Bioinformatics Advance Access published May 15, 2007

Gene Expression

Biological Network Mapping and Source Signal Deduction

Mark P. Brynildsen[1], Tung-Yun Wu[2], Shi-Shang Jang[2], and James C. Liao[1,*]

[1]Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, CA 90095, USA, [2]Department of Chemical Engineering, National Tsing-Hua University, Hsinchu 30043, Taiwan, R.O.C.

Associate Editor: Dr. Chris Stoeckert

ABSTRACT

Motivation: Many biological networks, including transcriptional regulation, metabolism, and the absorbance spectra of metabolite mixtures, can be represented in a bipartite fashion. Key to understanding these bipartite networks are the network architecture and governing source signals. Such information is often implicitly imbedded in the data. Here we develop a technique, Network Component Mapping (NCM), to deduce bipartite network connectivity and regulatory signals from data without any need for prior information.

Results: We demonstrate the utility of our approach by analyzing UV-vis spectra from mixtures of metabolites and gene expression data from *Saccharomyces cerevisiae*. From UV-vis spectra, hidden mixing networks and pure component spectra (sources) were deduced to a higher degree of resolution with our method than other current bipartite techniques. Analysis of *S. cerevisiae* gene expression from two separate environmental conditions (zinc and DTT treatment) yielded transcription networks consistent with ChIP-chip derived network connectivity. Due to the high degree of noise in gene expression data the transcription network for many genes could not be inferred. However, with relatively clean expression data, our technique was able to deduce hidden transcription networks and instances of combinatorial regulation. These results suggest that NCM can deduce correct network connectivity from relatively accurate data. For noisy data, NCM yields the sparsest network capable of explaining the data. In addition, partial knowledge of the network topology can be incorporated into NCM as constraints.

Availability: Algorithm available on request from the authors. Soon to be posted on the web, http://www.seas.ucla.edu/~liaoj/

Contact: liaoj@ucla.edu

Supplementary Information: at *Bioinformatics* online.

1 INTRODUCTION

Bipartite network analyses, such as Singular Value Decomposition (SVD), Independent Component Analysis (ICA), Network Component Analysis (NCA) and a number of other techniques, have been successfully used to study data from biological systems (Alter *et al.*, 2000; Holter *et al.*, 2000; Bussemaker *et al.*, 2001; Liebermeister, 2002; Yeung *et al.*, 2002; Alter *et al.*, 2003; Lee and Batzoglou, 2003; Gao *et al.*, 2004; Beal *et al.*, 2005; Tran *et al.*, 2005; Yang and Liao, 2005; Yang *et al.*, 2005; Galbraith *et al.*, 2006; Li *et al.*, 2006; Sanguinetti *et al.*, 2006). In part, this can be attributed to the existence of bipartite networks in many natural systems, including transcriptional regulation, metabolism, and the absorbance of compounds. Bipartite networks consist of two tiers, source and output, connected by edges, where each edge represents an influence of a source on an output. In transcriptional regulation a bipartite network can be formed by linking the expression of a gene (output) to the activity of its regulators (sources), while in the case of absorbance, the spectrum of a mixture (output) is connected to the pure component spectra (source) of its constituents. For data created from bipartite systems, two features are paramount to interpretation. These features are the network topology and the source signals that pass through the network to generate the observed data.

All bipartite network analyses attempt to identify the network and source signals hidden within data. However, since the factorization is ill-defined there are many ways to decompose the data. In general, bipartite network analyses can be classified into two types, confirmatory and exploratory. In confirmatory analyses prior knowledge of the system is used to constrain a solution. This knowledge can take the form of: 1) postulated network architectures, such as in the case of NCA, Bayesian Sparse Hidden Components Analysis, and a variety of state-space models (SSM) (Liao *et al.*, 2003; Galbraith *et al.*, 2006; Li *et al.*, 2006; Sabatti and James, 2006; Sanguinetti *et al.*, 2006), 2) completely defined networks, such as in the case of REDUCE and MA-Networker, (Bussemaker *et al.*, 2001; Gao *et al.*, 2004), or 3) knowledge of the hidden source signals, such as in the case of generalized Network Component Analysis (gNCA) when used with transcription factor knock-out experiments (Tran *et al.*, 2005). While confirmatory methods have the benefit of providing solutions consistent with *a priori* system information, and are thus readily interpretable, their applicability is limited to situations where such information is available. Under circumstances where only the observed data is known, such as expression data from poorly characterized organisms or from a well characterized organism in a poorly characterized environment, confirmatory approaches cannot be used. However, exploratory approaches are designed explicitly for situations where only the observed data is known. In these methods assumptions that are not system-specific are used to constrain a solution. Examples of these assumptions are: 1) orthogonality of the source signals (SVD) (Alter *et al.*, 2000; Holter *et al.*, 2000; Alter *et al.*, 2003), 2) statistical independence of the source signals (ICA) (Liebermeister, 2002; Lee and Batzoglou, 2003), and 3) structural simplicity of the network (Exploratory Factor Analysis (EFA) and Beal *et al.* 2005). Typically, the interpretability of SVD and ICA solutions are a concern. This stems for the fact that orthogonality and statistical independence often lack physical meaning since they are mathematically defined. In fact, it is generally accepted that

*To whom correspondence should be addressed.

physically meaningful source signals are most likely oblique (Thurstone, 1947; Browne, 2001). In addition, SVD and ICA assume that the hidden network topology is fully connected, and thus every source signal could contribute to every output. This is not an appropriate assumption for systems such as transcriptional regulation where it is accepted that transcription networks are generally sparse. Exploratory Factor Analysis somewhat alleviates these issues by searching for a rotation of a factorization that maximizes a user-specified sparsity criterion, under the guidelines that the final source signals be orthogonal or oblique (also user-specified) (Browne, 2001). However, EFA has had difficulty with data where the complexity of the network exceeds that of maximal sparsity (one connection per output to the source layer) (Browne, 2001). The SSM of Beal *et al.* 2005 also focuses on network simplicity, but approaches the problem from a probabilistic perspective. Due to the large degree of data replication required by this method, and the existence of degeneracy in the deduced source signals (same network, different source signals/hidden variables), it is not of the same class as SVD, ICA, and EFA. With these issues in mind we sought to develop an exploratory technique based on structural network simplicity that requires a minimal amount of user specified information and can deduce true networks that exceed maximal sparsity. We have based our method on principles developed in (Brynildsen *et al.*, 2006) and (Liao *et al.*, 2003), and termed it Network Component Mapping (NCM).

By utilizing the concepts of network versatility, nonversatility, and NCA we have created a method that assumes nothing about the nature of the source signals beyond linear independency, considers the network connectivity a key feature of analysis, and only requires users to specify a threshold for edge significance that can easily be varied to obtain an idea of the solution landscape. Network Component Mapping searches for the sparsest network structure capable of explaining the data under a given noise threshold. We demonstrate the utility of NCM by analyzing UV-Vis absorbance spectra from metabolite mixtures and gene expression data from *Saccharomyces cerevisiae*. Analysis of UV-Vis spectra requires knowledge of pure component spectra for identification and quantification. However, for some compounds chemical standards are difficult to obtain due to purification, stability, or other issues. Analysis of mixtures of these types of compounds has proven particularly challenging. Here we effectively identified the mixing network and source spectra in systems with and without the presence of chemical standards, showcasing that standards are unnecessary when analyzing UV-Vis spectra with NCM. For gene expression analysis we realized that verification of the deduced source signals and transcription networks is difficult. To validate the performance of NCM on gene expression data we chose to compare the deduced transcription network with that obtained from ChIP-chip binding assays (Lee *et al.*, 2002; Harbison *et al.*, 2004), a technique that has been employed previously (Qian *et al.*, 2003). However, transcription factor binding is environmentally dependent and binding does not always confer regulation (Gao *et al.*, 2004; Harbison *et al.*, 2004; Boulesteix and Strimmer, 2005; Papp and Oliver, 2005; Brynildsen *et al.*, 2006). With this in mind the Gibbs sampler of (Brynildsen *et al.*, 2006) was employed to screen for genes with consistent expression and ChIP-chip derived connectivity data. Genes deemed consistent by the Gibbs sampler, possessed a ChIP-chip derived transcription network capable of explaining their expression. The expression of these genes was analyzed with NCM to demonstrate that NCM can deduce experimentally derived (ChIP-chip) transcription networks from expression data.

Lastly, it is important to note that for noisy data NCM deduces the sparsest network that can explain the data, and if partial network knowledge is available it can be incorporated into NCM such that the deduction is the sparsest network consistent with prior information.

2 METHODS

2.1 Background

2.1.1 Bipartite Networks

Network Component Mapping deals with uncovering hidden network connectivity and source signals from the output of bipartite networks. A bipartite network represents an output $e_i(t)$ by the linear mixing of sources, $p_j(t)$, through a mixing rule described by:

$$e_i(t) = \sum_{j=1}^{L} a_{ij} p_j(t) \tag{1}$$

where a_{ij} are the connectivity strengths. The mixing rule can be written in matrix form:

$$\mathbf{E} = \mathbf{AP} \tag{2}$$

where $\mathbf{E}$ is the output data (NxM), $\mathbf{A}$ is the matrix of network connectivity strengths (NxL), and $\mathbf{P}$ is the collection of source signals (LxM). Bipartite networks can further be generalized by considering only the connectivity pattern of matrix $\mathbf{A}$:

$$\mathbf{Z_A} = \left\{ \mathbf{A} \in \mathbb{R}^{NxL} \mid a_{ij} = 0, \text{ for a given set of } (i, j) \right\} \tag{3}$$

where the values of the nonzero a_{ij} are left unconstrained and can take on any value, positive, negative, or zero. For the purpose of this paper, networks with varying connectivity strengths but the same connectivity pattern, $\mathbf{Z_A}$, will be discussed identically.

2.1.2 Versatility and NCA-compliance

Network Component Mapping utilizes the concepts of bipartite network versatility and NCA-compliance (Liao *et al.*, 2003; Brynildsen *et al.*, 2006). Versatility is a property solely defined by the network topology. A method to check if a network is versatile can be found in Brynildsen *et al.* 2006. Consider a network with N outputs and L sources. If the network is versatile it can explain any data within $\mathbb{R}^L$. In other words, it can describe any dataset with N outputs and $\leq L$ non-zero singular values perfectly, regardless of the generating network. If there is noise and there are $\geq L$ non-zero singular values, a versatile network can describe the best rank L approximation of the data. Due to this ability, all versatile networks of the same size are equivalent in terms of their ability to describe

data. Versatile networks have a range of edge densities, with fully connected networks existing on one side of the spectrum and minimal versatile networks on the other. Minimal versatile networks are those topologies that will no longer be versatile if a single edge is lost. These networks are used to initialize NCM, and the procedure will be described in the next section. It is important to note that if the underlying network responsible for a dataset is versatile it cannot be deduced from the output data. This results from the ability of all versatile networks to explain any data within $\mathbb{R}^L$. However, since versatile networks are fairly dense (see Brynildsen *et al.* 2006 for details) the majority of networks are non-versatile. Indeed, transcription networks are extremely sparse, and thus certainly non-versatile. This makes transcription networks good candidates for deduction from gene expression data.

NCA-compliance deals with the uniqueness of a particular solution. A series of criteria define NCA-compliance, and these can be found in Liao *et al.* 2003. The criteria involve both network topological constraints on $\mathbf{A}$, and rank requirements on $\mathbf{A}$ and $\mathbf{P}$. These criteria are used in NCM to ensure that every step of the algorithm provides a unique solution up to a scaling factor (see Liao *et al.* 2003 for details). We recognize that the true underlying network for a given dataset may not be NCA-compliant, however, without requiring our solution to be NCA-compliant, another more artificial constraint such as orthogonality or statistical independence would need to be used to obtain uniqueness.

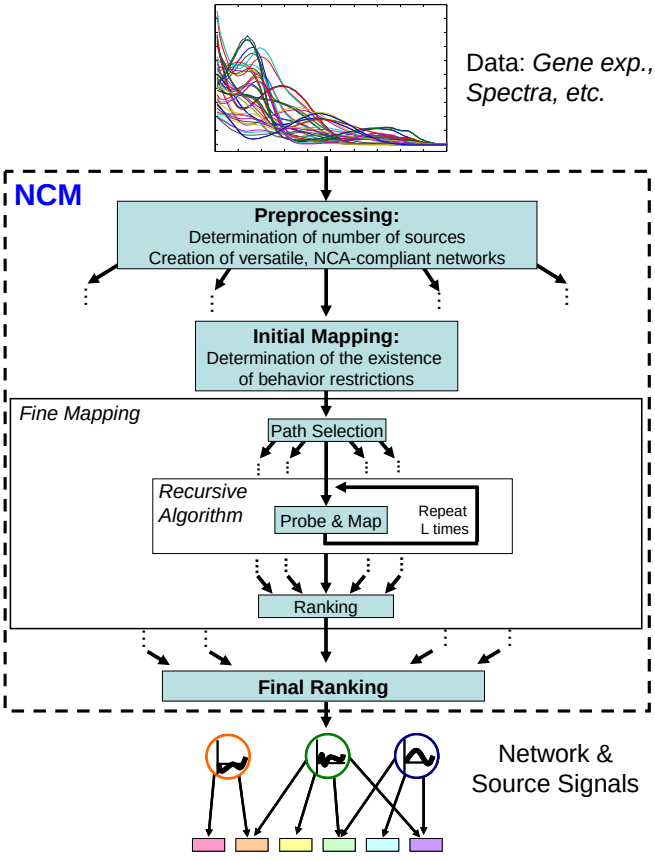

Fig.1: Schematic of NCM algorithm

2.2 Network Component Mapping Overview

Network Component Mapping is based upon the principles of network versatility and nonversatility described in (Brynildsen *et al.*, 2006). The technique follows the flow diagram shown in Figure 1. The purpose of NCM is to deduce the hidden network structure and source signals responsible for a given set of data. This is typically an ill-posed problem since the factorization in Eq. (2) is non-unique. Any number of invertible LxL matrices, $\mathbf{Y}$, could be used to transform the factorization in Eq. (2):

$$\mathbf{E} = \mathbf{AYY}^{-1}\mathbf{P} = \hat{\mathbf{A}}\hat{\mathbf{P}} \qquad (4)$$

where $\hat{\mathbf{A}}$ does not have to equal $\mathbf{A}$ or be related to it by a scalar , and $\hat{\mathbf{P}}$ does not have to equal $\mathbf{P}$ or be related to it by a scalar. Therefore, constraints need to be placed on the system to identify a unique solution. The constraint NCM uses involves network simplicity. Under the premise that the sparsest network is most likely the true network, NCM searches for the sparsest NCA-compliant topology capable of describing the data given a certain noise level. The assumption that the sparsest network is most likely the true network has been used previously (Yeung *et al.*, 2002), and justification comes from the empirical principle of parsimony that states the number of parameters in a model should not increase unless a significant improvement to fit is observed (Akaike, 1987). In practice this translates into, given a number of models that all fit the data similarly the one chosen to represent the system should be the one with the least number of parameters. In our case, this would be the sparsest network.

2.3 Preprocessing

The algorithm begins by prompting the user to input the data, and if known the number of sources/components. If the number of sources is unknown a preprocessing step is initiated which utilizes SVD to determine how many sources there are by the number of significant singular values. In addition, model selection criteria such as Akaike Information Criterion (AIC), Schwarz/Bayesian Information Criterion (SIC), and Risk Inflation Criterion (RIC) could be used to determine the number of factors (Wu *et al.*, 2004). After the number of sources has been determined, L, the algorithm begins by generating a series of initial guess networks, $\mathbf{Z}_{\mathbf{ig}}$ (NxL), formulated at random but required to be both versatile and NCA-compliant. We require $\mathbf{Z}_{\mathbf{ig}}$ to be versatile so that we do not introduce any artificial bias into our analysis (Brynildsen *et al.*, 2006), and we require $\mathbf{Z}_{\mathbf{ig}}$ to be NCA-complaint because we desire a unique solution at every stage of our algorithm (Liao *et al.*, 2003). The only networks that are both versatile and NCA-compliant are those that contain the minimal versatile connectivity (Brynildsen *et al.*, 2006).

The minimal versatile connectivity defines a class of networks where all members contain $L(L-1)$ missing edges, although at different positions, and are versatile. There are many choices of network that contain the minimal versatile connectivity that can be used for $\mathbf{Z}_{\mathbf{ig}}$. Since the true network, $\mathbf{Z}_{\mathbf{tr}}$, is unknown and cannot be deduced unless a $\mathbf{Z}_{\mathbf{ig}} \subset \mathbf{Z}_{\mathbf{tr}}$ (the zero positions in $\mathbf{Z}_{\mathbf{ig}}$ are a subset of those in $\mathbf{Z}_{\mathbf{tr}}$), a series of $\mathbf{Z}_{\mathbf{ig}}$ is used to ensure that in at least one instance $\mathbf{Z}_{\mathbf{ig}} \subset \mathbf{Z}_{\mathbf{tr}}$.

2.4 Initial Mapping

Once a $\mathbf{Z}_{\mathbf{ig}}$ has been randomly selected it enters an initial mapping procedure. The procedure is based upon the relationship between NCA and SVD:

$$\mathbf{E} = \mathbf{USV}^{\mathbf{T}} = \mathbf{AP} \tag{5}$$

where $\mathbf{E}$ is the output data (NxM), $\mathbf{A}$ is the network (NxL) defined by the zero pattern $\mathbf{Z_A}$, $\mathbf{P}$ is the collection of source signals (LxM), $\mathbf{S}$ is the diagonal matrix (LxL) of the first L singular values of $\mathbf{E}$ oriented in decreasing order, and $\mathbf{U}$ (NxL) and $\mathbf{V}$ (MxL) are unitary matrices of the right and left singular vectors of the elements in $\mathbf{S}$. The component matrices of the two decompositions can be related as follows:

$$\mathbf{A} = \mathbf{UX} \tag{6}$$

$$\mathbf{P} = \mathbf{X}^{-1}\mathbf{SV}^{\mathbf{T}} \tag{7}$$

where $\mathbf{X}$ (LxL) is an invertible matrix that relates $\mathbf{U}$ to $\mathbf{A}$ and $\mathbf{SV}^{\mathbf{T}}$ to $\mathbf{P}$. For a versatile, NCA-compliant network, an invertible $\mathbf{X}$ can found to satisfy Eq. (6) and (7) for any data, $\mathbf{E}$. The first step in the initial mapping procedure is to do just that.

We recognize that $\mathbf{X}$ can be calculated from either Eq. (6) or (7). Since nothing is known about the values of $\mathbf{P}$, and $\mathbf{A}$ is characterized by $\mathbf{Z_A}$ (zero locations are known) we use Eq. (6) to calculate $\mathbf{X}$. We transform Eq. (6) into:

$$\begin{bmatrix} \mathbf{A}_{\mathbf{c}_1} \\ \vdots \\ \mathbf{A}_{\mathbf{c}_L} \end{bmatrix} = \begin{bmatrix} \mathbf{U} & & \mathbf{0} \\ & \vdots & \\ \mathbf{0} & & \mathbf{U} \end{bmatrix} \begin{bmatrix} \mathbf{X}_{\mathbf{c}_1} \\ \vdots \\ \mathbf{X}_{\mathbf{c}_L} \end{bmatrix} \longrightarrow \quad \mathbf{A_c} = \mathbf{U_c X_c} \tag{8}$$

where $\mathbf{A}_{\mathbf{c}_i}$ is the i^{th} column of $\mathbf{A}$, and $\mathbf{X}_{\mathbf{c}_i}$ is the i^{th} column of $\mathbf{X}$. By collecting all of the zeros in $\mathbf{A_c}$ we can obtain the workable equation:

$$\begin{bmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{U}_{\mathbf{c}_1^r} & & \mathbf{0} \\ & \vdots & \\ \mathbf{0} & & \mathbf{U}_{\mathbf{c}_L^r} \end{bmatrix} \begin{bmatrix} \mathbf{X}_{\mathbf{c}_1} \\ \vdots \\ \mathbf{X}_{\mathbf{c}_L} \end{bmatrix} \longrightarrow \quad \mathbf{0} = \mathbf{U}_{\mathbf{c}^r} \mathbf{X_c} \tag{9}$$

where $\mathbf{U}_{\mathbf{c}_i^r}$ is the reduced form of $\mathbf{U}$ which corresponds to the zero entries in the i^{th} column of $\mathbf{A}$.

We know that the initial mapping procedure uses $\mathbf{Z_{ig}}$ for $\mathbf{Z_A}$, which means there will be $L(L-1)$ zeros in $\mathbf{A}$ and in particular $L-1$ zeros per $\mathbf{A}_{\mathbf{c}_i}$ (Brynildsen *et al.*, 2006). Since every $\mathbf{X}_{\mathbf{c}_i}$ has L unknowns and every $\mathbf{A}_{\mathbf{c}_i}$ has $L-1$ zeros, the null space for all $\mathbf{U}_{\mathbf{c}_i^r}$ will exist and non-trivial solutions for all $\mathbf{X}_{\mathbf{c}_i}$ will exist. In addition, since most data has some degree of noise, the nullity of $\mathbf{U}_{\mathbf{c}_i^r}$ will be 1 and all solutions of $\mathbf{X}_{\mathbf{c}_i}$ will be related to one another by a scaling factor. Therefore, a null space calculation can be used to determine $\mathbf{X}$ uniquely up to a scaling factor that works per column of $\mathbf{X}$. Since $\mathbf{A}$ is NCA-compliant this does not present a problem because the columns of $\mathbf{A}$ are uniquely determined up to a scaling factor that works per column of $\mathbf{A}$ (Liao *et al.*, 2003).

Once $\mathbf{X}$ and $\mathbf{A}$ have been determined a trimming procedure is performed (Supplementary Information section 2.3.1). Trimming of an edge occurs when its source signal contribution is less than a user-specified threshold. A variety of model selection criteria including AIC, SIC, RIC, and cross validation (CV) were tested against the performance of the threshold parameter. However, only threshold trimming proved effective with our data (see Supplemental Information section 2.3.2).

Not all initial mappings yield a trimmed network. If a particular $\mathbf{Z_{ig}}$ cannot elicit any non-versatile data signatures, the initial mapping will simply yield $\mathbf{Z_{ig}}$ as a result. This is an issue because in versatile networks the network connectivity does not carry any physical significance, since the edges may be rearranged in many different ways without impacting the system (Brynildsen *et al.*, 2006). Therefore to continue onto the next stage of the algorithm the following two criteria must be met after trimming: 1) every source/component (column of $\mathbf{Z_{ig}}$) has had at least one edge from it trimmed, resulting in every source being non-versatile (see Supplemental Information section 4.2 for details) 2) the resultant network ($\mathbf{A}$) and source signal matrix ($\mathbf{P}$) are NCA-compliant. We require every source to be non-versatile so that the position of zeros within every column would have significance, and we require $\mathbf{A}$ and $\mathbf{P}$ to be NCA-complaint to ensure that the solution is unique. If these two criteria are not met, the algorithm chooses another $\mathbf{Z_{ig}}$ and the initial mapping procedure is conducted again (Note: due to complexities in gene expression data necessitating analysis of small datasets combined with the presence of high noise levels, the first of these criteria was relaxed to the uncovering of a single zero for the whole network instead of per column. The criteria may also be neglected with the possible cost of a larger number of iterations necessary for deduction).

2.5 Fine Mapping

Although the initial mapping procedure identifies portions of the network map that are unnecessary, it may not identify all of the non-essential sections. Hence, the newly trimmed network must enter a fine mapping procedure which will further probe the data for inherent constraints. The fine mapping procedure has three components, which are path selection, recursive algorithm, and ranking.

The fine mapping procedure does not utilize a null space calculation as the initial mapping procedure does. In theory, if the data was devoid of noise and error a null space calculation could be utilized in the fine mapping procedure. Recall that a versatile network could satisfy Eq. (6) and (7) for any data. This includes any noise present (see Supplemental Information section 4.1). Nonversatile networks, on the other hand, can only satisfy Eq. (6) and (7) for data that contain their signatures. Therefore, a null space calculation could be used with non-versatile networks if the appropriate signatures are present in the data. However, any addition of noise to the data will obscure those signatures, resulting in the destruction of the null space of the columns of $\mathbf{X}$. This complication has been noted previously (Brynildsen *et al.*, 2006), and leads to the necessity of path selection.

The path selection process allows calculation of the nonzero entries of $\mathbf{A}$ without the use of a null space calculation. By taking advantage of the scaling rules present in NCA, we are able to select at random a nonzero element from every column of $\mathbf{A}$ and set that to 1, transforming Eq. (9):

$$\begin{bmatrix} \mathbf{c}_1^r \\ \vdots \\ \mathbf{c}_L^r \end{bmatrix} = \begin{bmatrix} \mathbf{U}_{\mathbf{c}_1^r} & & \\ & \vdots & \\ & & \mathbf{U}_{\mathbf{c}_L^r} \end{bmatrix} \begin{bmatrix} \mathbf{X}_{\mathbf{c}_1} \\ \vdots \\ \mathbf{X}_{\mathbf{c}_L} \end{bmatrix} \longrightarrow \quad \mathbf{c}^r = \mathbf{U}_{\mathbf{c}^r} \mathbf{X_c} \tag{10}$$

where $\mathbf{c}_i^r$ is the reduced form of the i^{th} column of $\mathbf{A}$, and $\mathbf{U}_{c_i^r}$ is the reduced form of $\mathbf{U}$ which corresponds to the i^{th} column of $\mathbf{A}$. The reduced form, $\mathbf{c}_i^r$, is the collection of zeros and a single nonzero entry from the i^{th} column of $\mathbf{A}$, while the reduced form, $\mathbf{U}_{c_i^r}$, are those rows of $\mathbf{U}$

associated with the entries of $\mathbf{c}_i^r$ through Eq. (8). The actual selection of non-zero entries to place in $\mathbf{c}^r$ is random and is termed path selection. Path selection provides both a nontrivial solution for $\mathbf{X}$, and a set of permanently present edges. Since these edges are selected at random and could possibly be absent from $\mathbf{Z}_{\mathbf{tr}}$, the path selection process must be performed multiple times for every network that enters the fine mapping procedure.

While the path selection process will provide a non-trivial solution for $\mathbf{X}$, it will not uncover any additional behavioral constraints. To detect any further non-versatile signatures the network is passed to the recursive algorithm. The recursive algorithm systematically probes for non-versatile signatures by deleting network edges one by one with subsequent evaluation by Eq.s (6), (7), and (10), after which another trimming procedure is conducted. Details of the recursive algorithm can be found in the Supplementary Information section 2.2. After completion of the recursive algorithm a single network from every path selection is provided to the ranking procedure.

The ranking procedure consists of two tiers. The first tier ranks the networks by the number of remaining edges. The network with the least number of edges is chosen as the NCM output, $\mathbf{Z}_{\mathbf{NCM}}$, unless there is a tie. If there are multiple networks with the same edge density, the residual error, as measured by the Frobenius norm, is used as a tiebreaker. Hence, the most sparse network with the smallest residual error is then used to determine the complimentary source signals, and both are reported as the result from that particular $\mathbf{Z}_{\mathbf{ig}}$.

2.6 Final Ranking

The final ranking procedure is identical to the ranking procedure of the fine mapping algorithm. The only exception is that the final ranking procedure is being used to discern $\mathbf{Z}_{\mathbf{tr}}$ from a series of trimmed networks from different $\mathbf{Z}_{\mathbf{ig}}$'s, while the fine mapping ranking procedure attempts to discern $\mathbf{Z}_{\mathbf{tr}}$ from networks created from different paths from the same $\mathbf{Z}_{\mathbf{ig}}$.

2.7 Random Processes

NCM relies upon two random processes. These are the initial selection of $\mathbf{Z}_{\mathbf{ig}}$ and the path selection process. To overcome errors instituted by the path selection process (edges selected are not present in $\mathbf{Z}_{\mathbf{tr}}$), the fine mapping procedure is performed multiple times for every $\mathbf{Z}_{\mathbf{ig}}$ (50 here for both spectrum and expression data). This provides a sampling of nonzero entry combinations empirically shown to allow identification of $\mathbf{Z}_{\mathbf{tr}}$. However, the path selection number can easily be changed, and exhibits a negligible effect on computation time compared to the selection of $\mathbf{Z}_{\mathbf{ig}}$. For NCM to converge to $\mathbf{Z}_{\mathbf{tr}}$ the following must be met: 1) $\mathbf{Z}_{\mathbf{ig}} \subset \mathbf{Z}_{\mathbf{tr}}$ and 2) there must be an NCA-compliant path from $\mathbf{Z}_{\mathbf{ig}}$ to $\mathbf{Z}_{\mathbf{tr}}$. If these conditions do not exist in any of the iterations of NCM $\mathbf{Z}_{\mathbf{tr}}$ will not be obtained. These conditions are both determined by $\mathbf{Z}_{\mathbf{ig}}$. The simplest solution is to test a large number of $\mathbf{Z}_{\mathbf{ig}}$, so confidence is high that the conditions had been met. The number of $\mathbf{Z}_{\mathbf{ig}}$ that should be tested to ensure $\mathbf{Z}_{\mathbf{ig}} \subset \mathbf{Z}_{\mathbf{tr}}$ is dependent on a number of factors, and has been discussed in the Supplemental Information section 2.1. However, for very dense networks the number of iterations necessary to obtain a proper selection of $\mathbf{Z}_{\mathbf{ig}}$ randomly could be substantial. Another solution exists if prior knowledge of the network is available. Such knowledge can then be incorporated into $\mathbf{Z}_{\mathbf{ig}}$ to expedite computation. Either way, in general NCM converges to $\mathbf{Z}_{\mathbf{tr}}$ more quickly for sparse networks due to the ease with which a proper $\mathbf{Z}_{\mathbf{ig}}$ may randomly be obtained, and that incorporation of *a priori* system knowledge into the method may decrease computation time.

3 RESULTS

3.1 Spectrum Data

To demonstrate the utility of NCM we constructed two chemical spectra networks with 5 chemical components: creatinine, hypoxanthine, shikimic acid, tryptophan and tyrosine. This was done by creating a series of mixtures and varying the concentrations of particular components in different mixtures. In this framework the 5 pure components populate the source layer, while each mixture represents an output in the output layer. An edge is drawn between an output and source if for that particular mixture the concentration of the source is >0. The first network constructed contained 35 mixtures (outputs) where each output connected to ≥ 2 sources (pure component spectra absent). The second network constructed contained 50 mixtures where each output connected to ≥ 1 source (pure component spectra present). Absorbance of the output spectra were measured from 205-354nm, and our goal was to deduce the network and source signals solely from the output spectra. For comparative purposes the performances of SVD, ICA, orthogonal EFA, and oblique EFA were also evaluated in addition to NCM.

The goal of analyzing the first network was to simply demonstrate the utility of NCM in spectrum analysis and show that NCM does not require chemical standards to successfully deduce the hidden network and source signals. The first system consists of 26 outputs that are two component mixes, and 9 outputs that are 3 component mixes. The network can be visualized in Figure2A, and a plot of the normalized singular values of the spectra is presented in Figure 3A. It is obvious from this plot that there are 5 significant singular values, and thus 5 components were inferred as expected. For the spectrum data AIC, SIC, and RIC all identified more sources than were present, and thus singular values have been adopted in this work to determine the number of sources.

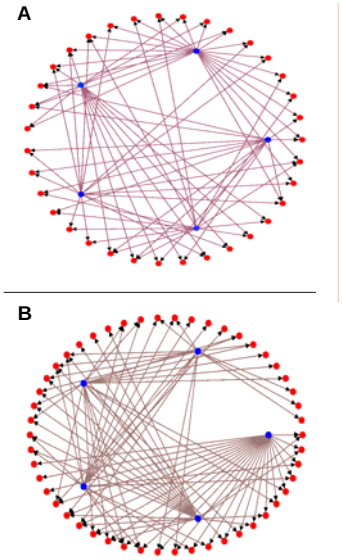

Fig.2: A) Chemical Network 1 (35 mixtures), B) Chemical Network 2 (50 mixtures), blue nodes indicate sources, while red nodes indicate outputs

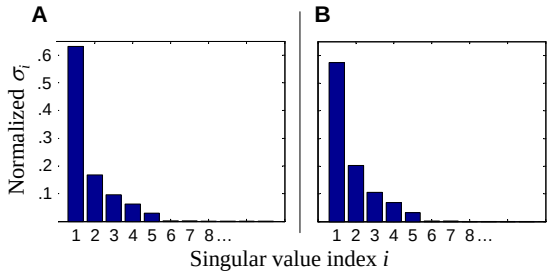

Fig. 3: Plot of singular values for spectrum data, where normalized $\sigma_i = \sigma_i / \sum_{i=1}^{N} \sigma_i$

After using NCM, the true network, $\mathbf{Z_{tr}}$, was determined with a frequency of 1/44 when sampled over 1000 iterations, which means that on average 44 $\mathbf{Z_{ig}}$'s passed to fine mapping were required to obtain $\mathbf{Z_{tr}}$. The correlation coefficient between the real pure-component spectra and the NCM approximations was excellent, with a median of .9998 for the 5 components when compared to triple repeat pure component spectrum data. The difference between the concentrations calculated from analysis with pure component spectra and that obtained from the NCM deduction was maximally 11.1%, with a mean of 1.4%. This example demonstrates the utility of investigating UV spectra with NCM when pure component spectra are not available. This network was also analyzed with SVD, ICA, orthogonal varimax EFA, and oblique promax EFA (see Supplementary Information section 3.1-3). The results of these analyses compared to NCM can be found in Table 1A. Concentrations were not calculated since the networks deduced by the other methods were inaccurate.

	Creatinine	*Tryptophan*	*Tyrosine*	*Shikimic Acid*	*Hypoxanthine*	*Network Accuracy*
A						
SVD	0.36	0.10	0.21	0.95	0.77	49%
ICA	0.58	0.83	0.48	0.80	0.76	55%
EFA (orth)	0.78	0.92	0.96	0.99	0.92	59%
EFA (obl)	0.77	0.93	0.96	0.99	0.90	63%
NCM	1.00	1.00	1.00	1.00	1.00	100%
B						
SVD	0.36	0.09	0.20	0.96	0.77	50%
ICA	0.59	0.83	0.52	0.79	0.76	52%
EFA (orth)	1.00	0.97	0.99	1.00	0.98	73%
EFA (obl)	0.98	0.97	0.99	0.99	0.97	58%
NCM	1.00	1.00	1.00	1.00	1.00	100%

Table 1: Correlation coefficients (CC) and network accuracy (NA) for analysis of A) System 1, B) System 2 spectrum data by different methods (CC, NA as discussed in Supplemental Information section 3.2-3)

The network for the second system contained the first system along with 15 pure component spectra (3 from each component). The network can be seen in Figure2B, and a plot of the normalized singular values is presented in Figure 3B. It is obvious from this plot that there are 5 significant singular values, and thus 5 components. After using NCM, $\mathbf{Z_{tr}}$ was realized with a frequency of 1/11 when sampled over 1000 iterations. The correlation coefficient between the real pure component spectra and the approximated pure component spectra was minimally .9999 and maximally 1.000, with a median of .9999 for the 5 components when compared to triple repeat pure component spectrum data. The con centrations when compared against an analysis performed with the pure component spectra were maximally 6.0% different, with a mean of 0.7%. This example demonstrates that as the sparsity of $\mathbf{Z_{tr}}$ increases, even while the size of the system increases, the number of iterations necessary to obtain the true answer decreases. This can be attributed to the higher likelihood of $\mathbf{Z_{ig}} \subset \mathbf{Z_{tr}}$. In addition, this network was analyzed with SVD, ICA, orthogonal varimax EFA, and oblique promax EFA. The results of these analyses compared to NCM can be found in Table 1B.

3.2 Gene Expression Data

To demonstrate the applicability of NCM for transcriptional regulation transcription networks were deduced from gene expression data. Transcription networks were verified with ChIP-chip derived network connectivity screened for accuracy by the Gibbs sampler developed in (Brynildsen *et al.*, 2006). The Gibbs sampler was a necessary step due to the presence of experimental noise, environmental dependence in regulator binding, and uncorrelation between binding and regulation. Transcription factor activities derived from NCM were not verified with an outside source due to their unavailability. The majority of literature concerned with TFAs deduces them from expression data, resulting in activities subject to the assumptions and biases of a particular method or model. To avoid this artificial comparison we assumed that if NCM deduced the proper transcription networks, appropriate TFAs would likely result. This is evidenced in the results obtained for the chemical spectra networks.

Gene ID	**Stress**	**Regulator(s)**	**Gene ID**	**Stress**	**Regulator(s)**
YAL061W	Zinc	SOK2	*YLL067C*	Zinc	YAP5
YBR115C	Zinc	GCN4	*YLR120C*	Zinc	AFT2
YCL048W	Zinc	SUM1	*YLR299W*	DTT	YAP7
YCR075C	DTT	FKH1	*YLR349W*	Zinc	HSF1
YDL198C	Zinc	GCN4, GLN3	*YLR392C*	Zinc	SMP1
YDL204W	DTT	YAP7	*YLR394W*	Zinc	SMP1
YDR403W	Zinc	SUM1	*YLR461W*	Zinc	AFT2
YER052C	Zinc	GCN4	*YMR053C*	Zinc	PHO2
YER139C	DTT	SWI6	*YMR062C*	Zinc	GCN4
YGL138C	Zinc	SUM1	*YMR149W*	DTT	ROX1
YGL261C	Zinc	AFT2	*YNL141W*	Zinc	GLN3
YGR168C	DTT	MCM1,MGA1	*YNL253W*	Zinc	ZAP1
YHR024C	Zinc	GCN4	*YNL254C*	Zinc	ZAP1
YIL102C	DTT	ROX1	*YNR076W*	Zinc	AFT2
YJL056C	Zinc	ZAP1	*YOL161C*	Zinc	AFT2
YJL161W	Zinc	PHO2	*YPL044C*	DTT	MCM1
YJL223C	Zinc	AFT2	*YPL226W*	DTT	FKH1
YJR067C	DTT	GAT3	*YPL273W*	Zinc	GCN4
YLL064C	Zinc	AFT2	*YPR196W*	Zinc	HSF1
YLL066C	Zinc	YAP5	*YPR197C*	DTT	MGA1

Table 2: NCM deduced transcription networks

In Table 2 we present transcription networks deduced by NCM from gene expression data from *Saccharomyces cerevisiae*. Tran-

scription factors were assigned to genes by aligning NCM-deduced networks with the corresponding ChIP-chip derived connectivities (see Supplemental Information section 3.1). For the networks presented, NCM deduced networks identical to those defined by ChIP-chip (see Supplemental Information section 1.2), therefore, each TF-gene interaction deduced by NCM was validated with ChIP-chip binding data. One network was deduced from expression data obtained during stress from zinc, while the other was deduced from data under reductive stress induced by DTT. An interesting feature to note is that NCM deduced combinatorial regulation in both zinc and DTT experiments. As a comparison PCA, ICA, orthogonal varimax EFA, and oblique varimax EFA were used to deduce transcription networks from the same expression data. The results of these analyses compared to NCM can be found in Table 3.

There are two important features to note about the application of NCM to gene expression data. The first is that the number of experiments (μ-arrays) to be analyzed limits the number of regulators a particular NCM can deduce. The second is that an excess of noise in expression data impacts the resolution with which NCM can deduce transcription networks. These aspects will be addressed in detail within the Discussion.

	Zinc Network Accuracy	DTT Network Accuracy
SVD	81%	71%
ICA	80%	69%
EFA (orth)	99%	96%
EFA (obl)	98%	95%
NCM	100%	100%

Table 3: Comparison of network accuracy deduced by different methods referenced to ChIP-chip connectivity (see Supplemental Information section 3.1 for details).

4 DISCUSSION

Here we have presented NCM, a technique that utilizes concepts from (Brynildsen *et al.*, 2006), NCA, and SVD to reconstruct regulatory networks and source signals from the output of bipartite systems. Network Component Mapping searches for the sparsest network capable of explaining data given a certain noise threshold, under the premise that the sparsest network is most likely the true network. The ability of NCM to deduce hidden networks and source signals has been demonstrated with UV-Vis spectra and gene expression data. This ability was compared to that of other popular bipartite techniques. The performance of NCM was superior to that of other techniques. The extent to which this performance enhancement was dependent on the trimming procedure was explored for both spectrum and expression data. As described in Supplemental Information section 3.4, the performance of EFA becomes comparable to NCM if the true network is very sparse and a large trimming threshold is used. For a detailed discussion on the conceptual differences between EFA and NCM see Supplemental Information section 3.7.

Network Component Mapping deduced all chemical networks exactly, and inferred source signals that were all exceptionally well correlated with pure component spectra. With expression data NCM was able to deduce transcription networks consistent with ChIP-chip derived connectivity. However, the natures of transcription systems and μ-array data propose a challenge to NCM.

Unlike chemical spectra where the number of wavelengths is often greater than the number of chemicals ($M > L$), in transcription systems it is not uncommon to have fewer experiments (μ-arrays) than acting transcription factors ($M < L$). Exploratory techniques such as NCM, SVD, ICA, and EFA, cannot deduce more regulators than there are experiments (see Supplemental Information section 3.6). This is an issue when attempting to deduce transcription networks with NCM. For one, transcription networks change with environment. This means that experiments in a single analysis should be closely related to ensure the degree of transcription network variation is small. During our current analysis this translated into analyzing datasets with ≤ 10 experiments. Hence, the transcription networks we could infer would have ≤ 10 regulators. To mitigate this situation, both experimental and computational approaches can be used. Experimentally, a larger number of μ-arrays could be performed at smaller time intervals or slightly varying conditions to ensure minimal network variation. Due to noise present in μ-array data, data replicates could also be used. However, if experiments were being designed for use with exploratory bipartite techniques, data from separate conditions would be recommended over replicates. Ideally, the number of μ-arrays would exceed the number of factors thought active in a system. However, transcriptional responses can involve large scale expression changes effected by a large number of transcription factors, yielding experimental strategies extremely labor intensive. Under these circumstances computational strategies can be used to lower the number of necessary μ-arrays, both for future experiments and currently available data. One strategy that may be employed focuses on the isolation of sub-networks where $\leq M$ transcription factors are known to function (Yang and Liao, 2005). After independent analysis of the sub-networks, results can be recombined to get a global view of the transcription system. Indeed this strategy has worked previously, and has been adopted here (see Supplementary Information section 1.3).

Conceivably, after employing the strategy of Yang and Liao, 2005 NCM should be able to infer most of the transcription system from expression data. However, the level of noise present in μ-array data remained an issue. With excessive noise the network signatures embedded in data that are utilized by NCM become obscured. The Gibbs sampler was implemented to identify genes whose network connectivity was capable of generating their expression data despite the presence of noise and error.

The Gibbs sampler identified genes with accurate expression and binding data. It did not process the data to remove noise or error, yet simply identified those genes with less error and noise in their expression and binding data. Thus genes identified by the Gibbs sampler as accurate would be the best candidates to work with NCM. However, NCM did not deduce ChIP-chip derived

connectivity for all those genes identified. Deduced connectivity that did not match ChIP-chip connectivity often erred on the side of more regulators per gene. This is indicative of increased noise levels, since deduced networks will tend toward versatility by mistaking noise for signal at higher levels.

Despite these difficulties NCM successfully deduced transcription networks consistent with ChIP-chip connectivity solely from gene expression data. This shows the potential NCM has for defining transcription networks. In particular, when connectivity data is unavailable or is available in a different environment NCM could be used to identify connectivity if expression data is comparatively clean. Network Component Mapping requires only expression data and a user-specified edge significance threshold, and assumes nothing beyond a log-linear transcription model and linear independence of TFAs. In fact, even with noisy expression data NCM could be used on its own to infer the sparsest network at a given noise threshold, or it could be used in conjunction with partial network knowledge to infer the sparsest network consistent with prior information.

However, no discussion about deducing transcription networks from gene expression is complete without mention of Bayesian Networks (BN). Bayesian Networks are a popular technique to deduce regulatory interactions from expression data (Friedman *et al.*, 2000; Pe'er *et al.*, 2002; Segal *et al.*, 2003; Friedman, 2004). Using joint probability distributions within expression data acyclic regulatory maps are inferred. These regulatory maps are not confined to be bipartite as the analyses discussed here are, but take on a nested tier structure that dictates when the expression of one gene is dependent on the expression of another gene. The dependent gene is interpreted as being regulated by the gene whose expression its expression is dependent on. While this strategy has had success discerning regulatory interactions from expression data its assumption of regulator activity correlating with transcript level could be troublesome, especially when post-translational modifications define activity and combinatorial regulation is present. In NCM, regulator activities are never assumed correlated with single transcript levels, but are deduced from all transcript levels present. Indeed in the first chemical network not a single output spectrum was representative of the constituent spectra, yet NCM deduced the source signals easily. When BNs were used to analyze spectrum data from the first chemical network the resulting regulatory map was excessively complex (see Supplemental Information section 3.9 for details). This was most likely due to the absence of representative constituent spectra, and the high degree of similarity between the constituent sources. When the activities of multiple regulators are highly related BNs could encounter problems, since the joint distribution may find everything interdependent. Complex maps deduced from these situations are difficult to interpret and could lead to improper inferences. On the other hand, NCM deduced connectivity was explicit and easily interpretable.

Lastly, it is worth noting that the performance of NCM is expected to improve as technical advancements in the DNA μ-array technique become available, and further improvement to the algorithm progresses. In this work the performance of four different model selection techniques (AIC, SIC, RIC, CV) and our threshold trimming procedure were investigated. While all model selection techniques performed poorly with spectrum data, Leave-One-Out cross validation (LOO-CV) showed promise for the analysis of μ-array data (Supplemental Information section 2.3.2). However, LOO-CV is computationally intensive, especially as the number of data points increases. In consideration that a trimming step is needed more than one thousand times per iteration of NCM for the relatively small networks of the current data, incorporation of LOO-CV at this time is infeasible. Currently, the approach for selection of a trimming threshold requires classification of data into one of two categories, clean or noisy. For data from sources known to yield relatively clean data (eg. spectrophotometer) we suggest a strict trimming threshold (initial mapping: 0.01, fine mapping: 0.05), while for data from sources known to produce noisy data (eg. DNA μ-array) we suggest a more relaxed trimming threshold (initial mapping: 0.02, fine mapping: 0.2-0.25). However, as demonstrated in the Supplemental Information section 3.4, NCM deduces networks that are highly accurate for a large range of thresholds (fine mapping, spectrum: 0.01-0.25, expression: 0.10-0.35). This illustrates that NCM can produce highly accurate results without the use of an optimal trimming threshold. This is particularly attractive for situations when the organism is poorly characterized, or the response of an organism to a particular environment is poorly understood. In addition, the threshold can easily be varied to obtain a comprehensive view of the solution landscape. Ideally, a trimming procedure dependent on the data that is computationally feasible could be implemented in order to reduce the degree of user input. Also, even though NCM does not require additional information about the system to perform its analysis, prior information regarding the network topology can be incorporated.

ACKNOWLEDGEMENTS

This work has been supported by the Center for Cell Mimetic Space Exploration (CMISE) a NASA University Research, Engineering and Technology Institute (URETI) under award number #NCC 2-1364, NSF-ITR CCF-0326605, and the UCLA-DOE Institute for Genomics and Proteomics.

REFERENCES

Akaike, H. (1987) Factor Analysis and AIC. *Psychometrika*. **52**, 317-332.

Alter, O. *et al.,* (2000) Singular value decomposition for genome-wide expression data processing and modeling. *Proc Natl Acad Sci U S A.* **97**, 10101-6.

Alter, O. *et al.,* (2003) Generalized singular value decomposition for comparative analysis of genome-scale expression data sets of two different organisms. *Proc Natl Acad Sci U S A.* **100**, 3351-6.

Beal, M. J. *et al.,* (2005) A Bayesian approach to reconstructing genetic regulatory networks with hidden factors. *Bioinformatics.* **21**, 349-56.

Boulesteix, A. L. and K. Strimmer (2005) Predicting transcription factor activities from combined analysis of microarray and ChIP data: a partial least squares approach. *Theor. Biol. Med. Model.*, **2**, 23.

Browne, M. (2001) An Overview of Analytic Rotation in Exploratory Factor Analysis. *Multivariate Behavioral Research.* **36**, 111-150.

Brynildsen, M. P. *et al.,* (2006) A Gibbs sampler for the identification of gene expression and network connectivity consistency. *Bioinformatics.*

Brynildsen, M. P. *et al.,* (2006) Versatility and Connectivity Efficiency of Bipartite Transcription Networks. *Biophys. J.*, **91**, 2749-59.

Bussemaker, H. J. *et al.,* (2001) Regulatory element detection using correlation with expression. *Nat. Genet.*, **27**, 167-71.

Friedman, N. (2004) Inferring cellular networks using probabilistic graphical models. *Science.* **303**, 799-805.

Friedman, N. *et al.,* (2000) Using Bayesian networks to analyze expression data. *J Comput Biol.* **7**, 601-20.

Galbraith, S. J. *et al.,* (2006) Transcriptome network component analysis with limited microarray data. *Bioinformatics.*

Gao, F. *et al.,* (2004) Defining transcriptional networks through integrative modeling of mRNA expression and transcription factor binding data. *BMC Bioinformatics.* **5**, 31.

Harbison, C. T. *et al.,* (2004) Transcriptional regulatory code of a eukaryotic genome. *Nature.* **431**, 99-104.

Holter, N. S. *et al.,* (2000) Fundamental patterns underlying gene expression profiles: simplicity from complexity. *Proc Natl Acad Sci U S A.* **97**, 8409-14.

Lee, S. I. and S. Batzoglou (2003) Application of independent component analysis to microarrays. *Genome Biol.* **4**, R76.

Lee, T. I. *et al.,* (2002) Transcriptional regulatory networks in Saccharomyces cerevisiae. *Science.* **298**, 799-804.

Li, Z. *et al.,* (2006) Using a state-space model with hidden variables to infer transcription factor activities. *Bioinformatics.* **22**, 747-54.

Liao, J. C. *et al.,* (2003) Network component analysis: reconstruction of regulatory signals in biological systems. *Proc. Natl. Acad. Sci. U S A.* **100**, 15522-7.

Liebermeister, W. (2002) Linear modes of gene expression determined by independent component analysis. *Bioinformatics.* **18**, 51-60.

Papp, B. and S. Oliver (2005) Genome-wide analysis of the context-dependence of regulatory networks. *Genome Biol.* **6**, 206.

Pe'er, D. *et al.,* (2002) Minreg: inferring an active regulator set. *Bioinformatics.* **18 Suppl 1**, S258-67.

Qian, J. *et al.,* (2003) Prediction of regulatory networks: genome-wide identification of transcription factor targets from gene expression data. *Bioinformatics.* **19**, 1917-26.

Sabatti, C. and G. M. James (2006) Bayesian sparse hidden components analysis for transcription regulation networks. *Bioinformatics.* **22**, 739-46.

Sanguinetti, G. *et al.,* (2006) Probabilistic inference of transcription factor concentrations and gene-specific regulatory activities. *Bioinformatics.* **22**, 2775-81.

Segal, E. *et al.,* (2003) Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nat Genet.* **34**, 166-76.

Thurstone, L. (1947). The Simple Structure Concept. In *Multiple Factor Analysis: A Development and Expansion of The Vectors of Mind*. The University of Chicago Press, Chicago, 319-346.

Tran, L. M. *et al.,* (2005) gNCA: a framework for determining transcription factor activity based on transcriptome: identifiability and numerical implementation. *Metab. Eng.*, **7**, 128-41.

Wu, F. X. *et al.,* (2004) Modeling gene expression from microarray expression data with state-space equations. *Pac Symp Biocomput.* 581-92.

Yang, Y. L. and J. C. Liao (2005) Determination of functional interactions among signalling pathways in Escherichia coli K-12. *Metab. Eng.*, **7**, 280-90.

Yang, Y. L. *et al.,* (2005) Inferring yeast cell cycle regulators and interactions using transcription factor activities. *BMC Genomics.* **6**, 90.

Yeung, M. K. *et al.,* (2002) Reverse engineering gene networks using singular value decomposition and robust regression. *Proc Natl Acad Sci U S A.* **99**, 6163-8.