Transcription Network Analysis by A Sparse Binary Factor Analysis Algorithm

Shikui Tu[1], Runsheng Chen[2,†], Lei Xu[1,†,*]

[1]Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong, China

[2]Bioinformatics Laboratory and National Laboratory of Biomacromolecules, Institute of Biophysics, Chinese Academy of Sciences, Beijing 100101, China

Summary

Transcription factor activities (TFAs), rather than expression levels, control gene expression and provide valuable information for investigating TF-gene regulations. The underlying bimodal or switch-like patterns of TFAs may play important roles in gene regulation. Network Component Analysis (NCA) is a popular method to deduce TFAs and TF-gene control strengths from microarray data. However, it does not directly examine the bimodality of TFAs and it needs the TF-gene connection topology to be *a priori* known. In this paper, we modify NCA to model gene expression regulation by Binary Factor Analysis (BFA), which directly captures switch-like patterns of TFAs. Moreover, sparse technique is employed on the mixing matrix of BFA, and thus the proposed sparse BYY-BFA algorithm, developed under Bayesian Ying-Yang (BYY) learning framework, can not only uncover the latent TFA profile's switch-like patterns, but also be capable of automatically shutting off the unnecessary connections. Simulation study demonstrates the effectiveness of BYY-BFA, and a preliminary application to *Saccharomyces cerevisiae* cell cycle data and *Escherichia coli* carbon source transition data shows that the reconstructed binary patterns of TFAs by BYY-BFA are consistent with the ups and downs of TFAs by NCA, and that BYY-BFA also works well when the network topology is unknown.

1 Introduction

High-dimensional data from DNA microarray are typically controlled by low-dimensional regulatory signals through an interacting network [1]. The gene expression is controlled by one transcription factor (TF) alone or several TFs in combination on the target promoter regions. Transcription factor activities (TFA) rather than levels of transcription factor expression play roles in transcriptional regulations. It is a challenging problem in system biology to reconstruct the dynamics of the hidden regulatory signals of TFs from the transcript levels of the genes they control [1].

Network component analysis (NCA) [1, 2] is a popular method to deduce TFA and TF-gene regulation control strengths from transcriptome data and a priori network structure information which is usually constructed from ChIP-chip binding assays. In NCA, the (relative) gene expression is formulated as the product of each (relative) TFA to the power of the control strength

† These authors contributed equally to this work

*To whom correspondence should be addressed. Email: lxu@cse.cuhk.edu.hk

from that TF to the gene, or equivalently, the relationship between TFA and and gene expression is approximated as a log-linear model, where the log expression level of a gene is modeled by a mixture of log TFAs weighted by the control strengths. NCA takes the available network topology into account to compute the component TFAs, different from principal component analysis (PCA) or independent component analysis (ICA) which are based on statistical properties that may hinder the biological interpretations [1]. Three identifiability criteria need to be satisfied in NCA for a unique decomposition [1, 3]. NCA has been successfully applied to determine the transcription regulatory activities, on microarray data generated from yeast *Saccharamyces cerevisiae* during cell-cycle process or *Escherichia coli* carbon source transition from glucose to acetate [1, 4], and so on. Other related methods include the REDUCE [5] that takes normalized motif binding copy number as the regulation strength of the TF for that gene, and obtains TFA profiles from gene expression data by linear regression.

Based on the NCA framework, this paper modifies NCA to model gene transcriptional regulation by Binary Factor Analysis (BFA), which assumes the latent TFAs take binary values, e.g., indicating whether or not a TF is activated. The identification of bimodal activity is useful to identify the biological variation of TFs whose regulatory dynamics are tightly around two discrete levels which are usually corrupted by noise. The switching pattern can be shown in many ways, including high expression versus low expression, activation versus deactivation, on versus off, and so on. To examine the functional of the protein or RNA products through gene expression level, a simple and natural way is to divide the gene expression levels into two categories as by Barcode in [6], which converts relative measure of expression from a single microarray into a reliable absolute measure of expressed/unexpressed calls for each gene, aiming to find which genes are expressed in a given cell type. Switch-like genes are shown to exhibit tissue and disease-specific expression signatures [7]. The extracted binary pattern can be the results of stochastic switches in a gene regulatory network, or epigenetic differences in the form of methylation of cis-regulatory regions of genes contributing to regulation of gene expression [8], or thresholds in target gene expression generated by miRNAs which can act both as a switch and as a fine-tuner of gene expression [9]. Thus, examining the switch-like patterns of TFA variation may shed light upon the underlying regulatory mechanisms.

Another problem in transcription network analysis is to determine the network topology and control strengths. NCA needs a priori knowledge about the connection topology of the TF-gene regulatory network. However, in most organisms, the connectivity information is currently unavailable. Due to the noise in experiments, the known connectivity data may not be reliable. A two-stage method that integrates NCA with stepwise regression was proposed to trim the network with the help of a modified Akaike information criterion [10]. To avoid such a repeated implementation, we impose a sparsity penalty on the mixing matrix of control strengths, so that the extra entries are *automatically* pushed to zeros if there is not enough evidence for the existence of connections. Efforts on sparsity have been made on sparse learning or Lasso shrinkage by L1 norm penalty or a Laplacian prior, and so on [11, 12]. Within the Bayesian paradigm, we consider on each entry of the mixing matrix a joint Normal-Jeffreys prior distribution which is shown to implement sparsity well without any hyper-parameters to be determined [13]. We propose to implement BFA under Bayesian Ying-Yang (BYY) [14, 15] learning framework. The derived BYY-BFA algorithm can utilize both the available connectivity information and sparsity property on the mixing matrix.

To test our proposed algorithm, experiments are conducted on both synthetic data and real data. Simulation study demonstrates the effectiveness of sparse BYY-BFA in recovering the hidden

dynamics of TF regulatory signals, and in estimating the connectivity topology and control strengths. When applied on the yeast cell cycle data [16], the reconstructed binary TFAs by BYY-BFA are consistent with the ups and downs of the continuous ones by NCA. If sparsity is activated in BYY-BFA, many connections are shut off as the their regulation strengths are pushed towards zeros, but without changing the cyclic patterns of the hidden TFAs. This suggest that some connections are not necessary or may be false positive. Moreover, the problem becomes more difficult if no network topology is used. However, the connectivity and the regulatory dynamics can still be inferred to some extent. Most connections are shut off, and approximately 40% the obtained connections are consistent with experimental ChIP-chip data. Therefore, our algorithm directly captures the underlying activation patterns of TFAs, and also extends NCA to the case when the topology of the TF-gene network is not available or not reliable. In addition, results on E. coli data show that BYY-BFA is effective to detect activations of TFAs corresponding to the adaptation to carbon source transition from glucose to acetate [4].

The rest of this paper is organized as follows. In Section 2, we provide a brief review on NCA, based on which we proceed to introduce BFA for gene regulation modeling, with a BYY-BFA algorithm developed. Section 3 is devoted to experiments on synthetic data and real data, to validate the performance of BYY-BFA. Finally, conclusion is made in Section 4. The algorithm details are left in Appendix.

2 Methods

2.1 A brief review on NCA

NCA approximates gene expression as the product of the contribution of each TF regulatory activity using the following model [1]:

$$\frac{E_i(t)}{E_i(0)} = \prod_{j=1}^{m} \left(\frac{TFA_j(t)}{TFA_j(0)} \right)^{CS_{ij}}, \tag{1}$$

or equivalently a log-linear model in canonical matrix form:

$$X = AY + \Gamma, \tag{2}$$

where $E_i(t)$ is the gene expression level, $TFA_j(t)$ is the activity of the TF j, and CS_{ij} represents the control strength of TF j on gene i, and $X = [\log(E_i(t)/E_i(0))]_{n\times N}$, $Y = [\log(TFA_j(t)/TFA_j(0))]_{m\times N}$, $A = [CS_{ij}]_{n\times m}$, and Γ is the residual.

NCA is to minimize the residual Γ to get a decomposition $X \approx \hat{A}\hat{Y}$, from the observed gene expression profile X and the known connectivity for A, i.e., C_{ij} is fixed at zero if TF j does not bind to the promoter region of gene i, otherwise the control strength is non-zero. To guarantee a unique decomposition up to some normalization factors, NCA requires A and the resultant connectivity matrix by removing a regulatory node together with its connected neighbor genes to have full-column rank, and requires Y to have full-row rank. It should be noted that the NCA solution does not assume any relationship between the TFAs.

2.2 Binary Factor Analysis

Binary Factor Analysis (BFA) explores latent binary structures of data. Unlike the conventional factor analysis where the latent factor is Gaussian, BFA assumes the observations are generated by Bernoulli distributed binary factors. From an information theoretic perspective, the latent independent random bits take the role of causal sources. Research on BFA has been conducted with wide applications, on analysis of binary data (e.g., social research questionnaires, market basket data, etc.) with the aid of Boolean algebra [17], or to discover binary factors in continuous data, e.g., see page 839-840 of [18] and also see [19, 20, 21].

BFA model assumes that an n-dimensional observation $\boldsymbol{x}$ is formed by n mixtures of m independent binary factors $y_1, \ldots, y_m$ and added with a Gaussian noise $\mathbf{e}$, i.e.,

$$\boldsymbol{x} = A\boldsymbol{y} + \mathbf{a}_0 + \mathbf{e}, \tag{3}$$

where A is an $n \times m$ mixing matrix, and $\mathbf{a}_0$ is a bias vector. Moreover, the distributions of the variables are given by

$$q(\boldsymbol{y}) = \prod_{j=1}^{m} q(y_j), \quad q(\boldsymbol{x}|\boldsymbol{y}) = G(\boldsymbol{x}|A\boldsymbol{y} + \mathbf{a}_0, \Sigma_e), \tag{4}$$

where Σ_e is a diagonal covariance matrix, and $G(z|\mu, \Sigma)$ denotes a Gaussian probability density with mean μ and covariance Σ. One natural way to model a binary variable y_j in Eq.(4) is Bernoulli distribution [18, 19, 21]. Equivalently, we consider that $\boldsymbol{y}$ takes binary values from a two-component Gaussian mixture, that is,

$$q(y_j) = \sum_{\ell=1}^{2} \alpha_{j\ell} G(y_j|\mu_{j\ell}, \sigma_{j\ell}^2), \tag{5}$$

where $0 \leq \alpha_{j\ell} \leq 1$, and $\sum_{\ell=1}^{2} \alpha_{j\ell} = 1$, $\mu_{j1} = 0, \mu_{j2} = 1$, and $\sigma_{j\ell}$ is fixed at a very small value.

According to Eq.(2)&(3), we use BFA to modify the framework of network component analysis (NCA) [1] by considering the mixing matrix A as the connectivity matrix between the transcription factors (TF) and the genes, the latent factor $\boldsymbol{y}$ to encode the state of the TF activity, and the observation $\boldsymbol{x}$ to be the gene expression.

2.3 Sparsity on the mixing matrix

As indicated by the literature [1], the number of target genes of a TF is usually small, and thus the connectivity matrix A is sparse, i.e., with many zero entries. If connectivity is partially known from biological experiments or computational predictions, then the case falls in the general framework of semi-blind factor analysis, e.g., see Sec.4.3 in [22]. However, the connectivity information may be not available or not reliable. One possible way is to deactivate those false positive connections by automatically shrinking control strengths to zeros during learning with the help of a sparseness constraint. Efforts on sparsity have been made on sparse learning or Lasso shrinkage by L1 norm penalty or a Laplacian prior, and so on [11, 12]. In

this paper, we adopt the following joint Normal-Jeffreys probability density on each entry of the mixing matrix [13]:

$$q(A|\boldsymbol{\gamma})q(\boldsymbol{\gamma}) \propto \prod_{i,j} G(a_{ij}|0,\gamma_{ij}) \cdot \prod_{i,j} \frac{1}{\gamma_{ij}}, \tag{6}$$

where γ_{ij} is the variance of a_{ij} and controlled by a Jeffreys prior without hyper-parameters.

2.4 Implementing sparse BFA under BYY framework

First proposed in [14] and systematically developed over a decade and a half [23], Bayesian Ying-Yang (BYY) harmony learning is a general statistical learning framework for parameter learning and model selection under a best harmony principle, which aims to maximize the following harmony measure:

$$H(p\|q) = \int \sum_{L} p(X)p(\boldsymbol{\Theta}|X)p(Y,L|X,\boldsymbol{\Theta}) \ln[\\ q(X|Y,L,\boldsymbol{\Theta})q(Y|L,\boldsymbol{\Theta})q(L|\boldsymbol{\Theta})q(\boldsymbol{\Theta})]dYdXd\boldsymbol{\Theta}, \tag{7}$$

where $q(\cdot)$ gives the Ying representation, and $p(\cdot)$ gives the Yang representation. For the BFA model introduced in Sec. 2.2, components in Ying representation follow from the above specifications in Eq.(4)-(6) with $X = \{\boldsymbol{x}\}$, $Y = \{\boldsymbol{y}\}$, $L = \{j\}$ and $\boldsymbol{\Theta} = \{A, \mathbf{a}_0, \Sigma_e, \alpha_{j\ell}, \mu_{j\ell}, \sigma^2_{j\ell}\}$. In Yang representation, the empirical density $p(X) = \delta(X - X_N)$ is adopted with $X_N = \{\boldsymbol{x}_t\}_{t=1}^N$, and the other components are free, i.e., no constraints on forms of their probability densities.

The derived algorithm to maximize $H(p\|q)$ is called BYY-BFA, and its details are referred to the Tab. 4 in Appendix. Sparse learning on the mixing matrix A is activated by $q(\boldsymbol{\Theta}) = q(A|\boldsymbol{\gamma})q(\gamma)$, or is shut off by $q(\boldsymbol{\Theta}) = 1$. Moreover, the given connectivity data can be utilized by fixing the corresponding entries of A at zero through $\widetilde{A}$ if there is no known connection in Tab. 4, or by a confidence probability for a flexible incorporation [22]. The obtained combinations of implementations of BYY-BFA are summarized in Tab. 1.

The last one BYY-BFA(n+f) is actually a special case of the BYY harmony learning algorithm for non-Gaussian factor analysis (NFA) proposed in Sect. 5 of [24] and Sect IV(C) in [15], at the special case of a two-component Gaussian mixture by Eq.(5), or the BYY BFA learning given on p840 of [15] and also the one studied in [19, 21]. The other three BYY-BFA algorithms extends these previous studies by considering either or both of a priori connectivity and a priori $q(\boldsymbol{\Theta})$.

The above is a brief introduction to BYY. Readers are referred to not only a summary of nine aspects on the novelty and favorable natures of BYY best harmony learning, made at the end of Sect. 4.1 in [23], but also the roadmap shown in Fig. A2 of in [23], as well as to a systematic outline on the 13 topics about best harmony learning in Sect. 7 of [25].

Table 1: Implementation of the algorithms. "c": constraining A with the connectivity topology from e.g., ChIP-chip assay; "f": using a full A (no a priori connectivity); "n": implementing BYY-BFA without sparse prior; "s": implementing BYY-BFA with sparse prior.

algorithm	details
NCA	implemented by NCA toolbox [1]
BYY-BFA(n+c)	constraining A as NCA, $\tau = 0$ in Tab. 4
BYY-BFA(s+c)	constraining A as NCA, $\tau = 1$ in Tab. 4
BYY-BFA(s+f)	$\widetilde{A} = \mathbf{1}_{n\times m}$ (a matrix of all ones), $\tau = 1$ in Tab. 4
BYY-BFA(n+f)	$\widetilde{A} = \mathbf{1}_{n\times m}$, $\tau = 0$ in Tab. 4

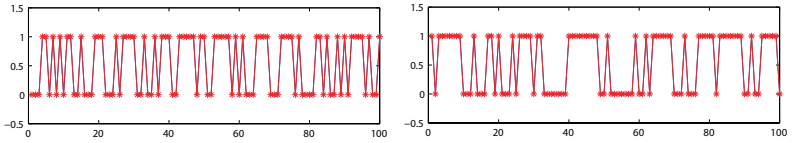


Figure 1: The true binary activities (in blue, covered by the red points) is correctly reconstructed by the ones (in red) by BYY-BFA

3 Results

3.1 On simulated data

We demonstrate the effectiveness of the proposed algorithm by simulated data sets. We set $n = 6, m = 2, k_1 = k_2 = 2, \Sigma_e = 0.1\mathbf{I}_n, \mathbf{a}_0 = \mathbf{0}$. The mixing matrix A is randomly generated and then randomly set $p_s = 40\%$ percentage of the entries to be zero. The final mixing matrix is given by A_{ob} in Eq.(8). The factor distributions are two-component Gaussian mixture with means $\mu_{j1} = 0, \mu_{j2} = 1$ and variances fixed at $\sigma_{j1}^2 = \sigma_{j2}^2 = 10^{-5}$, and k_1 and k_2 are fixed at 2. Then, a synthetic data set X_N of sample size $N = 100$ is randomly generated according to the BFA model given in Eq.(3)-(4).

The sparse BYY-BFA algorithm (i.e., BYY-BFA(s+f)) is implemented on X_N by randomly initializing $\mathbf{\Theta} = \{A, \Sigma_e, \mathbf{a}_0, \alpha_{rj_r}\}$ with $m = 2$. It can recover the true connectivity matrix A_{ob} in Eq.(8) by a sparse matrix $\hat{A}_{sb}$, with the corresponding elements pushed to zeros when there is no enough evidence to support such connections. If not imposing the prior by Eq.(6), the obtained A_{fb} in Eq.(8) is not sparse enough. Figure 1 shows that the underlying 0-1 switch-like factor activities are correctly recovered.

$$A_{ob} = \begin{pmatrix} -0.9785 & 2.1000 \\ 0.4908 & 0.1858 \\ 0 & -0.4478 \\ 0 & 0 \\ 0.6946 & 1.4923 \\ 0 & 1.1556 \end{pmatrix}, \hat{A}_{fb} = \begin{pmatrix} -1.0924 & 1.9265 \\ 0.5292 & 0.1218 \\ 0.0172 & -0.4930 \\ 0.0560 & 0.0234 \\ 0.6886 & 1.5070 \\ -0.0580 & 1.0953 \end{pmatrix}, \hat{A}_{sb} = \begin{pmatrix} -1.0799 & 1.9233 \\ 0.4862 & 0.0223 \\ 0.0004 & -0.4662 \\ 0.0012 & 0.0003 \\ 0.6669 & 1.4951 \\ -0.0013 & 1.0903 \end{pmatrix} \quad (8)$$

Table 2: Reconstruction mean square errors (MSE) of NCA and BYY-BFA on yeast cell-cycle data. We implement BYY-BFA(n) and BYY-BFA(s) in a combination of whether (c) constraining A with the connectivity topology from ChIP-chip assay or (f) using a free A.

algorithm	MSE(n+c)	MSE(s+c)	MSE(s+f)
NCA	0.1320	-	-
BYY-BFA	0.1677	0.1727	0.1495

3.2 On real data

3.2.1 Yeast cell-cycle data

We apply our algorithm to microarray data sets that are about yeast cell-cycle regulation. The data were taken from wild-type *S. cerevisiae* cultures synchronized by three independent methods, α-factor arrest, elutriation, and arrest of a *cdc15* and temperature-sensitive mutant [16], as well as *cdc28* data [26]. The connectivity information between transcription factors and their regulated genes comes from the genomewide location or ChIP-chip assay [27].

We focus on 6 transcription factors (TF) that are known to be related to cell-cycle regulation [27, 1]. Based on the "NCA Toolbox" [1], 137 genes regulated by these TFs were selected, with the connectivity information obtained from [27].

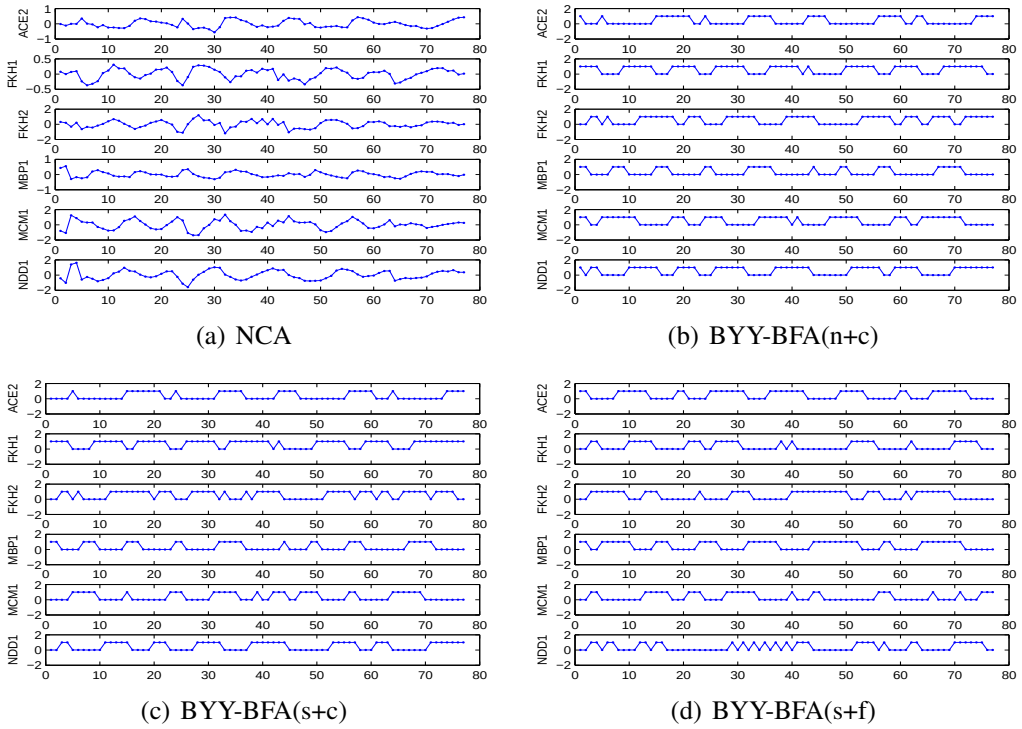
First, we implement the BYY-BFA(n+c) in Tab. 1. Similar to NCA, we construct $\widetilde{A} = [\widetilde{a}_{ij}]_{n\times m}$ in Tab. 4 as: $\widetilde{a}_{ij} = 1$ if the j-th TF regulates the i-th gene according to the connectivity data; otherwise, $\widetilde{a}_{ij} = 0$. The reconstruction mean square errors (MSE), i.e.,

$$MSE = \frac{1}{nN}\sum_{t=1}^{N} \|\boldsymbol{x}_t - \hat{A}\hat{\boldsymbol{y}}_t - \hat{\mathbf{a}}_0\|^2 \tag{9}$$

are given in Tab. 3, where $\boldsymbol{x}_t$ is the gene expression vector at time t, and $\hat{A}, \hat{\boldsymbol{y}}_t, \hat{\mathbf{a}}_0$ are connectivity matrix, TFAs and mean vector, estimated by NCA or BYY-BFA(n+c). The two algorithms both reconstruct the microarray expression data with small errors. BYY-BFA(n+c) obtained a relative larger MSE because it constrains TFAs to be binary. The estimated regulatory activities are presented in Fig. 2(a)&2(b). Similar to NCA results, the dynamics of the TF activities estimated by the BYY-BFA(n+c) also show cyclic behavior, which reveals the role of each TF during cell cycle regulation. The reconstructed TFAs by BYY-BFA(n+c) are highly consistent with those by NCA, i.e., 0 and 1 correspond to low and high, respectively.

Second, we implement BYY-BFA(s+c) given in Tab. 1. The estimated TFA profile is given in Fig. 2(c), which resembles Fig. 2(a). According to Tab. 3, 128 of 203 connections from the ChIP-chip assay were shut off by the sparse learning, which indicates the 128 connections might be false positive or a mechanism of redundancy to ensure the robustness [28].

Third, we implement BYY-BFA(s+f) given in Tab. 1. Table 3 shows that a large part of the relaxed connections are switched off, and about 40% of the remaining connections are consistent with the ChIP-chip experiment. As in Fig. 2(d), most of TFAs still preserve the cyclic pattern.



(a) NCA (b) BYY-BFA(n+c)

(c) BYY-BFA(s+c) (d) BYY-BFA(s+f)

Figure 2: The estimated TFA profiles on Yeast cell-cycle data by the algorithms in Tab. 1.

Table 3: Confusion matrices of the reconstructed connectivity by sparse BYY-BFA on yeast cell-cycle data against the known connectivity from available experiments. For sparse BYY-BFA, a connection is determined to be present if the absolute value of its estimated control strength is larger than a threshold 0.02. Notations: $\hat{0}$ or $\hat{1}$ denotes reconstruction, 0^* or 1^* denotes the known connectivity.

BYY-BFA(s+c)

	0^*	1^*
$\hat{0}$	0	128
$\hat{1}$	0	75

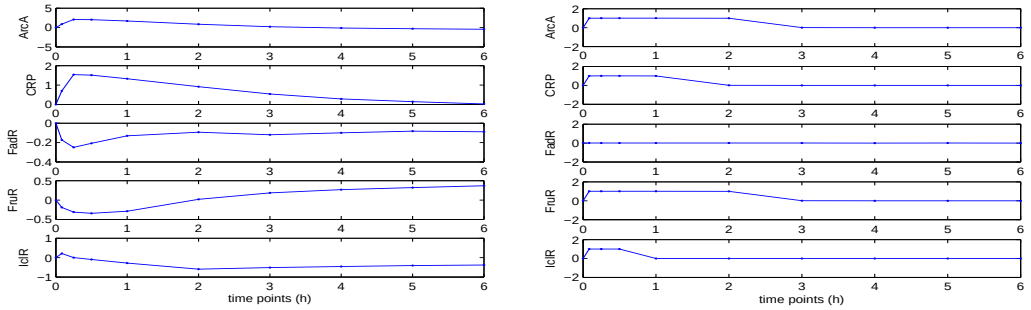
BYY-BFA(s+f)

	0^*	1^*
$\hat{0}$	535	152
$\hat{1}$	84	51

3.2.2 *E. coli* carbon source transition data

We further apply BYY-BFA to temporal gene expression profiles of *E. coli* during transition from glucose to acetate, with samples taken at $5, 15, 30, 60$ min and every hour until 6 h after transition [4]. Similar to [4], the repeated data points are averaged, and we demonstrate the effectiveness of BYY-BFA on 5 of 16 TFs that were analyzed by NCA in [4]. Based on the available connectivity data and the applicability for NCA, 22 genes were selected by the NCA toolbox [1], and there were 30 regulations from TFs to genes.

Similar to the reconstructed TFA profiles by NCA in Fig. 3(a), most TFAs by BYY-BFA(s+c) in Fig. 3(b) show activation immediate after transition and then gradually become stable, corresponding to the adaptation of cells to the new environmental condition. However, different from NCA, BYY-BFA(s+c) computed the TF "FadR" to be not activated after transition. Moreover, 16 of 30 regulation strengths are pushed to zero, or the corresponding regulations are shut off. Similar to Fig. 2, the results by other implementations of BYY-BFA can also be computed.



(a) NCA (b) BYY-BFA(s+c)

Figure 3: The estimated TFA profiles on E.Coli data: (a) by NCA with MSE 0.0117**; (b) by BYY-BFA(s+c) with MSE** 0.02213**.**

4 Conclusion

Based on the NCA framework, we modify NCA to model the gene transcriptional regulation by BFA, which assumes the latent regulatory signals to have two states, e.g., activated or deactivated. Formulated within Bayesian paradigm, BFA can properly incorporate the *a priori* knowledge from experiments and the sparseness feature on the connectivity matrix via a Normal-Jeffreys prior. Synthetic experiments have demonstrated the effectiveness of our derived sparse BYY-BFA algorithm in uncovering the latent TFA profile, and estimating the control strengths. The extra connections are shut off by pushing the corresponding control strengths to zeros due to the sparseness feature. Moreover, a preliminary application to *Saccharomyces cerevisiae* cell cycle data and *Escherichia coli* carbon source transition data shows that BYY-BFA not only reconstructs the hidden TF regulatory signals to be consistent with the up-and-down patterns of the results by NCA, but also is capable of shutting off unnecessary or unreliable connections.

Acknowledgements

The work described in this paper was fully supported by National Key Basic Research & Development Program 973 under Grant No. 2009CB825404.

References

[1] J. C. Liao, R. Boscolo, Y.-L. Yang, L. M. Tran, C. Sabatti and V. P. Roychowdhury. Network component analysis: Reconstruction of regulatory signals in biological systems. *Proceedings of the National Academy of Sciences*, 100(26):15522–15527, 2003.

[2] L. M. Tran, M. P. Brynildsen, K. C. Kao, J. K. Suen and J. C. Liao. gNCA: A framework for determining transcription factor activity based on transcriptome: identifiability and numerical implementation. *Metabolic Engineering*, 7(2):128 – 141, 2005.

[3] S. J. Galbraith, L. M. Tran and J. C. Liao. Transcriptome network component analysis with limited microarray data. *Bioinformatics*, 22(15):1886–1894, 2006.

[4] K. C. Kao, Y.-L. Yang, R. Boscolo, C. Sabatti, V. Roychowdhury and J. C. Liao. Transcriptome-based determination of multiple transcription regulator activities in Escherichia coli by using network component analysis. *Proceedings of the National Academy of Sciences of the United States of America*, 101(2):641–646, 2004.

[5] H. J. Bussemaker, H. Li and E. D. Siggia. Regulatory element detection using correlation with expression. *Nature Genetics*, 27:167–174, 2001.

[6] M. N. McCall, K. Uppal, H. A. Jaffee, M. J. Zilliox and R. A. Irizarry. The Gene Expression Barcode: leveraging public data repositories to begin cataloging the human and murine transcriptomes. *Nucleic Acids Research*, 39:D1011–D1015, 2011.

[7] M. Gormley and A. Tozeren. Expression profiles of switch-like genes accurately classify tissue and infectious disease phenotypes in model-based classification. *BMC Bioinformatics*, 9(1):486, 2008.

[8] M. R. Bennett and J. Hasty. A DNA methylationcbased switch generates bistable gene expression. *Nature Genetics*, 39(2):146 – 147, 2007.

[9] S. Mukherji, M. S. Ebert, G. X. Zheng, J. S. Tsang, P. A. Sharp and A. van Oudenaarden. MicroRNAs can generate thresholds in target gene expression. *Nature Genetics*, 43(9):854–859, 2011.

[10] L. Tran, D. Hyduke and J. Liao. Trimming of mammalian transcriptional networks using network component analysis. *BMC Bioinformatics*, 11(1):511–521, 2010.

[11] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288, 1994.

[12] P. M. Williams. Bayesian regularisation and pruning using a laplace prior. *Neural Computation*, 7:117–143, 1994.

[13] Y. Guan and J. G. Dy. Sparse probabilistic principal component analysis. In *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics (AISTATS'09)*, volume 5, pages 185–192. 2009.

[14] L. Xu. Bayesian-Kullback coupled YING-YANG machines: unified learning and new results on vector quantization. In *Proceedings of International Conference on Neural Information Processing, Oct 30-Nov.3, Beijing, China*, pages 977–988. 1995. (A further version in NIPS8, D.S. Touretzky, et al (Eds.), MIT press, 444-450).

[15] L. Xu. Advances on BYY harmony learning: information theoretic perspective, generalized projection geometry, and independent factor autodetermination. *IEEE Transactions on Neural Networks*, 15(4):885–902, 2004.

[16] P. T. Spellman, G. Sherlock, M. Q. Zhang, V. R. Iyer, K. Anders, M. B. Eisen, P. O. Brown, D. Botstein and B. Futcher. Comprehensive identification of cell cyclecregulated genes of the yeast saccharomyces cerevisiae by microarray hybridization. *Molecular Biology of the Cell*, 9(12):3273–3297, 1998.

[17] A. Keprt and V. Snásel. Binary factor analysis with help of formal concepts. In V. Snásel and R. Belohlávek (editors), *CLA*, volume 110 of *CEUR Workshop Proceedings*, pages 90–101. CEUR-WS.org, 2004.

[18] L. Xu. BYY harmony learning, independent state space, and generalized APT financial analyses. *IEEE Transactions on Neural Networks*, 12(4):822–849, 2001.

[19] Y. An, X. Hu and L. Xu. A comparative investigation on model selection in binary factor analysis. *Journal of Mathematical Modelling and Algorithms*, 5(4):447–473, 2006.

[20] G. W. Taylor, G. E. Hinton and S. T. Roweis. Modeling human motion using binary latent variables. In B. Schölkopf, J. Platt and T. Hoffman (editors), *NIPS*, pages 1345–1352. MIT Press, Cambridge, MA, 2007.

[21] K. Sun, S. Tu, D. Y. Gao and L. Xu. Canonical dual approach to binary factor analysis. In T. Adali, C. Jutten, J. M. T. Romano and A. K. Barros (editors), *ICA*, volume 5441 of *Lecture Notes in Computer Science*, pages 346–353. Springer, 2009.

[22] L. Xu. Codimensional matrix pairing perspective of BYY harmony learning: hierarchy of bilinear systems, joint decomposition of data-covariance, and applications of network biology. *Frontiers of Electrical and Electronic Engineering in China*, 6:86–119, 2011.

[23] L. Xu. Bayesian Ying-Yang System, Best Harmony Learning, and Five Action Circling. *A special issue on Emerging Themes on Information Theory and Bayesian Approach, Frontiers of Electrical and Electronic Engineering in China*, 5(3):281–328, 2010.

[24] L. Xu. Independent component analysis and extensions with noise and time: A Bayesian Ying-Yang learning perspective. In *Neural Information Processing Letters and Reviews*, pages 1–52. 2003.

[25] L. Xu. On essential topics of BYY harmony learning: Current status, challenging issues, and gene analysis applications. *Frontiers of Electrical and Electronic Engineering in China (A special issue on Machine learning and intelligence science: IScIDE (C))*, 7:147–196, 2012.

[26] R. J. Cho, M. J. Campbell, E. A. Winzeler et al. A genome-wide transcriptional analysis of the mitotic cell cycle. *Molecular Cell*, 2:65–73, 1998.

[27] T. I. Lee, N. J. Rinaldi, F. Robert et al. Transcriptional regulatory networks in saccharomyces cerevisiae. *Science*, 298(5594):799–804, 2002.

[28] L. MacNeil and A. J. Walhout. Gene regulatory networks and the role of robustness and stochasticity in the control of gene expression. *Genome Research*, 21:645–657, 2011.

Appendix: Details of the sparse BYY-BFA algorithm

In Yang representation, the empirical density $p(X) = \delta(X - X_N)$ is adopted with $X_N = \{\boldsymbol{x}_t\}$, and all the other components are free, i.e., no constraints on their probability functions. In such

a setting, maximizing $H(p\|q)$ leads the unknown Yang components to be Dirac delta functions, $p(\boldsymbol{\Theta}|X) = \delta(\boldsymbol{\Theta} - \boldsymbol{\Theta}^*)$ and $p(\boldsymbol{y}_t, \mathbf{j}|\boldsymbol{x}_t) = \delta([\boldsymbol{y}, \mathbf{j}] - [\boldsymbol{y}_t^*, \mathbf{j}_t^*])$. It follows that Eq.(7) becomes

$$H(p\|q) \approx \sum_{t=1}^{N} \ln[G(\boldsymbol{x}_t|A\boldsymbol{y}_t + \mathbf{a}_0, \Sigma_e)\boldsymbol{\alpha}_{\mathbf{j}^*} G(\boldsymbol{y}_t^*|\boldsymbol{\mu}_{\mathbf{j}^*}, \Lambda_{\mathbf{j}^*})] + \tau \ln q(A|\boldsymbol{\gamma})q(\boldsymbol{\gamma}) \tag{10}$$

$$[\boldsymbol{y}_t^*, \mathbf{j}_t^*] = \arg\max_{\boldsymbol{y}_t, \mathbf{j}_t} \{\ln[G(\boldsymbol{x}_t|A\boldsymbol{y}_t + \mathbf{a}_0, \Sigma_e)\boldsymbol{\alpha}_{\mathbf{j}} G(\boldsymbol{y}_t|\boldsymbol{\mu}_{\mathbf{j}}, \Lambda_{\mathbf{j}})]\} \tag{11}$$

where $\mathbf{j} = [j_1, \ldots, j_m]$ with $q(y_r|j_r) = G(y_r|\mu_{rj_r}, \sigma_{rj_r}^2)$, and $q(\boldsymbol{y}_t|\mathbf{j})$ is a multivariate Gaussian density with mean $\boldsymbol{\mu}_{\mathbf{j}} = [\mu_{1,j_1}, \ldots, \mu_{m,j_m}]$ and covariance matrix $\Lambda = diag[\sigma_{1,j_1}^2, \ldots, \sigma_{m,j_m}^2]$, and $diag[\mathbf{u}]$ denotes a diagonal matrix with the vector $\mathbf{u}$ as its diagonal, and $q(\mathbf{j}) = \boldsymbol{\alpha}_{\mathbf{j}} = \prod_{r=1}^{m} \alpha_{rj_r}$, and τ is an indicator of whether the last term in Eq.(10) takes effects. Specifically, $\tau = 0$ shuts off the effects from the last term.

A Ying-Yang alternative procedure is implemented between "Step Y" and "Step $\boldsymbol{\theta}$ with details given in Tab. 4. The obtained algorithm is called BYY-BFA. It should be noted that $\tau = 0$ removes the sparsity constraint in BYY-BFA, which degenerates back to a special case of the BYY-NFA learning algorithm proposed in Sect. 5 of [24] and Sect. IV(C) in [15], simplified with only two-component Gaussian mixtures. The binary matrix $\widetilde{A}$ is used to incorporate known connectivity data, and it is set to be a matrix of all ones when no connectivity is available. Moreover, $\mu_{r,2}$ can also be free to be updated accordingly, for a more general BFA.

Table 4: The BYY-BFA algorithm

Objective: maximize $H(p\|q)$ by Eq.(10)
Step Y: get $\boldsymbol{y}_t^*$ and $\boldsymbol{j}_t^* = [j_{1t}^*, \ldots, j_{mt}^*]$, for $t = 1, \ldots, N$. $\boldsymbol{y}_t^*(\boldsymbol{x}_t|\mathbf{j}_t) = (\Lambda_{\mathbf{j}_t}^{-1} + A^T\Sigma_e^{-1}A)^{-1}[A^T\Sigma_e^{-1}(\boldsymbol{x}_t - \boldsymbol{a}_0) + \Lambda_{\mathbf{j}_t}^{-1}\boldsymbol{\mu}_{\mathbf{j}_t}]$, $\mathbf{j}_t^* = \arg\max_{\mathbf{j}\in\mathcal{M}} \ln[G(\boldsymbol{x}_t|A\boldsymbol{y}_t^* + \mathbf{a}_0, \Sigma_e)\alpha_{\mathbf{j}} G(\boldsymbol{y}_t^*|\boldsymbol{\mu}_{\mathbf{j}}, \Lambda_{\mathbf{j}})]$, where $\mathbf{j} = [j_1, \ldots, j_m]$, $\boldsymbol{\alpha}_{\mathbf{j}} = \prod_{r=1}^{m} \alpha_{r,j_r}$, $\boldsymbol{\mu}_{\mathbf{j}} = [\mu_{1,j_1}, \ldots, \mu_{m,j_m}]$, and $\Lambda_{\mathbf{j}} = diag[\sigma_{1,j_1}^2, \ldots, \sigma_{m,j_m}^2]$, $\mathcal{M} = \{\mathbf{j} \,|\, j_r = 1, \ldots, k_r; 1 \le r \le m\}$.
Step $\boldsymbol{\theta}$: by gradient method $\boldsymbol{\theta}^{new} \leftarrow \boldsymbol{\theta}^{old} + \eta \cdot \partial\boldsymbol{\theta}$, $\partial\boldsymbol{\theta} = \frac{\partial H(p\|q)}{\partial\boldsymbol{\theta}}\Big\|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{old}}$, $\forall\boldsymbol{\theta} \in \boldsymbol{\Theta}$. $\mathbf{a}_0 \leftarrow \mathbf{a}_0 + \eta \cdot \sum_{t=1}^{N}\{\Sigma_e^{-1}\mathbf{e}_t\}$, $\mathbf{e} = \boldsymbol{x}_t - A\boldsymbol{y}_t - \mathbf{a}_0$, $A \leftarrow \left[A + \eta \cdot \left\{\sum_{t=1}^{N}\left\{\Sigma_e^{-1}\mathbf{e}_t\boldsymbol{y}_t^T\right\} + \tau \cdot B\right\}\right] \circ \widetilde{A}$, $B = -a_{ij}/\gamma_{ij}$; $\Sigma_e \leftarrow \Sigma_e + \eta \cdot \sum_t \left\{-\frac{1}{2}\Sigma_e^{-1} + \frac{1}{2}\Sigma_e^{-1}\mathbf{e}_t\mathbf{e}_t^T\Sigma_e^{-1}\right\}$, $\gamma_{ij} \leftarrow \gamma_{ij} + \eta\tau \cdot \left\{-\frac{3}{2\gamma_{ij}} + \frac{a_{ij}^2}{2\gamma_{ij}^2}\right\}$, $\beta_{r\ell'} \leftarrow \beta_{r\ell'} + \eta \cdot \{\sum_{t=1}^{N} z_{r\ell' t} - \alpha_{r\ell'}N\}$, $\alpha_{rj_r} \leftarrow \exp\{\beta_{rj_r}\}/\sum_{\ell'=1}^{k_r}\exp\{\beta_{r\ell'}\}$, where $z_{r\ell t} = 1$ if $\ell = j_r^*$, otherwise $z_{r\ell t} = 0$, $\forall\ell \in \{1, \ldots, k_r\}$, α_{rj_r} is relaxed to updating β_{rj_r}, and $k_1 = \cdots = k_m = 2$, $\mu_{r1} = 0$, $\mu_{r2} = 1$, $\sigma_{r1}^2 = \sigma_{r2}^2 = 10^{-5}$, η is a small learning step size, and $\tau = 1, 0$ indicate whether sparsity is taken into account or not, respectively, and $\widetilde{A} = [\widetilde{a}_{ij}]_{n\times m} \in \{0, 1\}^{n\times m}$ is specified according to the connectivity constraint, and "$\circ$" is an element-wise product.
Convergence: repeat Step Y and Step $\boldsymbol{\Theta}$ until $\mathcal{H} = H(p\|q)$ converges, i.e., $\|\mathcal{H}^{new} - \mathcal{H}^{old}\|/\|\mathcal{H}^{old}\| < 10^{-4}$.