Gene expression

Fast network component analysis (FastNCA) for gene regulatory network reconstruction from microarray data

Chunqi Chang\(^1,\)*, Zhi Ding\(^2\), Yeung Sam Hung\(^1\) and Peter Chin Wan Fung\(^3,\)*

\(^1\)Department of Electrical and Electronic Engineering, The University of Hong Kong, Pokfulam Road, Hong Kong, \(\text{Department of Electrical and Computer Engineering, University of California, Davis, CA 95616, USA}\) and \(\text{Department of Medicine, The University of Hong Kong, Pokfulam Road, Hong Kong}\)

Received on October 19, 2007; revised on February 25, 2008; accepted on April 8, 2008

Advance Access publication April 9, 2008

Associate Editor: John Quackenbush

ABSTRACT

Motivation: Recently developed network component analysis (NCA) approach is promising for gene regulatory network reconstruction from microarray data. The existing NCA algorithm is an iterative method which has two potential limitations: computational instability and multiple local solutions. The subsequently developed NCA-r algorithm with Tikhonov regularization can help solve the first issue but cannot completely handle the second one. Here we develop a novel Fast Network Component Analysis (FastNCA) algorithm which has an analytical solution that is much faster and does not have the above limitations.

Results: Firstly FastNCA is compared to NCA and NCA-r using synthetic data. The reconstruction of FastNCA is more accurate than that of NCA-r and comparable to that of properly converged NCA. FastNCA is not sensitive to the correlation among the input signals, while its performance does degrade a little but not as dramatically as that of NCA. Like NCA, FastNCA is not very sensitive to small inaccuracies in a priori information on the network topology. FastNCA is about several tens times faster than NCA and several hundreds times faster than NCA-r. Then, the method is applied to real yeast cell-cycle microarray data. The activities of the estimated cell-cycle regulators by FastNCA and NCA-r are compared to the semi-quantitative results obtained independently by Lee et al. (2002). It is shown here that there is a greater agreement between the results of FastNCA and Lee’s, which is represented by the ratio 23/33, than that between the results of NCA-r and Lee’s, which is 14/33.

Availability: Software and supplementary materials are available from http://www.eee.hku.hk/~cqchang/FastNCA.htm

Contact: cqchang@eee.hku.hk

1 INTRODUCTION

High-throughput microarray technology is an important advancement in the post-genome period. Much effort has been devoted to obtain information, particularly functions of living organisms using data from DNA microarrays. The starting point of the new field of systems biology, which has emerged in the past decade, is the awareness of the fact that cells must respond and adapt to environmental perturbation by adjusting their genomic expression program in short-term and long-term measures via positive or negative regulation of the DNA binding transcription regulators/factors (TF). The TF activity (TFA), which can be defined as the concentration of its subpopulation capable of DNA binding, is therefore an important parameter, but usually very difficult to be measured directly in an in vivo system (see e.g. Jorgensen and Tyers, 2000; Toone et al., 1997; Simon et al., 2001). A key aspect of systems biology is to develop mathematical networks to reconstruct the hidden dynamics of the transcription regulatory activities based on DNA microarray data. In the terminology of engineering, the biological system is taken as an information processing system and relevant mathematical techniques in the general field of information processing have been applied, developed with specific adaptation treatment, to analyze biological system in recent years. Some integrative or novel mathematic methods have also emerged along this line to obtain informative signals from a low-dimensional regulatory network system which generates a set of high-dimensional data.

Essentially such investigation amounts to solving the inverse problem under constraints. The principal component analysis (PCA) and independent component analysis (ICA) have been developed to solve the problem under the mathematical assumptions that the signal sources are mutually orthogonal and statistically independent, respectively (Alter et al., 2000; Liebermeister, 2002). However, neither PCA nor ICA is suitable in biological systems in general because the orthogonality property and the statistical independent property are not usually valid (Chang et al., 2006, 2007). Other types of constraints have to be imposed in systems biology.

The following linear model of gene regulatory network, with and without noise, is considered (Liao et al., 2003):

\[ X = AS, \quad (1) \]

\[ Y = X + \Gamma = AS + \Gamma, \quad (2) \]

where \( Y \) is the known \( N \times K \) matrix representing expression dynamics of genes measured by DNA microarray, \( S \) is the unknown \( M \times K \) matrix representing activities of the associated TFs, \( \Gamma \) is the inevitable measurement noise and the unknown \( N \times M \) matrix \( A \) specifies the connectivity strength (transcription regulation strength) between the regulatory domain (TFA) and output domain (expression dynamics), while \( N, M \) and \( K \)

*To whom correspondence should be addressed.
are the number of genes, number of TFs and number of time points, respectively. \(X\) is the ideal gene-expression matrix where there is no noise in the measurement. This model can be used to represent any linear regulatory networks in systems biology other than transcriptional regulatory network.

In general, gene regulation processes are dynamic and non-linear. The static linear model given in Equation (1) is derived under the following assumptions. Gene expression can be considered as the product of the regulating TFAs using a combinatorial power-law model, which can be viewed as a log-linear approximation of non-linear kinetic systems in multiple dimensions (Savageau, 1976;Torrest and Voit, 2002;Voit and Almeida, 2004). It is assumed that the time scale of TFA change is much longer than that of mRNA. Therefore, mRNA levels at most time are in a quasi-steady state, and thus in this quasi-steady state the dynamic model becomes approximately static. Furthermore, DNA microarrays are commonly expressed in log-ratio, and for this log-ratio the product of power-law model for the raw mRNA level becomes a linear model. This explains why we have a static linear model as in Equation (1). A detailed derivation of this model can be found in Liao et al. (2003) and Galbraith et al. (2006).

Our problem is to recover \(A\) and \(S\) from \(X\) according to Equation (2). The method developed in Liao et al. (2003) to solve such inverse problem is called network component analysis (NCA), which is based on the following three constraints referred to as NCA criteria: (i) the connectivity matrix \(A\) must be full-column rank; (ii) when an element in the regulatory domain is removed along with all the output elements connected to it, the connectivity matrix of the resulting network is still of full-column rank; (iii) \(S\) must have full-row rank.

The NCA criteria were shown in Liao et al. (2003) to render unique solution of the inverse problem if there is no noise. When NCA is employed to analyze a problem, first the identifiability property of \(A\), i.e. criterion (i) and criterion (ii), must be checked. Then the bi-linear objective function is minimized under the constraint that the network structure (the non-zero pattern of \(A\)) is conserved.

\[
\min_{A,S} |Y - AS|^2 \quad s.t. \quad A \in Z_0,
\]

(3)

where \(Z_0\) is the topology induced by the network structure. Then the actual estimation of \(A\) and \(S\) is performed by a two-step alternating least-squares algorithm which exploits the biconvexity properties of linear decompositions. The criterion (iii) for \(S\) is checked after an estimate is obtained.

Experimental validation of this NCA method has been carried out using a network of seven hemoglobin solutions which contain three hemoglobin components. Absorption spectra measurement was carried out the result of which was compared with that deduced theoretically using NCA, PCA and ICA methods. The analysis of Liao et al. (2003) demonstrates the superiority of NCA over the other two methods. In the same article, the NCA approach was also applied to study the dynamics of the TFA for 11 TF involved in cell-cycle regulation of *Saccharomyces cerevisiae*. The network topology, or connectivity between TF and genes was taken from genome wide location analysis (Lee et al., 2002)

Microarray datasets pertaining to *S.cerevisiae* were taken from cultures synchronized by elutriation, \(\alpha\)-factor arrest, and arrest of a cdc15 temperature sensitive mutant. The estimates of TFA can therefore be obtained using Equation (3).

It was later pointed out by Tran et al. (2005) that the bilinear optimization scheme in the NCA algorithm has two potential limitations: (1) ill-conditioned matrices generated during the iteration may make numerical algorithm unstable and suggest physically unreasonable solution; (2) there may exist multiple local minima. The Tikhonov regularization method was then applied to the optimization problem given in Equation (3) in order to overcome the first limitation, and it turned out that it was also helpful in solving the multiple solution issue. This Tikhonov regularization method is referred to as NCA-r.

The NCA approach to gene-regulatory network reconstruction is promising as evidenced by its successful applications (Kao et al., 2004, 2005). However, the limitations of NCA are only partially overcome by NCA-r, and the regularization makes NCA-r computationally much more complicated (and time consuming) than NCA whose computational complexity is already very high due to the inevitable occurrence of iterations. Therefore, there is a need to develop a fast, stable and globally convergent algorithm to reconstruct gene-regulatory networks that satisfy the three NCA criteria.

## 2 FAST NETWORK COMPONENT ANALYSIS

### 2.1 Analytical solution in the ideal case

In Equation (1), since we can always re-order the rows of \(X\) and \(A\), we assume without loss of generality that the first column of \(A\) can be partitioned as

\[
a_1 = \begin{bmatrix} \tilde{a}_1 \\ 0 \end{bmatrix}.
\]

(4)

In the ideal case where the noise term \(\Gamma = 0\), we partition Equation (1) accordingly as

\[
\begin{pmatrix} X_r \\ X_t \end{pmatrix} = \begin{pmatrix} \tilde{a}_1 & A_r \\ 0 & A_t \end{pmatrix} \begin{pmatrix} s_r^T \\ s_t^T \end{pmatrix} = \begin{pmatrix} \tilde{a}_1 s_r^T + A_r s_r^T \\ A_t s_t^T \end{pmatrix}.
\]

(5)

If we can remove the term \(A_r s_r^T\) in the expression of \(X_t\) above, then we can get \(\tilde{a}_1\) by a matrix factorization. In the following we show that this can be done by matrix projection. The basics of matrix projection are shown in the Appendix. To make the derivation clear we take transpose for both sides of Equation (5):

\[
X_r^T = s_r^T \tilde{a}_1^T + S_r^T A_r^T,
\]

(6)

\[
X_t^T = S_t^T A_t^T.
\]

(7)

If we can get the orthogonal projection matrix associated with \(S_r^T\), which is \(P_S\) as defined by Equation (A3) in the Appendix, to remove the term \(S_r^T A_r^T\) in Equation (6) we just apply \(P_S\) to both sides of the Equation (6), and since \(P_S^T S_t^T = 0\) according to Equation (A5) we have:

\[
P_S^T X_r^T = P_S^T s_r^T \tilde{a}_1^T = \tilde{s}_1 \tilde{a}_1^T,
\]

(8)
where \( s_i = P_{\text{NC}}^c s_i \neq 0 \) since \( s_i \) is linearly independent of \( S_i \) according to NCA criterion (iii). We have no access to \( S_i \), and thus no direct access to \( P_{\text{NC}}^c \), but fortunately we have access to \( X_c \). According to Equation (7) and Equation (A6), and since \( A_i \) is of full-column rank according to NCA criterion (ii), we have
\[
P_{\text{NC}}^c = P_{\text{NC}}^c S_i.
\]
Therefore, from Equation (8) and Equation (9) we can form the following matrix
\[
\tilde{X}_c^T = P_{\text{NC}}^c X_c^T = \tilde{s}_1 \tilde{a}_1^T.
\]
Hence a rank-1 factorization of \( \tilde{X}_c \) defined by Equation (10) gives an estimate of \( \tilde{a}_1 \) up to a scaling ambiguity, and therefore gives the estimate of the first column of \( A_i \) according to Equation (4). Since we can always re-order the columns of \( A \) and the rows of \( S \) accordingly while keeping the equality in Equation (1), the other columns of \( A \) can be estimated in the same way.

### 2.2 Numerical algorithm in the noisy case

The analytical solution given in Section 2.1 cannot be applied directly to the measured data \( Y \) because of noise. Noise can be coped with by singular value decomposition (SVD). Let the SVD of \( Y \) in Equation (2) be
\[
Y = \sum_{k=1}^{K} \sigma_k u_k v_k^T,
\]
where the singular values \( \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_K \geq 0 \), and \( u_k \) and \( v_k \) are the left and right singular vectors. If the noise term \( \Gamma \) is small and \( X \) is not ill-conditioned, the first \( M \) singular values are contributed mainly by \( X \), and the remaining insignificant singular vectors are contributed mainly by \( \Gamma \). Therefore, we can remove the insignificant singular values to increase signal-to-noise (SNR) ratio (Scharf, 1991). Depending on the noise level, the number of singular values to be removed can be flexible, and this provides a trade-off between bias and variance (Scharf, 1991; Scharf and Tufts, 1987).

Assume that we keep the first \( L \) (\( \geq M \)) singular values, where \( L \) may be different from \( M \). Let \( \Sigma_L \) and \( \Sigma_M \) be two diagonal matrices with diagonal entries be the first \( L \) leading singular values and the remaining \( K - L \) smaller singular values of \( Y \), respectively, then we can write the SVD of \( Y \) in Equation (11) as
\[
Y = U_L \Sigma_L V_L^T + U_M \Sigma_M V_M^T,
\]
where the first part is called a rank-\( L \) Eckart–Young–Mirsky (EYM) approximation of \( Y \), and we denote it as
\[
Y_L = U_L \Sigma_L V_L^T.
\]
It is known that the EYM approximation \( Y_L \) is optimal under the constraint that the rank of \( Y_L \) is \( L \) (Golub and van Loan, 1996).

We further ortho-normalize \( Y_L \) and work on the left singular vectors \( U_L \) rather than \( Y_L \) itself. Let \( W = U_L \), we have
\[
W = Y_L = Y V_L \Sigma_L^{-1} = (\mathbf{A} \mathbf{S} + \Gamma) V_L \Sigma_L^{-1} \overset{\text{def}}{=} \mathbf{A} \mathbf{S} + \Gamma \text{'},
\]
which has the same form as Equation (2), but with \( Y \) replaced by \( W \) (of \( L \) columns), \( S \) replaced by \( S \) (of full-row rank) which also satisfies NCA condition (iii), and \( \Gamma \) replaced by \( \mathbf{\tilde{\Gamma}} \) (transformed perturbation). Let \( W \) be partitioned in the same way as in Equation (5):
\[
W = \begin{bmatrix} W_r \\ W_f \end{bmatrix} = \begin{bmatrix} \mathbf{\tilde{a}}_1 & \mathbf{A}_r \\ 0 & \mathbf{A}_r \end{bmatrix} \begin{bmatrix} \mathbf{S}_r \\ \mathbf{\tilde{\Gamma}}_r \end{bmatrix}.
\]
We can take transpose of Equation (15) and get
\[
W_r^T = \mathbf{\tilde{s}}_1 \mathbf{\tilde{a}}_1^T + \mathbf{S}_r \mathbf{A}_r^T + \mathbf{\tilde{\Gamma}}_r^T.
\]
If we have the projection matrix \( P_{\text{NC}}^c \) and apply it Equation (16) then
\[
P_{\text{NC}}^c W_r^T = \tilde{s}_1 \tilde{a}_1^T + P_{\text{NC}}^c \mathbf{\Gamma}_r^T,
\]
where \( \tilde{s}_1 = P_{\text{NC}}^c \mathbf{\tilde{s}}_1 \neq 0 \) due to NCA criterion (iii). Unlike Equation (7), \( P_{\text{NC}}^c \mathbf{\tilde{\Gamma}}_r^T \) due to the noise term in Equation (17). Nevertheless, we can have an estimate of \( P_{\text{NC}}^c \) from Equation (17) by SVD. Let the SVD of \( W_r \) be partitioned as
\[
W_r = U_1 \Sigma_1 V_1^T + U_0 \Sigma_0 V_0^T,
\]
where the diagonal entries of \( \Sigma_1 \) and \( \Sigma_0 \) are the first \( M - 1 \) leading singular values and the remaining \( L - M + 1 \) smaller singular values of \( W_r \), respectively.

Since \( S_i \) is of full row rank \( M - 1 \) according to the partition Equation (15) and NCA condition (iii), an estimate of \( P_{\text{NC}}^c \) can be
\[
\hat{P}_{\text{NC}}^c = V_0 \Sigma_0^T.
\]
Then substituting \( \hat{P}_{\text{NC}}^c \) to \( P_{\text{NC}}^c \) in Equation (18) we have
\[
\hat{W}_r = \hat{P}_{\text{NC}}^c W_r = \hat{s}_1 \hat{a}_1^T + \hat{P}_{\text{NC}}^c S_r \mathbf{A}_r^T + \hat{P}_{\text{NC}}^c \mathbf{\Gamma}_r^T = \hat{s}_1 \hat{a}_1^T + \hat{\mathbf{\Gamma}}_r^T,
\]
where \( \hat{\mathbf{\Gamma}}_r^T \) is a noise term which is small when \( \Gamma \) is small. Then a rank-1 EYM approximation of \( \hat{W}_r \) gives an estimate of \( \hat{a}_1 \), which is the first right singular vector of \( \hat{W}_r \). With \( \mathbf{A} \) estimated, we can get a least square estimate of \( S \):
\[
S = \mathbf{A}^T Y_L.
\]
Now we summarize the algorithm, which will be referred to as Fast Network Component Analysis (FastNCA), since it is free of iterations and thus fast in computation.

**FastNCA algorithm:**

1. Perform a rank-\( L \) EYM approximation of \( Y \) by SVD as \( Y_L = U_L \Sigma_L V_L^T \), and let \( W = U_L \).
2. Estimation of \( \mathbf{A} \):
   - For \( i = 1, \ldots, M \)
     - Re-order rows of \( W \) so that the \( i \)th column of \( \mathbf{A} \) has the structure \( a_i = \begin{bmatrix} \mathbf{\tilde{a}}_i \\ 0 \end{bmatrix} \).
     - Partition \( W = \begin{bmatrix} W_r \\ W_f \end{bmatrix} \) conformally with this structure.
Do SVD for $W_r$ and get the last $L-M+1$ right singular vectors, denoted as a matrix $V_0$.

Compute $\tilde{a}_i = \text{the first left singular vector of } W_r V_0^T$.

3. Estimate the TFA matrix by $S = A^T Y_L$.

For the choice of $L$, it has been shown by our numerical simulation experiments that in the case of random Gaussian noise we can choose $L = M$ with good performance and superior computational efficiency. If $L = M$, due to the column dimension of $W_r$ (which is $M$) $V_0$ in Equation (20) becomes a 1-dimensional vector $v$, which is the $M$th (the last) right singular vector of $W_r$, so that

$$\hat{p}_{\tilde{v}}^T = vv^T. \quad (23)$$

Then Equation (21) becomes

$$\tilde{W}_r = \hat{p}_{\tilde{v}}^T W_r = vv^T W_r^T = \tilde{s}_1 \tilde{a}_1^T + F_0^T. \quad (24)$$

Hence we get an estimate of $\tilde{a}_i$ simply as

$$\tilde{a}_i = W_r v. \quad (25)$$

In the synthetic and experimental study presented in Sections 3 and 4, we shall adopt $L = M$. Compared to the NCA algorithm described by Equation (3), FastNCA is not only much faster but also not subjected to the problems of instability and multiple local solutions. FastNCA copes with measurement noise by subspace decomposition through SVD, different from NCA.

### 3 SYNTHETIC AND EXPERIMENTAL DATA

#### 3.1 Synthetic data

First, synthetic data are used to demonstrate the performance of the FastNCA and its comparison to the original iterative NCA algorithm. Many issues can be controlled in the network analysis model Equation (1). We consider the following aspects:

1. the size of the network; 2. the complexity of the network; 3. the correlation between input signals, i.e. TFAs; 4. the length of the data; 5. the level of noise and 6. inaccuracies of the network topology.

Three networks with different sizes and complexity are denoted as Networks A (321 × 25, 321 genes, 25 TFs), B (400 × 25) and C (100 × 15). Figure 1 shows the in-degree distribution of these networks, which represents the complexity of a network and is defined for each gene as the number of TFs regulating this specific gene (Boscolo et al., 2005). Note that Networks A and B are similar to the example networks in Boscolo et al. (2005), and Network C is similar to an example network in Tran et al. (2005). The non-zero entries in the connectivity matrix A are randomly generated with standard Gaussian distribution.

Input signals are zero-mean white Gaussian with an identical pairwise correlation coefficient denoted as $\rho$. Input data with $\rho = 0$, 0.5 and 0.9 are generated to represent uncorrelated, moderately correlated and strongly uncorrelated signals, respectively. Noises are assumed to be white Gaussian, while the noise level is defined as the ratio between the power of the noise ($F_r$, and the true gene expressions ($A S$). A variety of noise levels, 0.05, 0.1, 0.2, 0.3 and 0.75 are used to test the robustness of the algorithms to noise.

#### 3.2 Experimental data

As in Liao et al. (2003), here the cell-cycle regulation in *S.cerevisiae* is used as an example to test the applicability and assess the performance of the NCA algorithms. The data are downloaded from the website of Liao et al., http://www.ee.ucla.edu/~riccardo/NCA/nca.html, where the time-course microarray data of *S.cerevisiae* are from Spellman et al. (1998) and the network topology are from Lee et al. (2002). There are three experiments, based on three different kinds of synchronization. The first set of cell-cycle microarray data are taken from cultures synchronized by elutriation. It contains one cell cycle, taken every 30 min from 0 min to 390 min, 14 time points. The second set are from synchronization by $\alpha$-factor arrest, two cell cycles, 18 time points, taken every 7 min from 0 min to 119 min. The third set are from synchronization by arrest of a cdc15 temperature sensitive mutant, three cell cycles, 24 time points, taken every 20 min from 10 min to 70 min, and then every 20 min to 250 min, and then at 270 and 290 min. In order to reconstruct the TF-gene network, these three sets of microarray data are then cascaded as a whole (totally 56 time points) for network component analysis. There are 11 TFs which are related to yeast cell-cycle regulation and are of interest in this study. The names of these TFs are Ace2, Fkh1, Fkh2, Mbp1, Mcm1, Ndd1, Skn7, Sbf1, Swi4, Swi5 and Swi6.

We focus on these 11 TFs and their influenced genes. The original network contains 570 genes and 44 TFs. To see the relevance of the NCA criteria to the present problem, we will re-state the NCA criteria in the context of cell-cycle regulation as: (1) the TFs are structurally independent so that the structure of $A$ supports a full-column rank matrix, (2) if a TF is removed along with all the genes regulated by the removed TF, the remaining TFs are still structurally independent and (3) the TF activities are independent. Unfortunately, the original structure of $A$ of the original 570 × 44 network does not naturally satisfy NCA criterion (ii). However, we may trim $A$ by removing some selected TFs along with all the genes regulated by these TFs, and get a trimmed network that satisfies the NCA criteria (i) and (ii). We are able to find a trimmed network containing 441 genes and 33 TFs, in which the above mentioned 11 cell-cycle related TFs are included. The dimension of $S$ for this trimmed network is $33 \times 56$, which generically satisfies the last NCA criterion, and thus all three NCA criteria are satisfied.

The above mentioned Spellman’s wild type yeast cell-cycle microarray data (Spellman et al., 1998), combined with a...
dataset from a mutant that lacks two forkhead TFs (Zhu et al., 2000). Fkh1 and Fkh2 synchronized by α-factor arrest, have also been processed in line with the procedure described in Yang et al. (2005). Missing values of the gene-expression data are imputed by knn (Troyanskaya et al., 2001). After knn imputation, common genes without missing data points between the two datasets were selected for network component analysis. We have adopted the same sub-network analysis as in Yang et al. (2005). For each sub-network, the network topology required by network component analysis is determined from the ChIP-chip data by Lee et al. (2002) with the P-value threshold set to 0.001. Our four sub-networks are found to contain 1247, 943, 1114 and 921 genes, respectively. Judging from the statement of Yang et al. (2005) in the caption of Figure 2, we notice that their networks contain 1110, 847, 1015 and 793 genes, respectively. Even though the corresponding number of genes of the two sets of sub-networks are not identical, it is still fruitful to compare the reconstructed TFAs obtained from these two sets of networks. The slight difference in the number of genes of the two sets of networks may be caused by possibly different versions of the ChIP-chip dataset. It can be shown that these network topologies also satisfy the NCA criteria and thus allow the application of FastNCA algorithm. The data of our four sub-networks can be downloaded from our website.

4 RESULTS

4.1 Results on synthetic data

In this section, we assess and compare the performance of three algorithms, where the MATLAB implementation of NCA and NCA-r are obtained from the website of Liao et al., http://www.ee.ucla.edu/~riccardo/NCA/nca.html. Our FastNCA is also implemented in MATLAB. In order to assess the performance, the estimated connectivity matrix is properly scaled so that the sum of the absolute value of any column of the estimated matrix is equal to that of the corresponding column of the true connectivity matrix. The performance index is defined as the relative fitting error (in dB) of the TFAs.

We first test the performance of the algorithms under settings with a combination of a variety of noise level (0.05, 0.1, 0.2, 0.3, 0.6), signal length (K = 50, 200, 1000), and correlation between input signals (ρ = 0.0, 0.5, 0.9). We take 100 Monte Carlo experiments for NCA and FastNCA, but only 30 experiments for NCA-r because it is too time consuming. The mean and median of the fitting error (in dB) can then be calculated. It is found that for FastNCA and NCA-r the mean is almost equal to the median, but for NCA the mean is much greater than the median because that the solution of NCA is not stable and may be far away from the true solution if the initial point of the algorithm is not properly selected. Therefore the median of the fitting error is used as the summary statistics to be presented in this article. Results are shown in Figure 2 for uncorrelated and highly correlated input signals (TFAs). The results for ρ = 0.5 is very similar to that for ρ = 0 and so is not shown here.

It is shown that for all three network examples in general (except some cases for NCA), the performance of all three algorithms decreases when the noise level increases or the signal length decreases. Though these algorithms do not require the input signals to be uncorrelated, in practice high correlation between input signals will degrade the performance of the algorithms. In Figure 2(b) we can see that if the input signals are strongly correlated, for each example network there are cases where the median of the fitting error NCA is unacceptably high (>10 dB). This means that the original NCA algorithm can become extremely unstable when the input signals are strongly correlated. In all cases the performance of FastNCA is at least comparable to NCA (note that if the mean of fitting error is considered, the performance of NCA will be greatly degraded) but much better than NCA-r especially when the noise level is low. In the case of noise level equal to 0.05, the performance of FastNCA is about 5 dB better than NCA-r, despite the complexity of the network and the level of input
signal correlation. Only when the noise level is as high as 0.6, the performance of NCA-r becomes comparable to FastNCA. Due to the convergence problem, the performance of NCA becomes inferior to FastNCA when the input signals are highly correlated.

Now we compare computation complexity of the three algorithms using the above Monte Carlo simulations. The simulations are performed in MATLAB 7.3 installed on a Windows XP system with a 2.4G Hz CPU. The average computation time (in seconds) is shown in Table 1. It shows that in general FastNCA is several tens times faster than NCA and several hundreds times faster than NCA-r.

In practice, the connectivity topology of the TF-gene network is conventionally built from biological experiments, and such experimentally derived gene-TF interactions are often prone to errors. To test the robustness of the algorithms to the inaccuracy of the a priori knowledge on the network topology, in the previous set of simulations we introduce random errors in the hypothesized connectivity topology when applying the reconstruction algorithms. As in Boscolo et al. (2005), a number of TF-gene pairs in the network are randomly selected and changed to be connected if there is no connection or to be disconnected if there is already a connection. The number is equal to a percentage of the number of non-zero entries in the connectivity matrix. To simulate the topology error, such percentage is chosen to be 2, 5 and 10% in our simulation. In this simulation we take 100 Monte Carlo runs, and only FastNCA and NCA are tested since NCA-r is too time consuming. Here we just look at a representative example for Network A since it is not possible to present all the results. In the case where $K=200$, $\rho=0.5$ and noise level is 0.1, a topology error of 0% (no error), 2%, 5% and 10% will cause FastNCA a median fitting error of $-20.2$ dB, $-19.8$ dB, $-19.0$ dB and $-17.2$ dB respectively, while for NCA the fitting error are $-20.2$ dB, $-19.9$ dB, $-19.0$ dB, $-17.1$ dB, respectively. We see that small topology error does not greatly degrade the performance since a 10 percent error only degrades the performance by 3 dB. According to the median of the performance indices, there is no significant difference between FastNCA and NCA.

### 4.2 Results on experimental data

All the three algorithms are used to reconstruct the cell-cycle TF-gene-regulation network from the time course *S. cerevisiae* (yeast) microarray data. Since we are only interested in those TFs related to cell-cycle regulation, we show only the results for the 11 cell-cycle related TFAs. The results for FastNCA and NCA-r [according to Equation (27) of Tran et al., 2005] are shown in Figure 3. Bootstrap is used in Liao et al. (2003) to assess the variability of the estimates. However, it is not always justified to use bootstrap to assess the variability of an estimate (Efron and Tibshirani, 1998). Therefore, here we do not adopt this bootstrap procedure, and instead in another experiment we perform a sub-network analysis to assess the variability as in Yang et al., (2005).

Ren et al. (2000) carried out a genome-wide location analysis experiment for each of the 106 yeast strains that expressed epitope-tagged TFs/regulators. Then chromatin immunoprecipitation (ChIP) experiment was performed on each of these 106

<table>
<thead>
<tr>
<th>Signal length</th>
<th>NCA-r</th>
<th>FastNCA</th>
<th>NCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>119</td>
<td>584</td>
<td>0.07</td>
</tr>
<tr>
<td>200</td>
<td>195</td>
<td>854</td>
<td>0.01</td>
</tr>
<tr>
<td>1000</td>
<td>0.01</td>
<td>0.04</td>
<td>0.21</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Amplitude</th>
<th>Time (h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FastNCA</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>NCA-r</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
tagged strain samples. Promoter (of the associated TF) regions which were enriched through the ChIP experiment could be identified by hybridization to microarrays containing a genome-wide set of yeast promoter regions (Lee et al., 2002). Such genome-wide location data were analyzed and filtered by a special error model (Hughes et al., 2000). Identification of a set of promoter regions that are bound by certain TFs led Lee et al. (2002) to predict sequence motifs being bound by the TFs, forming network motifs. Based on six types of regulating network motifs thus found, an algorithm was developed to build up a model for the yeast cell-cycle transcriptional regulatory network.

The yeast *S.cerevisiae* has a popular model system for cell-cycle analysis. Centrifugal elutriation can be employed to obtain cells with uniform size. Since cell size is correlated to cell-cycle phase, the elutriated cells are synchronized with respect to position in the cycle. However, even though this method introduces relatively small degree of stress to the cells, the synchrony is generally poor, showing only one cell-cycle as demonstrated in our FastNCA and NCA-r results (the samples marked I). Another method to prepare samples for cell-cycle analysis is the block-and-release method, where cells are synchronized initially by blocking them at a certain phase (e.g. the phase using the mating pheromone α-factor or using a temperature-sensitive cdc 15 mutation), and then releasing them for growth. Samples are taken after the release at different times. These two arrest methods pertain to the samples (II), (III), in the analysis of this article, respectively. Both these block-and-release type of methods give cells synchronized for two cycles, but since biochemical agents have to be used, certain artifacts might be introduced in the samples. A detailed analysis of all the related factors is needed to have a good estimate of the TFs activities involved in this or other network component analysis, but for a general check on the consequences of the Fast NCA and NCA-r studies, the following analysis based on a comparison of these consequences with the high-throughput study of Lee et al. (2002) is a useful step at this stage.

Lee et al. (2002) gave the phase(s) of the cell-cycle during which each TF is activated. The four phases are roughly divided equally to span the circle which represents one cell-cycle. The combined experimental and computational method at that time only provided the periods of activities without knowledge about waveform variations of the activities. Depending on the method of arrest preparation, not only the total cell-cycle time varies in general, the length of each phase also varies. Notably the S phase is particularly short in real time. According to Figure 3 of this article, we have measured the four phases of the first cycle pertaining to each of the three preparations (I, II, III) and have determined the relative lengths of phases which span the circle in Figure 4a–c, respectively. We have then transformed the TFA data of Lee et al. (2002) according to our scales and have drawn the arcs to represent the extends of the TFA spans of 33 specimens in Figure 4a–c. Entries to Table 2 (Panel a–c) correspond, respectively to the three methods of sample preparation for cell synchronization and arrest stated in the text above. Row three of Table 2 (Panel a–c) [i.e. (iii)] give the phase(s) during which each TF is active according to Figure 4a–c. The entry (G1, S) in Table 2 represents the information that the activity period covers about the second half of the G1 to the first half of the S phase. The symbol (narrow) simply emphasizes that the activity period covers only a small proportion of the related phase. To simplify the discussion, we only enter the peak position of the activity-time trace for each TF with respect to difference phases based on the result calculated using the FastNCA and NCA-r methods (i.e. Fig. 3). The term ‘noise’ indicates that we cannot identify clear cyclic variation of activity within the phase domain based on established evidence. The symbol with a question mark represents an ambiguous situation of agreement. We observe from the entries in Table 2 (Panel a–c) that there are 23/33 clear agreements between the results, as represented by peak positions, calculated using FastNCA and the phase period deduced by Lee et al. (2002). The total clear agreements between rows (ii) and (iii) is only 14/33. We must remark that the above analysis only provides a rough comparison.

For the sub-network analysis, two NCA algorithms, gNCA (Tran et al., 2005) and FastNCA, are applied to estimate the TFAs for each of the four sub-networks. The result by gNCA (figure not shown here but can be found on our website) for most TFAs is very similar to the result presented in Yang et al. (2005). Figure 5 represents the such TFA plots applying the FastNCA method. When we compare this Figure 5 and Figure 3 of Yang et al. (2005), we notice that the charts of Ace2, Mbp1, Ndd1, Stb1, Swi4 and Swi5 from these two figures are...
basically in line. The other charts pertaining to Fkh1, Fkh2, Mcm1, Skn7 and Swi6 do not agree very well. Such a result is explainable: (1) because there is a slight difference in gene numbers of the four sub-networks used in these two studies and (2) the FastNCA and gNCA methods are different and it is expected some results are different also. Such a rough comparison however supports the statement that the two algorithms are workable in the type of analysis demonstrated and we might turn our attention in this article on the merits of the methodology introduced here. We can also see that for most plots in Figure 5, reconstructions from different sub-networks show good agreement, which suggests that the estimate by FastNCA is quite reliable.

### 5 DISCUSSIONS AND CONCLUSIONS

Expression of genes are regulated by TFs which themselves are also products of the expression of some genes. Microarray technologies enable the simultaneous measurement of all mRNA transcripts in a cell and thus make it possible to reconstruct the gene regulatory network from microarray data (Gardner and Faith, 2005). The network can be approximated by a static linear multi-input multi-output (MIMO) model. Efforts using such model for gene network analysis include singular value analysis (Alter et al., 2000), independent component analysis (Lee and Batzoglou, 2003; Liebermeister, 2002), and many others (Li et al., 2007; Sabatti and James, 2006; Yu and Li, 2005). However, in general SVD and ICA do not correctly reconstruct the gene-regulatory network since the underlying assumptions cannot be satisfied (Liao et al., 2003; Chang et al., 2006, 2007). It is then noted that some a priori information can be utilized to help solve the inverse problem. The fact that in a gene-regulatory network each gene is regulated by only a limited subset of the TFs is exploited by Liao et al. (2003) to develop the network component analysis (NCA) algorithm, with the finding that such a sparse network topology may satisfy the NCA criteria required by the algorithm. The prior information on the network topology can be obtained by genome wide location analysis through ChIP experiments (Lee et al., 2002) and if such ChIP results are not available, it is still possible to infer the topology from the microarray data (Brynildsen et al., 2007).

However, the NCA formulation in Liao et al. (2003) is a bilinear least squares problem and thus is non-convex and has local solutions, and it is not uncommon to have convergence problem. Though in Tran et al. (2005) Tikhonov regularization is introduced to make the problem well conditioned, it degrades the performance, further slows down the convergence, and does not completely overcome the limitation.

### Table 2. TFAs derived from (i) FastNCA, (ii) NCA-r and (iii) Lee et al. 2002, and agreement between (i) and (iii), (ii) and (iii)

<table>
<thead>
<tr>
<th>TFs</th>
<th>Acc2</th>
<th>Fkh1</th>
<th>Fkh2</th>
<th>Mbp1</th>
<th>Mcm1</th>
<th>Swi4</th>
<th>Swi5</th>
<th>Swi6</th>
<th>Stb1</th>
<th>Ndd1</th>
<th>Skn7</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) Synchronization method I: elutriation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FastNCA</td>
<td>G1/S</td>
<td>G2</td>
<td>S/G2</td>
<td>G1/S</td>
<td>G2</td>
<td>G2/S</td>
<td>G2/M</td>
<td>G1</td>
<td>G1/S</td>
<td>G2</td>
<td>G2/M</td>
</tr>
<tr>
<td>NCA-r</td>
<td>G1/S</td>
<td>S</td>
<td>S/G2</td>
<td>G1/S</td>
<td>G2</td>
<td>G2/S</td>
<td>G2/M</td>
<td>G1</td>
<td>G1/S</td>
<td>G2</td>
<td>G2/M</td>
</tr>
<tr>
<td>Lee</td>
<td>G1</td>
<td>S/G2?</td>
<td>G2</td>
<td>S/G2</td>
<td>G1</td>
<td>G1/S</td>
<td>G1/S</td>
<td>G2</td>
<td>G1/S</td>
<td>G2</td>
<td>G1/S</td>
</tr>
<tr>
<td>i versus iii</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ii versus iii</td>
<td>✓</td>
<td>?</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>✓</td>
<td>x</td>
</tr>
</tbody>
</table>

| (b) Synchronization method II: α-factor arrest |
| FastNCA | G2/M | G2 | S/G2 | G1 | G2 | S | M/G1 | M/G1 | G1 | G2/M | G2/M |
| NCA-r | G2/M | noise | M? | G1 | M/G1 | S | G2 | G1/S | S | G2/M | noise |
| Lee | G1 | S/G2 | G2 | S/G2 | G1 | G1/S | G1/S | G2 | G1/S | G2 | S |
| i versus iii | x | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ? | x | ✓ | ✓ |
| ii versus iii | x | x | ? | ✓ | x | ✓ | x | x | ✓ | x | x |

| (c) Synchronization method III: arrest of cdc15 temperature sensitive mutant |
| FastNCA | G2/M | G2 | S | G1 | G2 | G1 | M/G1 | noise | G1 | G2/M | noise |
| NCA-r | M,G2 | noise | noise | M,G1 | M,G1 | G1/S | G2 | noise | noise | G2 | S |
| Lee | G1 | G2 | S,G2 | G2 | G2 | G1 | G1 | G1/S | G2 | S |
| i versus iii | x | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ? |
| ii versus iii | x | x | x | x | x | ✓ | x | x | x | ✓ | ✓ |
FastNCA method, that the peak activities of the key activators are quite in line with that deduced from an independent study, which provided only broad activity periods over the cycle. Further improvement of the accuracy of FastNCA, together with close examination of the properties of phase arrest over the cycles in samples prepared by different techniques (like I, II, III mentioned in this and other articles) might clarify the meaning of the peculiar activator activity variation of some of the key activators (like Skn 7). At the present stage, we can only describe some of the results from FastNCA and NCA-r as ‘noises’, as listed in Table 2 (Panel a-c) here. Just for cell-cycle analysis alone, it is of fundamental importance to know in which part of the cycle, and further, in which portion of a certain phase, that the activity of each activator peaks.

We must notice that to apply FastNCA for TFA reconstruction from microarray data, as for the original NCA algorithm in Liao et al. (2003), the three NCA criteria must be strictly satisfied. It implies that the S matrix must be of full-row rank and when we remove any one TF and its influenced genes the resulting sub-network must have a connectivity matrix of full-column rank. The FastNCA algorithm cannot account for simultaneous constraints in A and S with additional assumptions on the structure of S, as was accounted for in gNCA (Tran et al., 2005). The requirement on the rank of S has also been conditionally relaxed in Galbraith et al. (2006), which cannot be accounted for by FastNCA in its current form. How to account for the aforementioned relaxed conditions in FastNCA would be an area for further studies.

ACKNOWLEDGEMENTS

The authors would like to thank the anonymous reviewers for their helpful comments.

Funding: This work is supported in part by the University of Hong Kong CRCG under Small Project Funding.

Conflict of Interest: none declared.

REFERENCES


APPENDIX

Given a matrix $H$, the projection matrix $P_H$, which projects a vector to the subspace spanned by the columns of $H$, is given by

$$P_H = HH^T,$$  \hspace{1cm} (A1)

where $H^T$ is the pseudo inverse of $H$, and if $H$ is a tall matrix having full-column rank, then

$$H^T = (H^TH)^{-1}H^T.$$  \hspace{1cm} (A2)

Denote by $P_{H^T}$ be the projection matrix that projects a vector to the space orthogonal to the subspace spanned by the columns of $H$. Then, we have

$$P_{H^T} = I - P_H,$$  \hspace{1cm} (A3)

where $I$ is the identity matrix with conformal dimension.

It can be easily checked that

$$P_{H^T}H = H,$$  \hspace{1cm} (A4)

$$P_H H^T = 0.$$  \hspace{1cm} (A5)

Furthermore, if $D$ has full-row rank,

$$P_{H^T} = (HD)(HD)^T = P_H.$$  \hspace{1cm} (A6)