, 4 respectively, and with gene 1 in turn enhancing gene 5.

To test GNR, we randomly choose the initial condition of the system and take several points of x as a measured time-course dataset. With three different initial conditions, we obtained three different datasets with 4, 4 and 3 time points respectively, and applied GNR to reconstruct the connectivity matrix or the Jacobian matrix J. To measure the discrepancies between the true network and the inferred network with n genes, we adopt the simple criterion in Yeung *et al*. (2002) as E_0 to assess the basic recovering ability:

$$E_0 := \sum_{i=1}^{n}\sum_{j=1}^{n} e_{ij}, \tag{15}$$

where e_{ij} takes 1 if $||J_{ij}^T - J_{ij}^R|| > \delta$, otherwise 0. δ is a prescribed small value for error tolerance related to noise level of the system. J_{ij}^T and J_{ij}^R are interaction strength from gene-j to gene-i for the true and inferred networks, respectively.

Furthermore to depict the accuracy or correctness of GNR, we introduce the following two criteria E_1 and E_2 as

$$E_1 := \sum_{i=1}^{n}\sum_{j=1}^{n} |J_{ij}^T - J_{ij}^R| \tag{16}$$

$$E_2 := \sum_{i=1}^{n}\sum_{j=1}^{n} (J_{ij}^T - J_{ij}^R)^2 \tag{17}$$

which are L_1 norm and L_2 norm errors respectively for all of interaction strengths.

The numerical results are depicted in Figures 2 and 3, which show the reconstructed networks without and with noises respectively. As indicated in Figure 2, clearly the more the datasets, the more accurate the inferred network. When using one dataset (Fig. 2b), it contains a wrong relation between x_5 and x_3. As two datasets are used, the topology of the network becomes correct (Fig. 2c). After using all three datasets, the predicted connectivity matrix, which represents the strengths among gene interactions, is very close to the true one (Fig. 2c). Such results imply that GNR is able to infer the solution of the highly under-determined problem in an accurate manner when a sufficient number of datasets (or experiments) are available even though each dataset has only a few time points and starts from different initial conditions. In GNR, we also introduce a scalar parameter λ to control the sparsity of the inferred network (see Methods for details). When there are

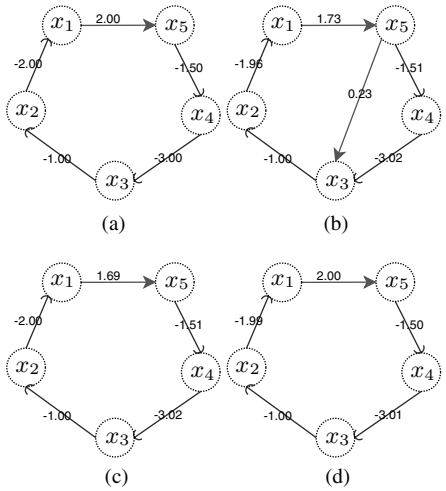

Fig. 2. The simulated example with $\lambda = 0$ and without noise. Arrows and arcs denote activation and repression, respectively. (**a**) The true network. (**b**) Using one dataset. (**c**) Using two datasets. (**d**) Using three datasets.

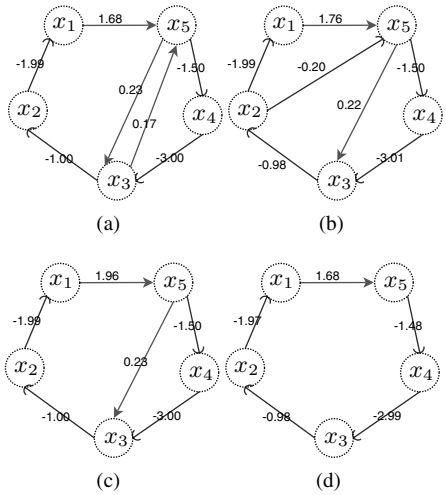

Fig. 3. The simulated example with noise. Arrows and arcs denote activation and repression respectively. (**a**) Using one dataset with $\lambda = 0.0$. (**b**) Using two datasets with $\lambda = 0.0$. (**c**) Using three datasets with $\lambda = 0.0$. (**d**) Using three datasets with $\lambda = 0.3$.

multiple solutions (which are typical) due to the under-determined nature, GNR prefers to infer a network with a sparse structure.

Figure 3 shows the results when noises are added to the dynamics. As indicated in Tu *et al*. (2002), the distribution function of the noise in microarray is more like a Gaussian distribution. Therefore we set all of noises $\xi_i(t)$, $i = 1, 2, 3, 4, 5$ obeying normal distribution in the simulated example. With gradual increase of noise level to $N(0, 0.005)$, the network eventually cannot be correctly inferred even using all three datasets (Fig. 3c) due to the effect of noises. For such an under-determined case, GNR can reconstruct the network by an additional constraint of sparsity, i.e. introducing a positive parameter λ, as shown in Figure 3d. With such a constraint, generally there is a better chance to construct a biologically plausible structure (Yeung *et al*., 2002) but at the expense of accuracy of

Table 1. Accuracies for different error criteria and confidence evaluation

	λ	E_0	E_1	E_2	$\bar{\sigma}$
Without noise					
One dataset	0.0	1	1.38	0.22	0.4145
Two datasets	0.0	0	1.16	0.21	0.0075
Three datasets	0.0	0	0.93	0.15	0.0032
With noise					
One dataset	0.0	2	1.42	0.27	0.4105
Two datasets	0.0	2	1.36	0.21	0.0131
Three datasets	0.0	1	0.93	0.13	0.0035
Three datasets	0.3	0	0.93	0.19	0.0197

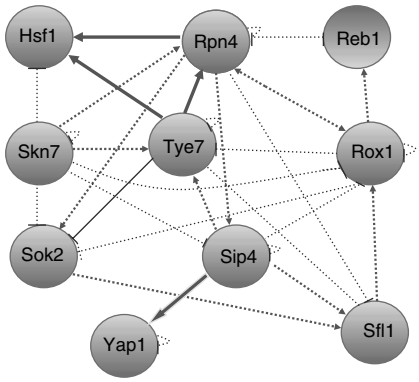

Fig. 4. Regulatory network reconstruction for set of 10 transcription factors for heat shock response microarray data in yeast. Activation is shown in red and repression in blue arrows. The confirmed edges are shown in bold arrows, while the potential edge is shown in yellow-red arrow indicating activation.

interaction strengths. We have also tested for a nonlinear gene network by replacing all linear terms into quadratic terms. As demonstrated in Supplementary Material (in particular Supplementary Figure A2 and Table A1), comparing with the linear and noise cases, the link strengths of reconstructed networks have certain errors. Nevertheless, the topology of the network can be correctly inferred using all three datasets (Supplementary Figure A2c). Table 1 shows the accuracies of different error criteria, i.e. E_0, E_1 and E_2 in the two cases without noise and with noise obeying $N(0, 0.005)$ normal distribution, which indicate that adding datasets improves the accuracy of the network reconstruction, e.g. the more the datasets, the smaller are the E_0, E_1 and E_2 values. This table also implies that a solution of GNR is a balance between the topology reconstruction (evaluated mainly by E_0) and the accuracy of interaction strength (evaluated mainly by E_1 or E_2). The trade-off between E_0 and E_1 (or E_2) can be controlled by the parameter λ. The deviation $\bar{\sigma}$ in Table 1 is introduced to evaluate the confidence of the inferred network (see Methods for details). The tendency of $\bar{\sigma}$ also indicates that adding datasets improves the confidence of the network reconstruction.

We also consider a large system to calibrate the proposed reverse engineering scheme. The results are listed in the Supplementary Material, which further confirm the effectiveness of GNR.

3.2 Application to experimental data

We applied GNR to experimental data. To ensure high quality of the data, we only used whole genome Affymetrix chips microarray experimental data, instead of any oligo or cDNA array data.

3.2.1 Heat-shock response data for yeast We first test GNR using a small number of genes. We created an input dataset for 10 transcription factors related to heat-shock response in yeast *Saccharomyces cerevisiae*. Out of the 10 transcription factors 2 (Hsf1p and Skn7p) are known to be directly involved in heat shock response. Hsf1p and Skn7p each are known to regulate 4 other transcription factors among the 10. This information was obtained from YEASTRACT (http://www.yeastract.com/index.php). For the 10 transcription factors, we used 4 microarray datasets at the Stanford Microarray Database (http://smd.stanford.edu/) (y11, y14, y16:57–60, y16:109–112, with 7, 5, 5, 4 time points, respectively) for gene expression under heat shock conditions. We applied GNR to this dataset. As shown in Figure 4, the prediction succeeded in reconstructing four edges of the network with documented known regulation and 1 edge with documented potential regulation.

3.2.2 Cell cycle data for yeast We tested GNR using the experimental data for cell cycle studies in *Saccharomyces cerevisiae* obtained from the Stanford Microarray Database (http://smd.stanford.edu/). We generated four datasets with different conditions. Supplementary Table A5 lists the experimental conditions and time points used for analysis. Among all the yeast genes, 140 of them have change of 2-fold up or down in at least 20% of the expression level across all datasets.

Application of GNR to the 140 differentially expressed genes of the four datasets generated consistent subnetworks with 64 links, 431 links, etc. depending on the scalar parameter used to control the sparsity or consistency of the subnetwork. Figure 5 shows a representation of the 64-link GN model. Figure 5a shows YGP1 in the center which is a cell wall-related secretory glycoprotein and induced by nutrient deprivation-associated growth arrest and upon entry into the stationary phase (Destruelle *et al*., 1994). In the predicted model, YGP1 activates three genes, i.e. DSE2, PIR3 and FET3. Both DSE2 and PIR3 relate to cell wall organization and biogenesis (Doolin *et al*., 2001; Mrsa and Tanner, 1999), whose activations may follow YGP1 at the entry of the stationary phase. Among the genes that YGP1 suppresses in the model, it is known that HLR1 suppresses the cell wall phenotypes (Alonso-Monge *et al*., 2001). Suppressing HLR1 by YGP1 is equivalent to enhance cell wall development, which is consistent to the activation of DSE2 and PIR3. TFA2 is TFIIE small subunit, involved in RNA polymerase II transcription initiation (Kornberg, 1998). In addition to the genes in Figure 5, other genes in the network show negative self regulation (data not shown).

3.2.3 Stress response data for Arabidopsis We also applied our method in studying stress response in *Arabidopsis thaliana*. We used whole genome Affymetrix chips microarray experimental data for Arabidopsis thaliana from the ATGenExpress database at The *Arabidopsis* Information Resources (TAIR) (http://www.arabidopsis.org/). We applied nine datasets related to the stress responses, each with six or more time points and each for the

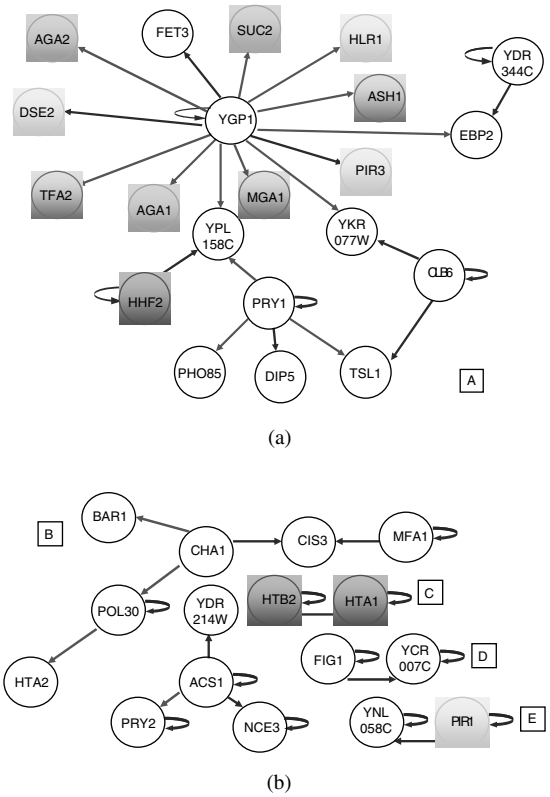

Fig. 5. Partial representation (with two connected sub-networks) of the 64-link inferred network in yeast based on cell cycle microarray experimental datasets. The isolated genes without interaction with others are not shown. The red arrows in the figure indicate repression while the blue arrows indicate activation. The circles in the same color indicate the same biological function.

root and shoot experiments. Supplementary Table A6 lists the experimental details and the time points used. We used the log ratios of the expression values for a treatment condition against the mock condition. We narrowed down the list of genes to 226 genes for the root experiments and 246 genes for the shoot experiments based on two-fold change (either up or down) in at least 70% of the ratios of a gene. This list represents the most statistically significant genes differentially expressed under stress in root and shoot based on all experiments.

GNR was applied to the 226 genes with the above nine datasets. Using different thresholds, we can predict various networks with different edge density, which are consistent with respect to all datasets. Figure 6 represents a 35-link sub-network in shoots. The network shows that the genes AT1G56600 and ATERF6 (AT4G17490) control the neighboring genes by either activating or suppressing them. We found that our network relates to some knowledge while predicting novel regulations. ATERF6 is a member of the ERF (ethylene response factor) subfamily B-3 of ERF/AP2 transcription factor family (Fujimoto *et al*., 2000). It is predicted to activate three genes encoding known or putative transcription factors, i.e. AT2G12940, AT3G49760 and AT2G40750. AT2G12940 is similar to transcription factor VSF-1; AT3G49760 is a Bzip transcription factor; AT2G40750 is a member of WRKY transcription factor family. Other genes have functions related to stress response. AT1G52560 is a heat shock protein. AT1G36030 encodes a member of the F-box family, whose members involved in regulating diverse cellular processes including cell cycle transition, transcriptional regulation and signal transduction.

4 DISCUSSION

Microarray gene expression data has increasingly become the common data source that can provide insights into biological processes at a system-wide level. As indicated in Soinov (2003), although a large amounts of data are increasingly accumulated, one of the major problems with microarrays is that data often come from different platforms, laboratories, etc. It is often difficult to compare or combine results of experiments done by different research groups for biological inference. In contrast to the conventional methods which require more time points in a single dataset to infer more accurate network owing to the dimensionality problem, the main contribution of this paper is that we developed a methodology to reconstruct GN using multiple datasets from different sources without normalization among the datasets. In other words, we provide a general framework to handle the microarray data by fully exploiting all available microarray data for a given species, so as to alleviate the problem of dimensionality or data scarcity. As a byproduct of the new method, it provides a new way to compare hypotheses generated from different datasets, and also a new way to derive a common substructure not from network alignment but from the raw microarray datasets. In particular, it is very effective to find an invariant structure when multiple datasets with different conditions or perturbations are used.

We have tested our approach on both simulated problems and experimental biological data, which verified the efficiency and effectiveness of the algorithm. Depending on the trade-off parameter λ, we can derive either a global structure with dense connection for a small λ or local substructure with sparse connection for a large λ. Furthermore, the role of parameter λ in the inference algorithm is discussed and tested by comparing the inferred network structures for different λs. Also we discuss how to specify the proper value and the searching strategy in the parameter space of λ (see the details in Supplementary Material).

There is an important assumption for the proposed method in this paper, i.e. the structure of the regulatory network is stationary, and does not 'rewire' under the environmental conditions for those different datasets. This means that the change of environmental conditions alters the level of gene expression instead of the network structure. Another assumption is that high resolution time-course microarray datasets are required so as to accurately infer the network structure because a genetic network is expressed by a set of differential equations with each gene expression level as a variable shown in Equation (1). Here high resolution data mean high quality time-course microarray data which are expected to capture the dynamic behavior of the gene regulatory network.

The linear differential equation model in this paper is used to identify gene regulation between RNA transcripts (Gardner and Faith, 2005). An advantage of such a strategy is that the model can implicitly capture regulatory mechanisms at the protein and metabolite levels that are not physically measured. That is, it is not restricted to describe only transcription factor/DNA interactions. By construction, the inferred model may accurately reflect a physical interaction if the regulator transcripts encode the transcription factors that directly regulate transcription. On the other

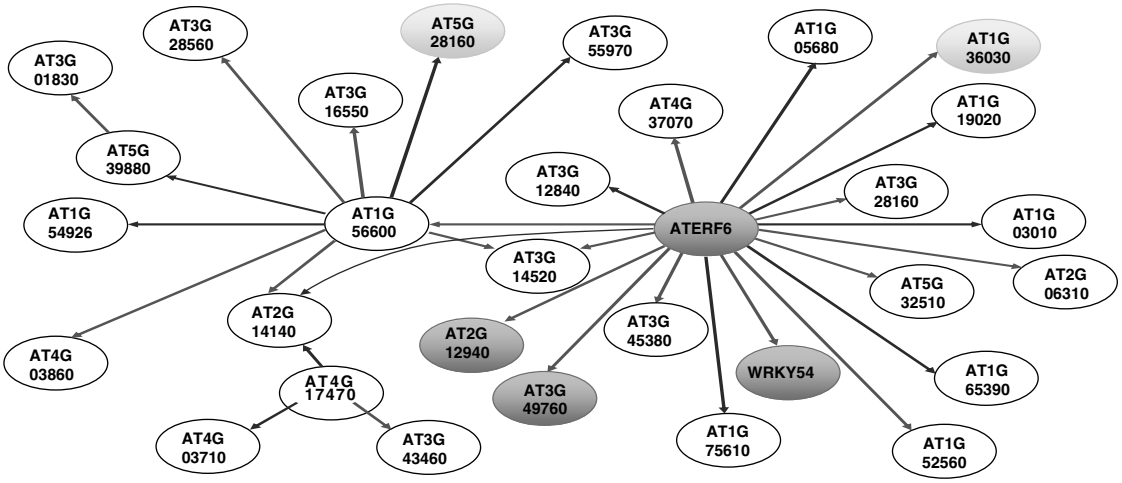

Fig. 6. Inferred network in *A.thaliana* based on 35 links generated from stress response datasets in shoots. The red arrows represent activation while the blue arrows represent suppression. Putative transcription factors are shown in purple. F-box family proteins are shown in yellow.

hand, the implicit description of hidden regulatory factors by this approach may lead to prediction errors. Generally, the mRNA levels measured in a microarray experiment are the results of a variety of complex events including gene transcription and mRNA degradation (Gardner and Faith, 2005). With such events, dynamic Bayesian networks can be used to derive the regulations among biomolecules (Nachman *et al*., 2004; Rangel *et al*., 2004; Beal *et al*., 2005; Li and Zhan, 2006).

To examine causal relation among genes, a major source of errors comes from the noises of the gene expression data intrinsic to microarray technologies (Thattai and van Oudenaarden, 2001). To reduce the defect of unreliable data, GNR is able to assess the quality of microarray datasets by comparing their inferred results, and remove the inconsistent dataset using a clustering method according to their degree of inconsistency. As a result, GNR can alleviate the impact of noises to improve the prediction accuracy. In addition, GNR is also effective for reducing the effects of stochastic fluctuations by introducing the sparsity leverage λ and combining the several datasets together (see the details in the Supplementary Material). Depending on the prior information of the data (e.g. reliability of experiments or number of experiments), we can also allocate different weights for the datasets to maximally utilize the information of reliable gene expression data.

Our method also has some limitations owing to the nature of microarray gene expression data, like other existing methods for GN inference. In particular, although GNR can provide a relationship in which the expression of one gene can lead to an increased (or decreased) expression of another gene, such a relationship does not show the exact mechanism. A predicted regulatory relationship does not always mean genetic regulation by a transcriptional factor. Some regulation can be at the post-transcriptional or post-translational level, which are often not reflected in mRNA expression levels detected by microarrays. Therefore, there is a need for integration with other information sources to derive regulatory networks in an accurate manner. In other cases, the transcriptional factors for direct regulation are not selected for GN construction due to their low expression levels or statistically insignificant changes. Hence, the GN models that we predicted include both direct and indirect regulations (i.e. via hidden variables). Typically one can interpret an edge in a GN model as the net effect if the gene from the source is deleted. For example, if an arrow pointing from gene A to gene B for activation, it is expected that deleting gene A will lead to an increased expression of gene B. Notice that the inferred results by GNR are only valid on the assumption that the dynamics of the system can be captured by the time intervals between the data points. Nevertheless, our predicted regulatory network is testable through a comparison in microarray data between wild type and mutant with specific deletion.

Currently, GNR is aimed to infer the consistent structure from a variety of datasets but for the same species or organism. With the similar mechanism, GNR can be extended to identify the conserved network patterns or motifs (Kelly *et al*., 2003) from the datasets of either the same species or different species, by adjusting the parameter λ, i.e. a higher λ results in a more consistent or conserved network.

ACKNOWLEDGEMENTS

The authors are grateful to the associate editor and anonymous referees for comments and helping to improve the earlier version. This work is partly supported by Grant sponsor: project Bioinformatics, Bureau of Basic Science, CAS. The effort of T.J. and D.X. was supported by a USDA grant CSREES 2004–25,604-14,708.

Conflict of Interest: none declared.

REFERENCES

Akutsu,T. *et al*. (2000) Inferring qualitative relations in genetic networks and metabolic pathways. *Bioinformatics*, **16**, 727–734.

Alonso-Monge,R. *et al*. (2001) Hyperosmotic stress response and regulation of cell wall integrity in saccharomyces cerevisiae share common functional aspects. *Mol. Microbiol*., **41**, 717–730.

Basso,K. *et al*. (2005) Reverse engineering of regulatory networks in human B cells. *Nat. Genet*., **37**, 382–390.

Beal,M.J. *et al*. (2005) A Bayesian approach to reconstructing genetic regulatory networks with hidden factors. *Bioinformatics*, **21**, 349–356.

Chen,L. and Aihara,K. (2001) Stability and bifurcation analysis of differential-difference-algebraic equations. *IEEE Trans. Circuits Syst. I*, **48**, 308–326.

Chen,L. and Aihara,K. (2002) Stability of genetic regulatory networks with time delay. *IEEE Trans. Circuits Syst. I*, **49**, 602–608.

Destruelle,M. *et al.* (1994) Identification and characterization of a novel yeast gene: the *ygp1* gene product is a highly glycosylated secreted protein that is synthesized in response to nutrient limitation. *Mol. Cell Biol.*, **14**, 2740–2754.

Dewey,T.G. and Galas,D.J. (2001) Dynamic models of gene expression and classification. *Funct. Integr. Genomics*, **1**, 269–278.

D'haeseleer,P. *et al.* (2000) Genetic network inference: from co-expression clustering to reverse engineering. *Bioinformatics*, **16**, 707–726.

D'Haeseleer,P., Wen,X., Fuhrman,S. and Somogyi,R. (1999) Linear modeling of mRNA expression levels during cns development. In Altman,R.B., Dunker,A.K., Hunter,L., Klein,T.E. and Lauderdaule,K. (eds), *Pacific Symposium on Biocomputing Vol. 4*, pp. 41–52.

Doolin,M.T. *et al.* (2001) Overlapping and distinct roles of the duplicated yeast transcription factors ace2p and swi5p. *Mol. Microbiol.*, **40**, 422–432.

Fujimoto,S.Y. *et al.* (2000) *Arabidopsis* ethylene-responsive element binding factors act as transcriptional activators or repressors of gcc box-mediated gene expression. *Plant Cell*, **12**, 393–404.

Gardner,T.S. and Faith,J. (2005) Reverse-engineering transcription control networks. *Phys. Life Rev.*, **2**, 65–88.

Gustafsson,M. *et al.* (2005) Constructing and analyzing a large-scale gene-to-gene regulatory network-lasso-constrained inference and biological validation. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, **2**, 254–261.

H,D.J. (2002) Modeling and simulation of genetic regulatory systems: a literature review. *J. Comput. Biol.*, **9**, 67–103.

Hartemink,A.J. (2005) Reverse engineering gene regulatory networks. *Nat. Biotechnol.*, **23**, 554–555.

Holter,N.S. *et al.* (2001) Dynamic modeling of gene expression data. *Proc. Natl Acad. Sci. USA*, **98**, 1693–1698.

Hughes,T.R. *et al.* (2000) Functional discovery via a compendium of expression profiles. *Cell*, **102**, 109–126.

Husmeier,D. (2003) Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks. *Bioinformatics*, **19**, 2271–2282.

Kelley,B.P. *et al.* (2003) Conserved pathways within bacteria and yeast as revealed by global protein network alignment. *Proc. Natl Acad. Sci. USA*, **100**, 11394–11399.

Kornberg,R.D. (1998) Mechanism and regulation of yeast rna polymerase ii transcription. *Cold Spring Harb. Symp. Quant. Biol.*, **63**, 229–232.

Levine,M. and Davidson,E.H. (2005) Gene regulatory networks for development. *Proc. Natl Acad. Sci. USA*, **102**, 4936–4942.

Li,H. and Zhan,M. (2006) Systematic intervention of transcription for identifying network response to disease and cellular phenotypes. *Bioinformatics*, **22**, 96–102.

Mrsa,V. and Tanner,W. (1999) Role of naoh-extractable cell wall proteins ccw5p, ccw6p, ccw7p and ccw8p (members of the pir protein family) in stability of the *Saccharomyces cerevisiae* cell wall. *Yeast*, **15**, 813–820.

Nachman,I. *et al.* (2004) Inferring quantitative models of regulatory networks from expression data. *Bioinformatics*, **20** (Suppl. 1), i248–i256.

Rangel,C. *et al.* (2004) Modeling T-cell activation using gene expression profiling and state-space models. *Bioinformatics*, **20**, 1361–1372.

Soinov,L. (2003) Supervised classification for gene network reconstruction. *Biochem. Soc. Trans.*, **31**, 1497–1502.

Tegner,J. *et al.* (2003) Reverse engineering gene networks: integrating genetic perturbations with dynamical modeling. *Proc. Natl Acad. Sci. USA*, **100**, 5944–5949.

Thattai,M. and van Oudenaarden,A. (2001) Intrinsic noise in gene regulatory networks. *Proc. Natl Acad. Sci. USA*, **98**, 8614–8619.

Tu,Y. *et al.* (2002) Statistics quantitative noise analysis for gene expression microarray experiments. *Proc. Natl Acad. Sci. USA*, **99**, 14031–14036.

van Someren,E.P. *et al.* (2001) Robust genetic network modeling by adding noisy data. In *Proceedings of 2001 IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing*.

Yeung,M.K.S. *et al.* (2002) Reverse engineering gene networks using singular value decomposition and robust regression. *Proc. Natl Acad. Sci. USA*, **99**, 6163–6168.

Zak,D.E. *et al.* (2003) Importance of input perturbations and stochastic gene expression in the reverse engineering of genetic regulatory networks: insights from an identifiability analysis of an *in silico* network. *Genome Res.*, **13**, 2396–2405.