**AFRL-IF-RS-TR-2006-278**
**Final Technical Report**
**September 2006**

# DNA MICROARRAY-ASSISTED MODELING OF METABOLIC AND REGULATORY NETWORKS WITH APPLICATIONS TO BIO-DEFENSE

**Regents of the University of California**

*APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.*

**STINFO FINAL REPORT**

**The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.**

**AIR FORCE RESEARCH LABORATORY**
**INFORMATION DIRECTORATE**
**ROME RESEARCH SITE**
**ROME, NEW YORK**

# NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report was cleared for public release by the Air Force Research Laboratory Rome Research Site Public Affairs Office and is available to the general public, including foreign nationals. Copies may be obtained from the Defense Technical Information Center (DTIC) (http://www.dtic.mil).

AFRL-IF-RS-TR-2006-278 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

FOR THE DIRECTOR:

/s/

THOMAS E. RENZ
Work Unit Manager

/s/

JAMES A. COLLINS, Deputy Chief
Advanced Computing Division
Information Directorate

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

# REPORT DOCUMENTATION PAGE

*Form Approved*
**OMB No. 0704-0188**

| | | |
|---|---|---|
| **1. REPORT DATE** *(DD-MM-YYYY)*<br>SEP 2006 | **2. REPORT TYPE**<br>Final | **3. DATES COVERED** *(From - To)*<br>Sep 01 – Apr 06 |

**4. TITLE AND SUBTITLE**

DNA MICROARRAY-ASSISTED MODELING OF METABOLIC AND REGULATORY NETWORKS WITH APPLICATIONS TO BIO-DEFENSE

**5a. CONTRACT NUMBER**

**5b. GRANT NUMBER**
FA8750-01-2-0557

**5c. PROGRAM ELEMENT NUMBER**

**6. AUTHOR(S)**

James C. Liao and Vwani Roychawdhury

**5d. PROJECT NUMBER**
BIOC

**5e. TASK NUMBER**
M3

**5f. WORK UNIT NUMBER**
06

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**
Regents of the University of California
10920 Wilshire Blvd 1200
Los Angeles CA 90024-6523

**8. PERFORMING ORGANIZATION REPORT NUMBER**

**9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)**

DARPA/IPTO
3701 N. Fairfax Dr.
Arlington, VA 22203-1714

AFRL/IFTC
525 Brooks Rd
Rome NY 13441-4505

**10. SPONSOR/MONITOR'S ACRONYM(S)**

**11. SPONSORING/MONITORING AGENCY REPORT NUMBER**
AFRL-IF-RS-TR-2006-278

**12. DISTRIBUTION AVAILABILITY STATEMENT**
*APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED. PA# 06-650*

**13. SUPPLEMENTARY NOTES**

**14. ABSTRACT**

During the course of the project four software tools for BioSPICE were developed. These include Network Component Analysis (NCA), GeneScreen, 1cDNA, and MIcroArray Experimental Spice (MIAMESpice). NCA uses available connectivity information between genes and transcriptional factors and gene expression level time course data (obtainable through DNA microarray experiments) to estimate parameters and infer a gene transcriptional network through a Matlab analysis routine. GeneScreen processes gene expression data with a collection of computational statistic routines to extract significant gene association patterns. 1cDNA estimates confidence intervals for messenger RNA, mRNA expression levels in microarray experiments, including elimination of extreme outliers, quality filtering, normalization of the log10 signal intensity ratios, and assessment of expression levels. MIAMESpice packages raw and normalized data files from a set of related microarray experiments, saving all associated data from an experiment (or set of experiments) into one archive file. Users can also enter experimental annotations, array design information, and array design files.

**15. SUBJECT TERMS**
Microarray, DNA, Gene Expression, BioSPICE

| | | | | | |
|---|---|---|---|---|---|
| **16. SECURITY CLASSIFICATION OF:** | | | **17. LIMITATION OF ABSTRACT** | **18. NUMBER OF PAGES** | **19a. NAME OF RESPONSIBLE PERSON**<br>Thomas Renz |
| **a. REPORT**<br>U | **b. ABSTRACT**<br>U | **c. THIS PAGE**<br>U | UL | 63 | **19b. TELEPONE NUMBER (*Include area code)*** |

# Abstract

During the course of the project, we developed four software tools for BioSPICE. These include Network Component Analysis, NCA, GeneScreen, lcDNA, and MIcroArray Experimental Spice, MIAMESpice. NCA uses available connectivity information between genes and transcriptional factors and gene expression level time course data (obtainable through DNA microarray experiments) to estimate parameters and infer a gene transcriptional network through a Matlab analysis routine. GeneScreen processes gene expression data with a collection of computational statistic routines to extract significant gene association patterns. lcDNA estimates confidence intervals for messenger RNA, mRNA expression levels in microarray experiments, including elimination of extreme outliers, quality filtering, normalization of the log10 signal intensity ratios, and assessment of expression levels. MIAMESpice packages raw and normalized data files from a set of related microarray experiments, saving all associated data from an experiment (or set of experiments) into one archive file. Users can also enter experimental annotations, array design information, and array design files.

# Table of Contents

# List of Figures

# List of Tables

# Acknowledgment

During this DARPA project timeline, a number of graduate students and post-doctoral researchers have been involved in the project: Riccardo Boscolo, Katy Kao, Young-leoyl Yang, Simon Galbraith, Daniel Hyduke, and Linh M. Tran. In addition, Professor Chiara Sabatti of UCLA Human Genetics Department made significant contributions to the development of NCA.

## Summary

Network component analysis (NCA) is a method for transcriptome network decomposition besides principal component analysis (PCA) and independent component analysis (ICA), whose goal is to uncover underlying component signals. Previously, NCA as well as PCA and ICA were restricted to analyzing a maximum number of regulators equal to the total sample size. As such, the total number of source signal components computed is limited to the total number of experiments rather than the total number of biological regulators. Unfortunately, transcriptional regulation networks usually have less transcriptome data points than the number of regulators of interest. It is thus imperative to develop methods that allow realistic source signal extraction based on relatively few data points. On the other hand, such methods would inherently increase numerical challenges leading to multiple solutions. Therefore, solutions of both problems are needed.

We have improved NCA for transcription factor activity (TFA) estimation for limited data, based the fact that most genes are regulated by only a few transcription factors. This observation allows the derivation of a new uniqueness criterion which is tested during numerical iteration. In addition, data whitening and symmetric orthogonalization were used to circumvent the problem of local minimum when data error is considerable. A new algorithm was developed that provides a significant speed up of computation of TFAs from larger transcription networks than available data points.

We demonstrated the scalability of this algorithm using simulated limited data at increasing noise levels using an *Escherichia coli* connectivity network. To show biological application, we used this algorithm to deduce potential cell cycle regulated transcription factors in the *Saccharomyces cerevisiae* cell cycle microarray data (69 samples) using a connectivity network of 74 transcription factors. We find 22 *strongly periodic* cell cycle regulated factors that are statistically significant, 15 of which are known cell cycle regulators.

GeneScreen is a collection of computational statistic routines whose goal is to process gene expression data (typically from DNA microarray time-course experiments), extracting significant gene association patterns. The basic principle behind the technique used in GeneScreen is that stochastic processes, which do not appear to be correlated when observed as a whole, often present a great degree of resemblance when they can be explained by a common cause. In statistical terms, if we think of gene expression levels as random processes, these can appear as independent when observed in an isolated manner, but can present a large degree of dependency, conditionally on one or more hidden variables.

In GeneScreen, the conditional mutual information among genes in a cluster (also known as the co-information), is used as a measure of conditional dependency. It is well known that the mutual information between a set of variables is a positive quantity and it is equal to zero if and only if the variables are mutually independent. In the framework developed for GeneScreen it is assumed (for the sake of simplicity) that a single

controller variable (also named parent node or conditioning factor) influences the expression patterns of a set of genes, in such a way that although they might appear uncorrelated when observed as a whole, they tend to show a common behavior when the common cause is known. The mutual information is used as a scoring metric for its capability of detecting dependencies of high-order, as opposed to a simple correlation measure which is only capable of expressing second order dependencies in the data.

The guiding principle of lcDNA is to provide a fundamental framework for the design and analysis of two channel cDNA microarray data. Currently, lcDNA provides functions for removing outliers from data; this is accomplished by eliminating extreme intensity values (intensity value = signal intensity - background intensity), and for microarrays with replicate probes an additional quality filtering test can be used to remove the intensity values for replicate probes that are not consistent among themselves. In addition to providing a function for linear normalization of cDNA microarray data, lcDNA provides a rank invariant normalization procedure which takes into account the non-linearity of cDNA microarray data. Finally, lcDNA provides a robust hierarchical model for assessing the significance of the observed cDNA microarray data. In order to make these mathematical functions easily accessible, lcDNA has a graphical user interface that facilitates batch processing and visualization of the raw data.

MIAMESpice packages raw and normalized data files from a set of related microarray experiments, saving all associated data from an experiment (or set of experiments) into one archive file. Users can also enter experimental annotations, array design information, and array design files.

# Chapter 1. Network component analysis:

## 1.1 Introduction

High-throughput techniques in biology, such as DNA microarrays [1], have generated a large amount of data that can potentially provide systems-level information regarding underlying dynamics and mechanisms. These high-dimensional output data are typically the end products of low-dimensional regulatory signals driven through an interacting network. As illustrated in Fig. 1.1, the relationship between the lower dimensional regulatory signals (or states) and output data can be modeled by a bipartite networked system, where the output signals (e.g., gene expression levels) are generated by weighted functions of the intracellular states (e.g., the activity of the transcription factors). A major challenge in systems biology is to derive methodologies for simultaneous reconstructions of the hidden dynamics of the regulatory signals.

In recent years, statistical techniques for determining low-dimensional representations of high-dimensional data sets, e.g., Principal Component Analysis (PCA) [2] or Singular Value Decomposition (SVD) [3-5] and Independent Component Analysis (ICA) [6] have been applied successfully to deduce biologically significant information from high-throughput data sets. It is important to recognize that such dimensionality reduction techniques are not designed to address the hidden dynamics reconstruction problem addressed in this article. For example, PCA and ICA would both generate linear networks for interpreting the observed data set, where the regulatory signals are constrained to be mutually orthogonal and statistically independent, respectively. However, both the reconstructed signals and the networks do not match the real system, and provide only a phenomenological modeling of the observed data. In fact, as we show later, it is impossible to reconstruct the underlying regulatory state, without additional constraints.

Fortunately, for many biological systems partial prior knowledge about the connectivity patterns of the bipartite networks is beginning to become available via high-throughput experiments [7] or via data-mining of interaction knowledge [8-10], even though the detailed mechanisms remain undiscovered. Currently, however, it is unclear whether and how such qualitative connectivity information can be used to generate quantitative regulatory signals and further network details. Motivated by this pressing question in systems biology, we first derive a set of criteria for such prior connectivity information to be sufficient to solve the reverse engineering problem. We then provide a framework for the reconstruction process once such criteria are satisfied. This approach, termed network composition analysis (NCA), is experimentally validated using absorbance spectra of reconstructed biological solutions where the mixing (connectivity) pattern is known. Finally, we demonstrate the utility of NCA to genome-wide gene expression data in yeast *Saccharomyces cerevisiae* during cell cycle. As the bipartite network shown in Fig. 1.1 can represent many different types of data that are determined by multiple competing factors, the method developed here, NCA, can be applied to a large number of problems, where qualitative network structural information is available.

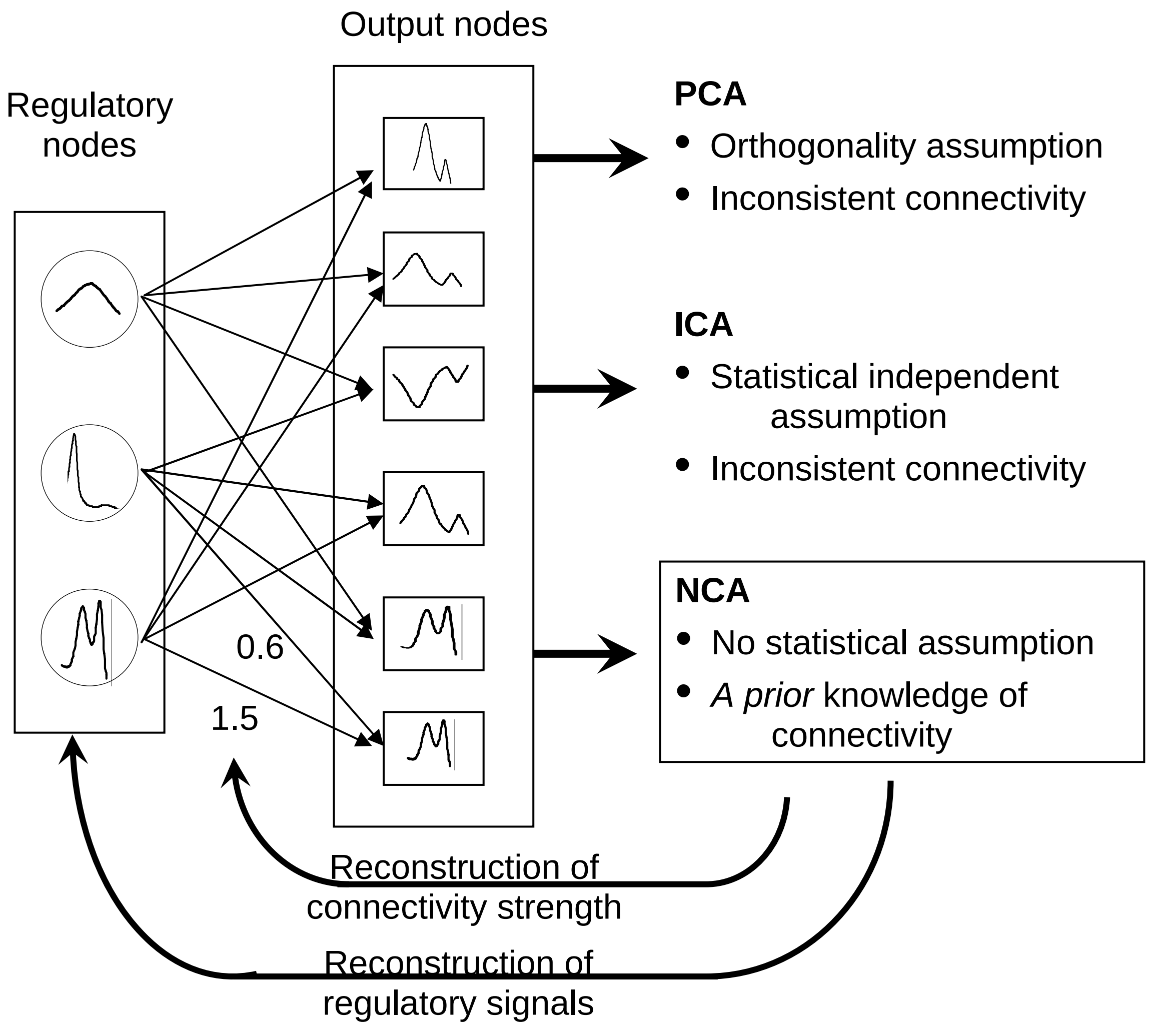

Output nodes
Regulatory nodes
PCA
Orthogonality assumption
Inconsistent connectivity
ICA
Statistical independent assumption
Inconsistent connectivity
NCA
No statistical assumption
A prior knowledge of connectivity
0.6
1.5
Reconstruction of connectivity strength
Reconstruction of regulatory signals

Figure 1.1. A regulatory system - in which the output data are driven by regulatory signals through a bipartite network. NCA takes advantage of partial network connectivity knowledge and is able to reconstruct regulatory signals and the weighted connectivity strength. For example, if a regulatory node or factor is known from experimental evidence to have negligible or no effect on an output signal, then the corresponding edge may be removed or equivalently, its weight set to zero. As discussed in the paper, such qualitative knowledge for a number of large biological systems is becoming available via high-throughput experiments. In contrast, traditional methods such as PCA and ICA depends statistical assumptions and cannot reconstruct regulatory signals or connectivity strength.

## 1.2 Mathematical Framework

The multidimensional data are organized in a format where *M* samples (or time points) of *N* output variables (such as the expression ratio of transcripts) is collected in the rows of a matrix [*E*] (size: *N rows*$\times$*M columns*). We seek to reconstruct a model of the type:

$$[E] = [A][P] \tag{1}$$

Here the matrix [P] (size: $L\times M$) consists of samples of *L* regulatory signals, where *L* is in general much smaller than *N*, thus resulting in the reduction in dimensionality. The matrix [*A*] (size: $N\times L$) encodes the connectivity strength between the regulatory layer and the output signals (Fig. 1.1). Eq. 1 represents the linear approximation of any detailed mechanistic model and is commonly used as the first approximation when the latter is unavailable.

The decomposition of a matrix [E] into two matrices, [A] and [P], according to Eq 1 is an inverse problem whose solution is in general not uniquely defined unless further assumptions on the matrices [*A*] or [*P*] are made. This can be seen by introducing a non-singular matrix [*X*] (*L x L*) such that [ $\overline{A}$ ] = [*A*][*X*] and [ $\overline{P}$ ] = [$X^{-1}$][*P*], and:

$$[E] = ([A][X])\,([X^{-1}][\,P]) = [\,\overline{A}\,][\,\overline{P}\,] \tag{2}$$

Thus, without further constraints, [*E*] cannot be uniquely decomposed to [*A*] and [*P*] according to Eq. 1. Conventional approaches, such as PCA and ICA typically seek a matrix [*A*] such that the resulting reconstructed signal matrix [*P*] satisfies orthogonality or independence criteria, respectively. When dealing with data generated from structured networks, such as biological systems, these decomposition techniques present two drawbacks. First, the implicit statistical assumptions on the regulatory signals lack biological foundation. Second, the reconstructed connectivity structure is unlikely to be consistent with the underlying network structure. Therefore, we seek a decomposition method that makes no assumption on the statistical properties of the regulatory signals and that, at the same time, allows proper handling of the prior knowledge on the structure characterizing a given system.

### 1.3 Criteria for Network Component Analysis

According to Eq. 2, multiple [*A*]'s and [*P*]'s can reconstruct data [*E*] equally well. However, when certain connectivity constraints are imposed on [A], the [X] matrix in Eq. 2 can only be diagonal (see Appendix A for proof). Furthermore, when [A] has full column rank and [P] has full row rank, Eq. 2 represents all the possible alternative solutions of the decomposition of [E] (see Appendix A for proof). Under these conditions, Eq. 1 results in a unique decomposition of the data, up to a scaling factor. Therefore, certain network structure enables the decomposition of data. This type of decomposition is defined as network component analysis (NCA). In summary, the criteria for NCA to be feasible are

1. The connectivity matrix [*A*] must have full-column rank.
2. When a node in the regulatory layer is removed along with all the output nodes connected to it, the resulting network must be characterized by a connectivity matrix that still has full-column rank. This condition implies that each column of [A] must have at least *L*-1 zeros.

3. [P] must have full row rank. In other words, each regulatory signal cannot be expressed as a linear combination of the other regulatory signals.

If these criteria are satisfied, the data matrix [E] can be uniquely decomposed to a connectivity matrix [A] and signal matrix [P] when a scaling rule applies. The matrix [A] contains the estimated connectivity strength on each edge, while the matrix [P] contains the regulatory signals of each regulatory node.

In order to test the feasibility of NCA, one first constructs an initial [A] matrix based on knowledge of connectivity. The [A]entry at i$^{th}$ row and j$^{th}$ column ($a_{ij}$) represents the control strength of each regulatory node j on output node i. If this pair is not connected, the value for $a_{ij}$ is zero. Otherwise, it is arbitrarily set to a non-zero number as an initial value. Thus, the [A] matrix has a dimension of *NxL*, where *N* is the number of output nodes and *L* is the number of regulatory nodes (e.g. transcription factors) considered. Given the initial connectivity matrix [A] (*N x L*), we first test whether it has full column rank (Criterion 1). If this criterion is satisfied, we then form a set of reduced matrices [$Ar_j$], by removing the j$^{th}$ column and all the rows of A corresponding to the non-zero entries of its j$^{th}$ column. For example, if:

$$[A] \equiv \begin{bmatrix} 5 & 0 & 0 & 2 \\ 0 & 2 & 4 & 0 \\ 3 & 0 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 6 \\ 0 & 8 & 0 & 0 \end{bmatrix} \qquad (3)$$

then

$$[Ar_2] \equiv \begin{bmatrix} 5 & \cancel{0} & 0 & 2 \\ \cancel{0} & \cancel{2} & \cancel{4} & \cancel{0} \\ 3 & \cancel{0} & 0 & 0 \\ 0 & \cancel{0} & 5 & 0 \\ 0 & \cancel{0} & 0 & 6 \\ \cancel{0} & \cancel{8} & \cancel{0} & \cancel{0} \end{bmatrix} = \begin{bmatrix} 5 & 0 & 2 \\ 3 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 6 \end{bmatrix} \qquad (4)$$

Criterion 2 is satisfied if and only if, for any possible choice of a single regulatory node, the corresponding reduced matrix has rank equal to *L-1*.

Criterion 3 cannot be tested *a priori*, but it implies the necessary condition that *L* (the number of regulatory nodes) must be less than *M* (the number of data points). If L is indeed less than M, the matrix [P] is likely to have full row rank for real biological data. This rank condition should be checked after [P] is obtained from NCA. If *L>M*, a sub-network should be generated to reduce *L*. This can be done by removing selected regulatory nodes together with all the output nodes they control. If the subsystem satisfy *L < M*, then proceed to test the other criteria. If the sub-system satisfies all three criteria, then it is NCA-compliant.

A simple example is shown in Fig. 1.2, which presents a completely identifiable network (Fig. 1.2a) and an unidentifiable network (Fig. 1.2b), although the two matrices have an identical number of constraints (zero entries). The network in Fig. 1.2b does not satisfy the identifiability criterion because of the connectivity pattern of $R_3$.

### 1.4 Method for Network Component Analysis

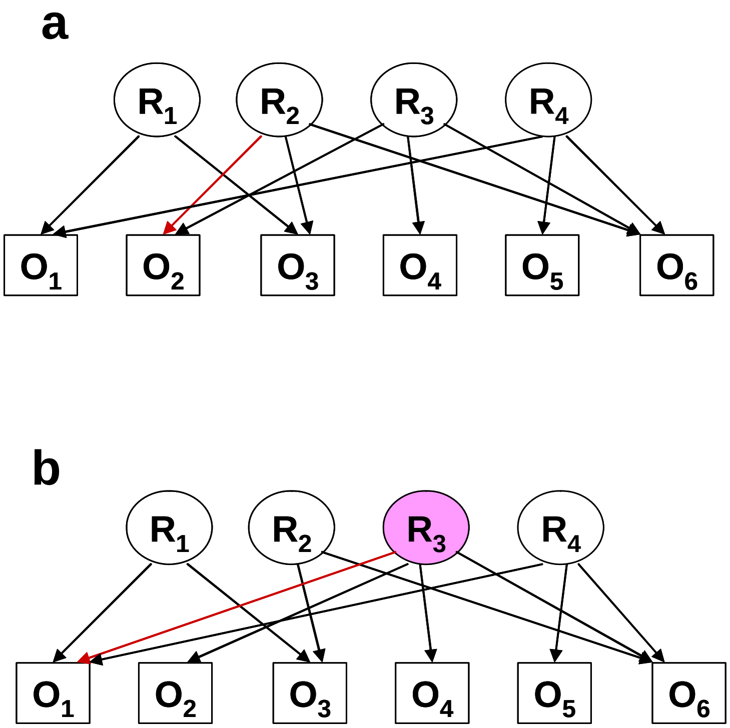
Once the identifiability of a given system has been established, the regulatory signals, [P], and the connectivity strength, [A], can be reconstructed through the following procedure. An initial guess for the connectivity matrix *A* is formed by setting to zero all the elements corresponding to missing edges between the regulatory layer and the output layer. The remaining elements can be initialized to an arbitrary value. Since the experimental measurements are noisy, an exact solution to the decomposition problem does not exist in general. However, when the above NCA criteria are satisfied, the estimation problem becomes well-posed, and a solution that provides the best fit in the least-squares sense can be computed. We proceed by minimizing the following objective function:

$$\begin{aligned} &\min \left\| [E]\text{-}[A][P] \right\|^2 \\ &s.t.\ A \in Z_0 \end{aligned} \qquad [5]$$

where $Z_0$ is the topology induced by the network connectivity pattern. Additional constraints on the nature of the regulation (positive or negative) can also be included in the optimization framework, but are not strictly required by the method in general.

The above objective function is equivalent to a constrained maximum likelihood procedure in the presence of Gaussian noise. The actual estimation of [*A*] and [*P*] is performed by using a two-step least squares algorithm, which exploits the bi-convexity properties of linear decompositions (Appendix B). The variability of our estimates is assessed using a bootstrap procedure (Appendix C).

Normalization of [A] and [P] can be achieved by a non-singular diagonal matrix [X] in Eq. 1.**2**. The elements of [X] should be selected according to the physical or biological nature of the data set. As an example, the columns of [A] (for each regulatory node across all the output node) can be normalized so that the mean absolute value of the non-zero elements is equal to the number of controlled output nodes. With this normalization, the rows of [P] for different regulatory nodes represent the average effect of the regulator on the output nodes it controls, and the columns of [A] represent the relative control strength for the same regulator on different output nodes.



$$\textit{initial}\ [A]=\begin{bmatrix}2&0&0&1\\0&5&4&0\\8&1&0&0\\0&0&6&0\\0&0&0&7\\0&3&5&1\end{bmatrix}$$

$$\textit{initial}\ [A]=\begin{bmatrix}5&0&3&1\\0&0&4&0\\2&1&0&0\\0&0&3&0\\0&0&0&1\\0&4&8&9\end{bmatrix}$$

Figure 1.2. **a**) A completely identifiable network, and **b**) an unidentifiable network. Although the two initial [A] matrices describing the network matrices have an identical number of constraints (zero entries). The network in **b** does not satisfy the identifiably conditions because of the connectivity pattern of $R_3$. The edges in red are the differences between the two networks.

### 1.5 Experimental Validation of Network Component Analysis

In order to verify experimentally the NCA method described above, we used a network of 7 hemoglobin solutions as test case. Each solution contains a combination of three components: oxyhemoglobin (oxyHb), methemoglobin (metHb) and cyano-methemoglobin (cyanoHb). These solutions were prepared according to Appendix D and the absorbance spectra were taken between 380 and 700 nm with 1 nm increments. According to Beer-Lambert Law, the absorbance spectra can be described as follows:

$$[Abs]=[C][\varepsilon] \qquad \textbf{[6]}$$

where the rows of [*Abs*] are the absorbance spectrum of each solution at each wavelength, the columns of the connectivity matrix [*C*] are the concentrations of each component, and the rows of [$\varepsilon$] are the spectra of pure components. The connectivity diagram of this solution network is shown in Fig. 1.3a, where the components of the four solutions are known, but the concentration of each component and the pure-component spectra are assumed to be unknown and will be determined from the solution spectra using NCA.

The connectivity matrix [C] is initiated by using non-zero random numbers and 0's for components present or absent, respectively, in the solution according to Fig. 3a. The initial [*C*] matrix was verified to satisfy the NCA criteria. The decomposition was carried out according to the NCA algorithm briefly described above and detailed in Appendix B. Results (Fig. 1.3b) show that the pure component spectra ([ε]) resulted from NCA agree well with the true spectra obtained from independent measurements of pure components. Despite the similarity among the pure component spectra, NCA was able to resolve the differences. In contrast, SVD or ICA cannot reconstruct the pure component spectra faithfully (Fig. 1.3b). In addition, the concentrations estimated from the [C] matrix show satisfactory agreement with the true concentrations (Table 1.1). Note that the spectra were decomposed using only the known components, but not the concentrations of the solutions. However, the NCA method was able to simultaneously determine the concentrations of each component as well as the spectra of pure components.

## 1.6 Application to Gene Expression Regulation.

Since the NCA method is experimentally verified using a test system, we now explore its utility in a more challenging system – transcriptional regulation in yeast. In general, transcription of genes is controlled by a smaller number of transcription factors, whose activation via post-translational modification or ligand binding is the determining factor for gene expression. The activated form of a transcription factor, rather than its expression level, is what controls promoters and dictates the physiological state of the cell. We consider the signal transmitted to different promoters as the transcription factor activity (TFA). Correspondingly, the control strength (CS) quantifies how each promoter receives the signal and it reflects the relative contribution of the transcription factor to the expression of different genes (Fig. 1.1). Determining TFAs provides a basis for pinpointing perturbations caused by drug effects, genetic mutation, or complex environmental challenges. However, these regulatory quantities, even individually, are difficult to measure.

Typically, the first-order regulatory relationships between transcription factor and gene expression is represented by a bipartite network similar to that shown in Fig. 1.1, where the connections (or edges) represent the binding of a transcription factor to the gene's promoter region. A recently introduced genome-wide location analysis (11,12) allows the detection of transcription factor binding to promoter regions, and provides a method for reconstructing such genome-wide transcription connectivity diagrams (Fig. 1.1). The availability of such information allows further inference of regulatory signal represented by the TFA, and the CS of the transcription factors on the genes.

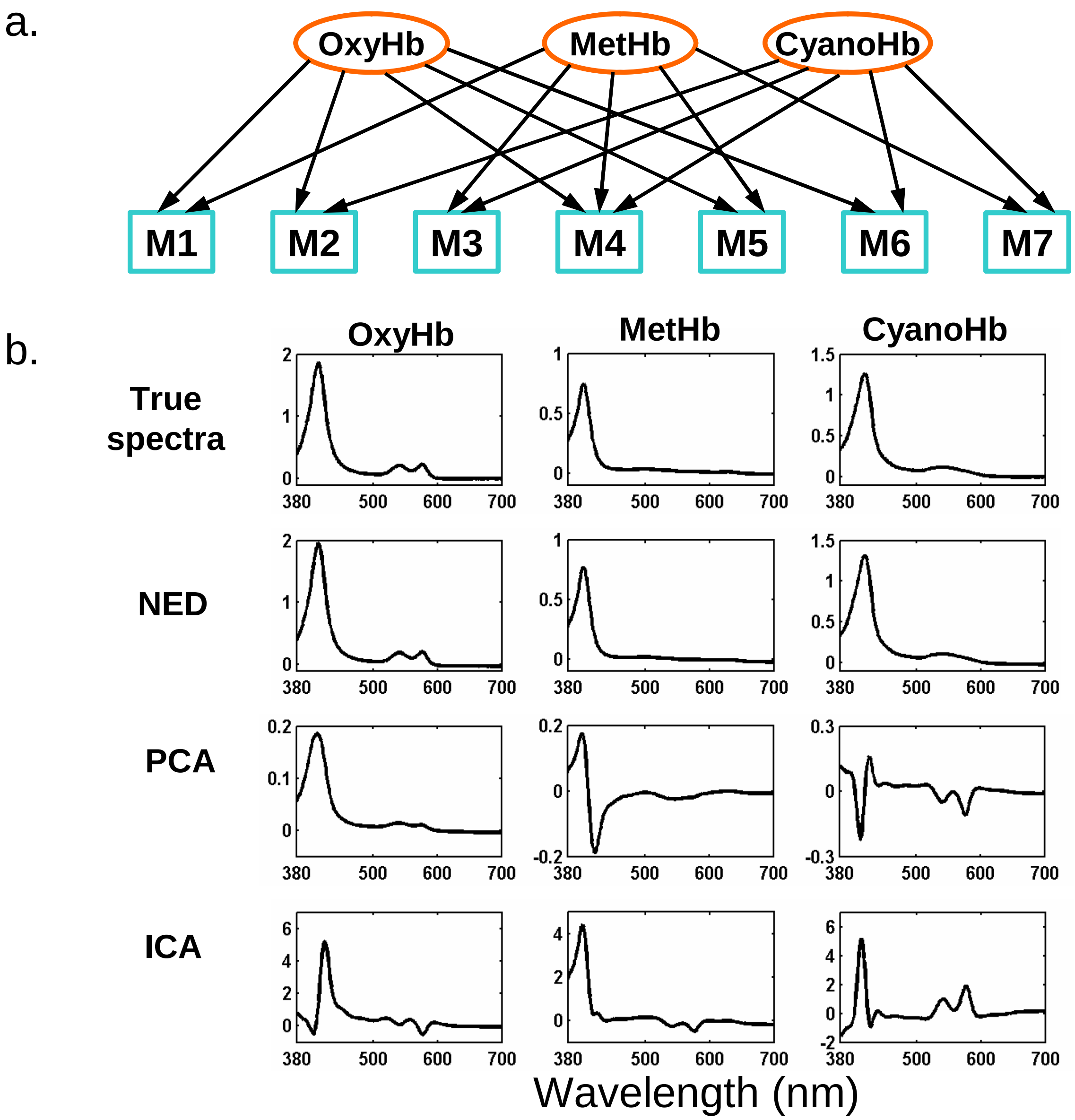


Figure 1.3 Experimental validation of the NCA method using absorbance spectra of hemoglobin solutions. **a**) The connectivity (mixing) diagram of the seven Hb solutions from three pure components which serve as the regulatory nodes. **b**) The regulatory signals (pure component spectra) derived from NCA agree well with the true values, while those derived from PCA or ICA do not.

Table 1.1**:** Concentrations of the hemoglobin solutions estimated from the NCA analysis agree reasonably well with the true values (in parenthesis).

| | OxyHb<br>μM | MetHb<br>μM | CyanoHb<br>μM |
|---|---|---|---|
| M1 | 0.13<br>(0.13) | 3.8<br>(4.3) | 0<br>(0) |
| M2 | 5.1<br>(6.4) | 0<br>(0) | 5.8<br>(5.8) |
| M3 | 0<br>(0) | 3.8<br>(4.3) | 1.2<br>(1.2) |
| M4 | 0.13<br>(0.13) | 3.3<br>(3.8) | 1.2<br>(1.2) |
| M5 | 2.6<br>(3.8) | 2.9<br>(3.3) | 0<br>(0) |
| M6 | 2.6<br>(2.6) | 0<br>(0) | 9.3<br>(9.3) |
| M7 | 0<br>(0) | 1.9<br>(2.4) | 5.8<br>(5.8) |

To analyze the gene expression data, we approximate the relationship between transcription factor activities and gene expression levels, by a log-linear model of the type:

$$\frac{E_i(t)}{E_i(0)} = \prod_{j=1}^{L}\left(\frac{TFA_j(t)}{TFA_j(0)}\right)^{CS_{ij}} \qquad \textbf{[7]}$$

where $E_i(t)$ is the gene expression level, $TFA_j(t)$, j=1,…,$L$ is a set of transcriptional regulator activities, and $CS_{ij}$ represents the control strength of transcription factor $j$ on gene *i*. Log-linear models are used in several disciplines as a standard tool to approximate nonlinear systems, and have the following advantages: (i) Since they represent linear approximations (i.e., in the log-log space), they inherit the usual benefits of linearization, i.e., they are locally accurate and computationally tractable. (ii) Unlike standard linear models (i.e., in the original data space), the log-linear models still allow a restricted nonlinear relationship between inputs and outputs. In the case of DNA microarray data, since gene expression levels are typically measured with respect to a reference level, it is particularly convenient to work with relative quantities as in Eq. **7**. As a further justification of our log-linear model, we show in Appendix E that Eq. **7** can be derived by linearizing a phenomenological model, based on Hill's equations, that has been used previously to describe the relationship between promoter activity and transcription factor activities [13]. In particular, the value of $CS_{ij}$ is determined by the Hill coefficients and the transcription factor affinity to the promoter region. The following expression in a matrix form can be derived from Eq. 7 after taking the logarithm:

$$\log [Er] \ = \ [CS] \ \log [TFAr] \tag{8}$$

where the elements $Er_{ij}(t)= E_{ij}(t)/E_{ij}(0)$ and $TFAr_{kj}(t)= TFA_{kj}\,(t)/TFA_{kj}(0)$ are the relative gene expression levels and transcription factor activities. The rows of *[Er]*(size: *NxM*) and *[TFAr]* (size: *LxM*) are the time-courses of relative gene expression levels and transcription factor activities, respectively, and *[CS]* (size: *NxL*) is the matrix with elements $CS_{ik}$. Several linear decompositions of the matrix log*[Er]* have been used extensively in the study of gene expression array: as an example Alter et al. (14) propose to use SVD in order to find the lower dimensional projections of the expression data that present the largest degree of variation. By using SVD, one implicitly assumes that the TFAs possess an orthogonal structure. Alternative approaches based for example on ICA have also been investigated [6]. These aim at finding a decomposition of the data into statistically independent basis functions, using an unsupervised learning method. Although any of these decomposition techniques have strong statistical foundations, their molecular basis is difficult to pin-point.

### 1.7 Application to *Saccharomyces cerevisiae* Cell Cycle Regulation

In eukaryotes, the transcriptional regulation can be grouped in terms of DNA-binding transcription factors, which recruit chromatin-modifying enzymes and components of transcription apparatus. Here, we used cell cycle regulation in S. *cerevisiae* as an example to test the applicability of the above approach. The connectivity between transcription factors and genes was obtained from the genome-wide location analysis [7]. Microarray data sets used for the yeast cell cycle were taken from cultures synchronized by elutriation, α-factor arrest, and arrest of a cdc15 temperature sensitive mutant [15]. We focused on the 11 transcription factors which are known to be related to cell-cycle regulation [7]. Initially, 570 genes regulated by these 11 transcription factors were selected from a total of 1134 genes in the data set. Because other transcription factors also contribute to the regulations of these genes, the network contains 44 transcription factors. This network was checked for NCA compliance by examining each of the reduced matrices for its rank.  By trimming transcription factors and associated genes that violate this test, the final data set contains 441 genes with 33 transcription factors.

Interestingly, the NCA provides a very good fit to most of the microarray expression data (Fig. 1.4a). The columns of [CS] were normalized so that the mean absolute value of the non-zero elements is equal to the number of controlled genes. Thus, the rows of [TFA] for different transcription factors represent the average effect of the regulator on the genes it controls, and the columns of [CS] represent the relative control strength for the same regulator on different genes. It is recognized that binding assays may yield false positive or false negative results, and that transcription factor binding does not guarantee regulation [16]. The general agreement between data and the NCA model provides evidence for the regulatory role of a transcription factor with respect to a particular gene. In particular, a very small value of the CS for a particular gene-transcription factor connection is usually indicative of poor likelihood for such regulatory role.

The dynamics of TFAs (Fig. 1.4b) reveal the role of each transcription factor during cell cycle regulation.  In contrast, the expression ratios obtained from DNA microarray experiments (Fig. 1.4c) do not reveal regulatory features by themselves.  Fig. 1.4c shows that TFAs of most of the recognized cell cycle regulators exhibited a cyclic behavior.

Among the 11 recognized cell cycle regulators [7], Stb1, Mcm1, and Mbp1 exhibited the greatest amplitudes in their TFAs, whereas Skn7 and Swi6 showed little cyclic behavior. Swi6 has been shown to associate with Mbp1 or Swi4 [17] while Skn7 has to bind to Mbp1 to exert cell cycle regulation [18].  Perhaps the oscillatory feature needed for cell cycle regulation comes from their binding partners. Indeed, Skn7 is also involved in oxidative stress response and heat shock response, and thus oscillatory feature in this transcription factor is not expected.

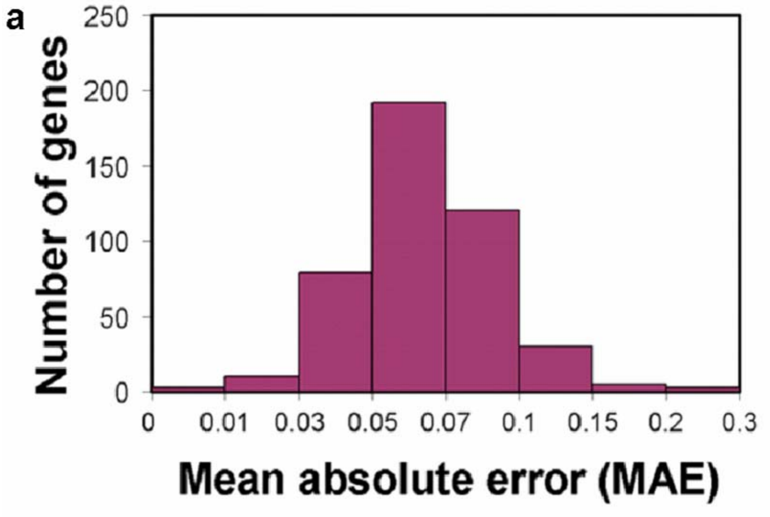
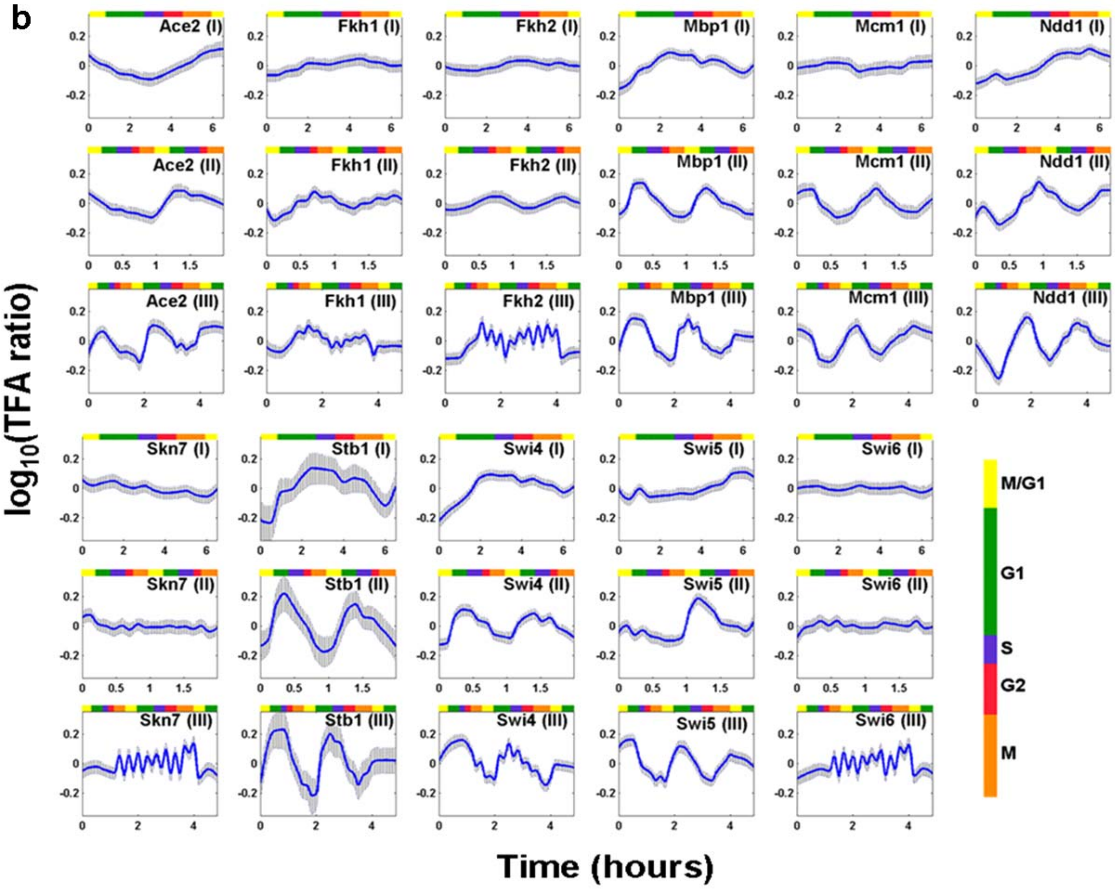
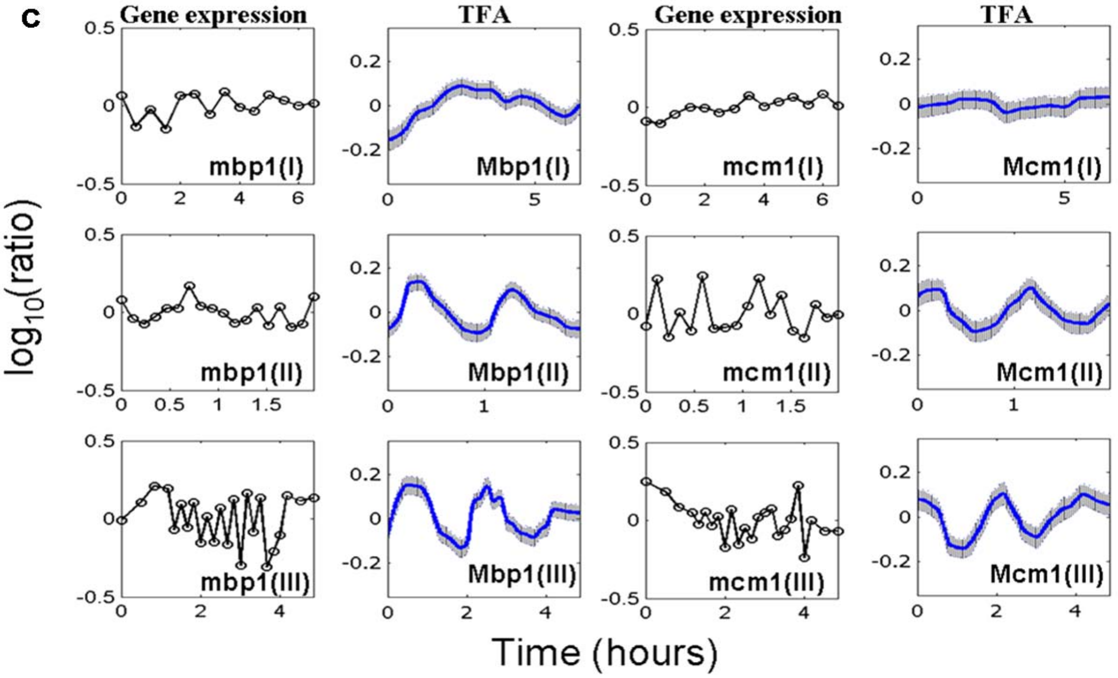


Figure 1.4. *S. cerevisiae* cell cycle regulation **a)** The histogram of mean absolute errors (MAE) shows that the majority of the genes were fitted reasonably well. MAE is defined as

$$\mathrm{MAE} = \frac{\sum_{i=1}^{N} \left| \log_{10} ER_i - \log_{10} \overline{ER_i} \right|}{N}$$



**b)** The dynamics of the TFAs for 11 transcription factors involved in cell cycle regulation. Different stages in the cell cycle are indicated by the color code. The three rows represent experiments using different synchronization methods: elutriation, α factor arrest, and arrest of a *cdc15* temperature-sensitive mutant. Shaded areas span four standard deviations (estimated using a bootstrap technique as explained in Appendix C).



**c)** The comparison between expression levels and activities of selected transcription factors shows that the expression levels do not exhibit an oscillatory behavior while TFA's levels do.

### 1.8 Conclusion

We developed a novel data decomposition method, NCA, for reconstructing regulatory signals and control strengths using partial and qualitative network connectivity information. As stated above, this method contrasts traditional methods such as PCA and ICA in that it does not make any assumption regarding the statistical properties of the regulatory signals. Rather, network structure, even if incompletely known, is used to generate a network-consistent representation of the regulatory signals. This method is validated experimentally using absorbance spectra and then applied to transcriptional regulatory networks.

Many other types of large-scale data, such as neuronal signals, signal transduction data, metabolic fluxes, and protein-protein interaction information, may potentially be modeled as the output of underlying functional networks that are driven by regulatory signals. Thus for determining the underlying regulatory states, the network connectivity structures cannot be ignored. In these cases, traditional methods such as PCA and ICA will yield to NCA as the underlying network topologies are determined or inferred at an iterative process to aid the deduction of network topology. Even when the network structural information is partially known, trial network structures can be used to generate regulatory signals.

As illustrated in this paper, perhaps the most immediate impact of the NCA analysis will be for DNA microarray data. Our technique builds on earlier pioneering work in related areas [13,19]. For example, Ronen *et al.* [13] propose a method for estimating the kinetic parameters of simple regulatory network architectures, by fitting a kinetic model to high-resolution promoter activity data. Such a method is capable of dealing with a basic architecture, where all operons are regulated by a single transcription factor, and where the regulatory mechanism is well characterized. Recently, Gardner et al. (19) presented a combined experimental-computational technique for inferring genetic network structure. This technique determines network connectivity in systems where both the input and the output signals are accessible.

Although the connectivity information between genes and transcription factors is not currently available for all organisms, it is expected that such information will be widely accessible in the near future using various methods [7, 11, 19-20]. Meanwhile, the amount of large-scale gene expression data obtained using either microarray or equivalent technologies are increasing rapidly, and the accuracy of these data is expected to improve. We expect that with both types of data widely available, quantitative reconstructions of transcriptional regulatory networks using NCA analysis will be routinely performed.

## 1.9 Reference:

[1] Schena, M., Shalon, D., Davis, R.W. & Brown, P.O., *Science* 270: 467-470, 1995

[2] Raychaudhuri, S., Stuart, J.M. & Altman, R.B.*Pac. Symp. Biocomput.*, 455-466, 2000

[3] Holter, N.S., Mitra, M., Maritan, A., Cieplak, M., Banavar, J.R. & Fedoroff, N.V., *Proc. Natl. Acad. Sci. USA*, 97: 8409-8414, 2000

[4] Holter, N.S., Maritan, A., Cieplak, M., Fedoroff, N.V. & Banavar, J.R., *Proc. Natl. Acad. Sci. USA*, 98:1693-1698, 2001

[5] Yeung, M.K., Tegner, J. & Collins, J.J., *Proc. Natl. Acad. Sci. USA*, 99:6163-6168, 2002

[6] Liebermeister, W., *Bioinformatics* 18:51-60, 2002

[7] Lee, T.I., Rinaldi, N.J., Robert, F., Odom, D.T., Bar-Joseph, Z., Gerber, G.K., Hannett, N.M., Harbison, C.T., Thompson, C.M., Simon, I., *et al*., *Science* 298:799-804, 2002

[8] Roth, F.P., Hughes, J.D., Estep, P.W. & Church, G.M., *Nat. Biotechnol*.**,** 16: 939-945, 1998

[9] Bussemaker, H., Li, H. & Siggia, E., *Proc. Natl. Acad. Sci. USA*, 97:10096-10100, 2001

[10] Bussemaker, H., Li, H. & Siggia, E.*Nat. Genet.*, 27:167-171, 2000

[11] Ren, B., Robert, F., Wyrick, J.J., Aparicio, O., Jennings, E.G., Simon, I., Zeitlinger, J., Schreiber, J., Hannett, N., Kanin, E., *et al*., *Science*, 290:2306-2309, 2000

[12] Zeitlinger, J., Simon, I., Harbison, C.T., Hannett, N.M., Volkert, T.L., Fink, G.R. & Young, R.A., *Cell*, 113:395-404, 2003

[13] Ronen, M., Rosenberg, R., Shraiman, B. I., & Alon, U., *Proc. Natl. Acad. Sci. USA,* 99:10555-10560, 2002

[14] Alter, O., Brown, P.O. & Botstein, D., *Proc. Natl. Acad. Sci. USA*, 97:10101-10106, 2000

[15] Spellman, P.T., Sherlock, G., Zhang, M.Q., Iyer, V.R., Anders, K., Eisen, M.B., Brown,.P.O., Botstein, D. & Futcher, B., *Mol. Biol. Cell*, 9:3273-3297, 1998

[16] Futcher, B., *Curr. Opin. Cell. Biol.*, 14:676-683, 2002

[17] Simon, I., Barnett, J., Hannett, N., Harbison, C.T., Rinaldi, N.J., Volkert, T.L., Wyrick, J.J., Zeitlinger, J., Gifford, D.K., Jaakkola, T.S., *et al*. *Cell*, 106:697-708, 2001

[18] Bouquin, N., Johnson, A. L., Morgan, B. A. & Johnston, L. H., *Mol. Biol. Cell*., 10:3389-3400, 1999

[19] Gardner, T.S., di Bernardo, D., Lorenz, D. & Collins, J.J., *Science,* 301:102-105, 2003

[20] Iyer, V.R., Horak, C.E., Scafe, C.S., Botstein, D., Snyder, M. & Brown, P.O., *Nature,* 409:533-538, 2001

## 1. 10 Appendix A: Proof of Network Component Analysis

**Definition: Connectivity Pattern**: Given a matrix A, and a set $Z_o \subset Z^2$ we say that A is characterized by the connectivity pattern imposed by $Z_0$ if and only if: $a_{ij} \equiv o$ *for* $i, j \in Z_o$. Note that a zero in $a_{ij}$ represents the absence of an edge connecting regulator *j* to output *i* in a bi-partite network.

**Network Component Analysis Theorem:** Given a matrix E (*N x M*), where:

$$E = A\ P \qquad \text{(A1)}$$

A: (*N x L*)

P: (*L x M*)

if the following conditions are satisfied, then the decomposition of E into A and P is unique up to a scaling diagonal matrix X (*LxL*) . Any alternative decomposition $E = \overline{A}\overline{P}$ where $\overline{A}$ (*N x L*) has full column rank and has the same connectivity pattern $Z_0$ as A, there exists a diagonal non-singular matrix X (*L x L*), such that:

$$\overline{A} = AX^{-1} \qquad \text{(A2)}$$

$$\overline{P} = XP \qquad \text{(A3)}$$

**Condition 1.** A has full column rank and represents a connectivity pattern defined by $Z_0$.
**Condition 2**. The reduced matrix $Ar_j$, defined by eliminating the j<sup></sup>th column of A and all the **rows** corresponding to non-zero elements in that **column** (see example in equation A17), has rank *L-1* for all j.
**Condition 3.** P has full row rank.

**Proof**:

Given A, P, $\overline{A}$, and $\overline{P}$ such that

$$E = AP = \overline{A}\overline{P} \qquad \text{(A4)}$$

we are going to show there exists a **diagonal** non-singular matrix X (*L x L*), such that

$$\overline{A} = AX^{-1} \qquad \text{(A2)}$$

$$\overline{P} = XP \qquad \text{(A3)}$$

Since $\overline{A}$ has full column rank we can write:

$$\overline{A}^T AP = \overline{A}^T \overline{A}\overline{P} \tag{A5}$$

$$\overline{P} = \left(\overline{A}^T \overline{A}\right)^{-1} \overline{A}^T AP \equiv XP \tag{A6}$$

From (A4) we obtain:

$$AP = \overline{A}XP$$

$$\left(A - \overline{A}X\right)P = 0 \tag{A7}$$

Since P has full row rank (**Condition 3**), the above equation implies that

$$A = \overline{A}X \tag{A8}$$

Now we need to show that X can only be diagonal if **Condition 2** is satisfied.

We can write (A8) as:

$$a_{ij} = \sum_{l=1}^{L} \overline{a}_{il} x_{lj} \tag{A9}$$

This equation can be rewritten in the following staggered form

$$\begin{bmatrix} Ac_1 \\ Ac_2 \\ : \\ : \\ Ac_L \end{bmatrix} = \begin{bmatrix} \overline{A} & & & 0 \\ & \overline{A} & & \\ & & : & \\ & & & : \quad \\ 0 & & & \overline{A} \end{bmatrix} \begin{bmatrix} Xc_1 \\ Xc_2 \\ : \\ : \\ Xc_L \end{bmatrix} \tag{A10}$$

where Ac$_i$ and Xc$_i$ are the i$^{th}$ columns of matrices A and X, respectively.
Here, the connectivity pattern Z$_0$ imposes the following constraints for the elements in X:

$$a_{ij} = \sum_{l=1}^{L} \overline{a}_{il} x_{lj} \equiv 0, \qquad\qquad for \quad i, j \in Z_0 \tag{A11}$$

Furthermore, in the above equation the diagonal terms of X do not appear since, if $a_{ij} = 0$ we necessarily have $\overline{a}_{ij} = 0$ and that

$$\bar{a}_{ij} x_{jj} = 0 x_{jj} \tag{A12}$$

In other words, equation (A11) constrains the *L(L-1)* off-diagonal elements in X, but not the diagonal elements. Thus, taking the zero elements in A (defined by $Z_o$) and eliminating the diagonal terms in X, we convert equation (A10) to

$$0 = \begin{bmatrix} \overline{Ar_1} & & & & 0 \\ & \overline{Ar_2} & & & \\ & & : & & \\ & & & : & \\ 0 & & & & \overline{Ar_L} \end{bmatrix} \begin{bmatrix} Xr_1 \\ Xr_2 \\ : \\ : \\ Xr_L \end{bmatrix} \tag{A13}$$

$$\text{or} \qquad \overline{A_u} X_r = 0 \tag{A14}$$

where $Xr_i$ is the i$^{th}$ column of X after deleting the diagonal element. $\overline{A_u}$ is defined as

$$\bar{A}_u \equiv \begin{bmatrix} \bar{A}r_1 & 0 & & 0 \\ 0 & \bar{A}r_2 & & \\ : & & : & 0 \\ 0 & & & \bar{A}r_L \end{bmatrix} \tag{A15}$$

where $\bar{A}r_i$ is the reduced matrix derived from $\bar{A}$ by eliminating i$^{th}$ column and all the rows that contain non-zero elements in that column. For example, if

$$[\bar{A}] \equiv \begin{bmatrix} 5 & 0 & 0 & 2 \\ 0 & 2 & 4 & 0 \\ 3 & 0 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 6 \\ 0 & 8 & 0 & 0 \end{bmatrix} \tag{A16}$$

then

$$[\bar{A}r_2] \equiv \begin{bmatrix} 5 & 0 & 2 \\ 3 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 6 \end{bmatrix} \tag{A17}$$

Since A and $\overline{A}$ have the same connectivity pattern, the rank of $\overline{A_u}$ is the same as the rank of $A_u$, which is equal to *L(L-1)* according to **Condition 2**. Note that there are *L(L-1)* unknown variables in $X_r$. Since $\overline{A_u}$ has rank *L(L-1),* the unique solution of the linear system of equations defined by (A12) is $X_r = 0$. Thus,

$$x_{ij} = 0 \qquad for \quad i \neq j \tag{A18}$$

$$x_{ij} = arbitrary \qquad for \quad i = j \tag{A19}$$

Therefore, X is a diagonal matrix. Because A is full column rank, a zero diagonal entry would violate (A8). This completes the proof.

### 1. 11 Appendix B – Network Component Analysis Algorithm

Given a matrix E (*N x M*) and a connectivity pattern $Z_0$, the goal is to find a decomposition of the type:

$$E = A\,P \tag{B1}$$

with $A=[a_{i,j}]$ (*N x L*) characterized by the connectivity pattern defined by $Z_0$, and $P=[p_{i,j}]$ being a matrix of size (*L x M*). We further assume that *A* and *P* satisfy the hypotheses of Theorem 1 (Appendix A), so that this decomposition is unique up to diagonal scaling. The optimal *A* and *P* can be found by solving:

$$\begin{aligned} &\min_{A,P} \| E - AP \|^2 \\ &\text{s.t. } A \in \mathcal{A}(Z_0) \\ &\qquad a_{i,j}^{(l)} \leq a_{i,j} \leq a_{i,j}^{(u)} \\ &\qquad p_{i,j}^{(l)} \leq p_{i,j} \leq p_{i,j}^{(u)} \end{aligned} \tag{B2}$$

where the norm is the matrix Frobenius norm. The box constraints defined by $a_{i,j}^{(l)}$, $a_{i,j}^{(u)}$, $p_{i,j}^{(l)}$, and $p_{i,j}^{(u)}$ are included to ensure that the elements of *A* and *P* will remain within the domain of biologically sensible values. (B2) defines a bi-convex optimization problem over the manifold of matrices with connectivity pattern $Z_0$, i.e. given either the matrix *A* or the matrix *P*, there is always a unique least-squares solution which allows one to identify the other matrix, and satisfies all the constraints.

This result permits us to solve the problem using an iterative optimization algorithm, where the matrices *A* and *P* are updated in two different stages. The main steps of the algorithm are as follows:

1. **Initialization**. Initialize $A_0$ to be a matrix with connectivity pattern $Z_0$. In our implementation we set all the non-zero entries of *A* to an arbitrary non-zero number.

2. ***P* update**. Given $A_{k-1}$ compute a new estimate $P_k$ by solving the following least square problem:

$$\min_{P_k} \| E - A_{k-1}P_k \|^2 \qquad \text{(B3)}$$
$$\text{s.t. } p_{i,j}^{(l)} \le p_{i,j} \le p_{i,j}^{(u)}$$

In order to pose the optimization problem defined by (B3) as standard least-squares estimation, we can write the matrices *E* and *P* as follows:

$$E = [e_{c,1} \quad e_{c,2} \quad \cdots \quad e_{c,M}] \qquad \text{(B4)}$$

$$P_k = [p_{c,1}^{(k)} \quad p_{c,2}^{(k)} \quad \cdots \quad p_{c,M}^{(k)}] \qquad \text{(B5)}$$

where $e_{c,i}$ is the *i*th column of *E* and $p_{c,i}^{(k)}$ is the *i*th column of $P_k$. Now define the following column vectors (size (*NM* x 1) and (*LM* x 1), respectively), which are obtained stacking the columns of *E* and *P*, respectively:

$$\mathbf{e_c} = \begin{bmatrix} e_{c,1} \\ e_{c,2} \\ \vdots \\ e_{c,M} \end{bmatrix} \qquad \mathbf{p_c^{(k)}} = \begin{bmatrix} p_{c,1}^{(k)} \\ p_{c,2}^{(k)} \\ \vdots \\ p_{c,M}^{(k)} \end{bmatrix} \qquad \text{(B6)}$$

Define also the following (*NM* x *LM*) block diagonal matrix:

$$\mathbf{A_{k-1}} = \begin{bmatrix} A_{k-1} & & \cdots & 0 \\ \vdots & A_{k-1} & & \\ & & \ddots & \vdots \\ 0 & & \cdots & A_{k-1} \end{bmatrix}$$

Hence, the optimization problem defined by (B3) can be written in canonical forı (B7)

follows:

$$\begin{aligned} &\min_{\mathbf{p}_{\mathbf{c}}^{(\mathbf{k})}} \|\mathbf{e}_{\mathbf{c}} - \mathbf{A}_{\mathbf{k-1}}\mathbf{p}_{\mathbf{c}}^{(\mathbf{k})}\|^2 \\ &\text{s.t. } p_{i,j}^{(l)} \le p_{i,j} \le p_{i,j}^{(u)} \end{aligned} \tag{B8}$$

This sparse constrained least squares problem can be solved using a standard convex optimization technique. In our implementation, we used the SBLS algorithm developed by Björck[*], which is based on the interior point method [1].

3. ***A* update**. Given $P_k$ compute a new estimate $A_k$ by solving the following least squares problem:

$$\begin{aligned} &\min_{A_k} \|E - A_k P_k\|^2 \\ &\text{s.t. } A_k \in \mathcal{A}(Z_0) \\ &\qquad a_{i,j}^{(l)} \le a_{i,j} \le a_{i,j}^{(u)} \end{aligned} \tag{B9}$$

The optimal $A_k$, satisfying the connectivity pattern constraints, can be obtained by observing that, given $P_k$, this problem can be decomposed, in a set of $N$ de-coupled estimation problems, where $L$ is equal to the number of columns of $A$. If we write:

$$E = \begin{bmatrix} e_{r,1} \\ e_{r,2} \\ \vdots \\ e_{r,N} \end{bmatrix} \qquad A_k = \begin{bmatrix} a_{r,1}^{(k)} \\ a_{r,2}^{(k)} \\ \vdots \\ a_{r,N}^{(k)} \end{bmatrix}, \tag{B10}$$

where $e_{r,i}$ is the $i$th row of $E$ and $a_{r,i}^{(k)}$ is the $i$th row of $A_k$, then (B9) is equivalent to the following set of least squares problems:

$$\begin{aligned} &\min_{a_{r,i}^{(k)}} \|e_{r,i} - a_{r,i}^{(k)} P_k\|^2 \qquad i = 1, \dots, N \\ &\text{s.t. } A_k \in \mathcal{A}(Z_0) \\ &\qquad a_{i,j}^{(l)} \le a_{i,j} \le a_{i,j}^{(u)} \end{aligned} \tag{B11}$$

The connectivity constraints on $A_k$ can be removed simply by eliminating from each $a_{r,i}^{(k)}$ those elements that are constrained to be identically zero, resulting in a new set of row vectors $a_{r,i}^{(k)}$. Thus (D11) becomes:

~

$$\min_{\tilde{a}^{(k)}_{r,i}} \| e_{r,i} - \tilde{a}^{(k)}_{r,i} \tilde{P}_k \|^2 \qquad i = 1, \dots, N \qquad \text{(B12)}$$

$$\text{s.t. } a^{(l)}_{i,j} \le a_{i,j} \le a^{(u)}_{i,j}$$

where $\tilde{P}_k$ is obtained from $P_k$ by removing the rows corresponding to the identically zero entries of $a_{r,i}$. This set of constrained least squares problems can be solved using the same optimization procedure used to solve (B8).

4. **Convergence criterion.** If the decrease in total least-square error, at the end of Step 3, is above a predetermined value, repeat from Step 2. The convergence threshold can be selected according to the desired degree of accuracy.

Because in each step of the iterative optimization procedure, the estimation error is guaranteed to be non-increasing, convergence to the optimal solution is assured as long as the hypotheses of Theorem 1 are satisfied.

## 1. 12 Appendix C- Bootstrap Confidence Intervals

The iterative two-step least square algorithm that we propose allows us to obtain estimates of the values of TFA and CS, as defined in our model. How good are these estimates? What is their precision? In order to answer these questions, we used a bootstrap procedure. Our choice was dictated by two observations: (i) we do not want to make any specific distributional assumptions on the errors in our model; (ii) given the iterative nature of the algorithm, we do not have a close form expression that links the estimated values to the observations. The bootstrap is a very general statistical procedure that allows one to learn about sampling variation using the one set of observation at hand[1]. By creating a pool of bootstrap datasets, obtained resampling with replacement from the actual data set, and evaluating the variability of our estimates across these bootstrap datasets, we can learn about the precision of our estimate. While this conveys the general idea of how to gather information on sample variability by pulling ourselves up on the bootstraps of our current sample, there are a variety of implementations of the bootstrap. Because of the constraints that our model has on both the P and A parameters, we implemented what is known as parametric bootstrap: we used our dataset to gather estimates $\overline{P}$ and $\overline{A}$ for P and A, respectively. This automatically leads to estimates of the errors in the gene expression values:

$$\overline{Err} = E - \overline{A}\overline{P} \qquad \text{(C1)}$$

We have then created a pool of bootstrap datasets E* by holding our estimates $\overline{P}$ and $\overline{A}$ as true and resampling from the estimated errors. So that

(C2)

$$Err_i^* = Resample_i(\overline{Err})$$

and

$$E_i^* = \overline{A}\overline{P} + Err_i^* \tag{C3}$$

Each of these bootstrap datasets $E_i^*$ is input in the two-step iterative procedure and leads to estimated values for $\overline{A_i^*}$ and $\overline{P_i^*}$. We carried out 200 of such bootstrap iterations. Using the quantiles of $\overline{P^*}$ and $\overline{A^*}$ in the bootstrap samples we were able to obtain 95% confidence intervals for all the parameter values.

## 1.13 Appendix D: Material and Methods

***Hemoglobin preparation***. Bovine blood was collected in a heparinized (10 units/ml) tube. The plasma and buffer coat were removed after centrifugation at 800 × *g* for 10 min. The cells were resuspended and washed three times in a buffer containing 40 mM Hepes/120 mM NaCl/5 mM glucose at pH 7.4, 282 milliosmolar (mOsm). After each wash, the cells were centrifuged at 800 × *g* for 10 min. RBCs were purified by filtration through a mixture of $\alpha$-cellulose and microcrystalline cellulose. The aliquot was frozen at -80$^o$ C overnight, then centrifuged at 22,000 × g, 4°C for 12 min to get the cell lysate. Oxygenated hemoglobin (oxyHb) was isolated by passing the cell lysate through a Sephadex G-25 fine column. To prepare methemoblobin (MetHb), $K_3Fe(CN)_6$ 0.2M was added to oxyHb isolated above at 2-fold excess. MetHb was purified by passing the solution through a Sephadex G-25 fine column. To prepare cyano-methemoglobin (CyanoHb), $K_3Fe(CN)_6$ 0.2M and KCN 0.1M were added to purified oxyHb at 2-fold excess. The extra chemical components were removed by passing the solution through a Sephadex G-25 fine solution. Mixtures of Hbs at different concentrations were generated by mixing pure solutions, and buffer was added to obtain the desired final concentration.

***Spectrophotometric measurements:*** The absorbance spectra of various Hb solutions were measured using a UV/Vis spectrophotometer (Beckman DU640) at wavelengths from 380 to 700 nm. Spectra data were collected for a wavelength increment of 1 nm.

## 1.14. Appendix E- Transcriptional regulation model

Transcription factors regulate promoter activity through binding to the promoter region. One can approximate the promoter activity using the Hill equation

$$V_{promoter,i}(t) = \frac{\lambda_i \prod_{j=1}^{L} \left( \frac{TFA_j(t)}{k_{i,j}} \right)^{h_{i,j}}}{1 + \prod_{j=1}^{L} \left( \frac{TFA_j(t)}{k_{i,j}} \right)^{h_{i,j}}} \qquad \text{(E1)}$$

The messenger RNA, mRNA level in the cell is a balance between rate of mRNA synthesis (promoter activity) and the rate of mRNA degradation, which is assumed to follow the first-order kinetics

$$V_{degradation,i}(t) = k_{d,i} mRNA_i(t) \qquad \text{(E2)}$$

Thus,

$$\begin{aligned} \frac{d(mRNA_i(t))}{dt} &= V_{promoter,i}(t) - V_{degradation,i}(t) \\ &= \frac{\lambda_i \prod_{j=1}^{L} \left( \frac{TFA_j(t)}{k_{i,j}} \right)^{h_{i,j}}}{1 + \prod_{j=1}^{L} \left( \frac{TFA_j(t)}{k_{i,j}} \right)^{h_{i,j}}} - k_{degradation,i} mRNA_i(t) \end{aligned} \qquad \text{(E3)}$$

On time scales greater than 10 minutes[4], the mRNA levels reach a quasi-steady state, and the above equation can be set to zero. Thus, we have

$$mRNA_i(t) = \frac{\frac{\lambda_i}{k_{degradation,i}} \prod_{j=1}^{L} \left( \frac{TFA_j(t)}{k_{i,j}} \right)^{h_{i,j}}}{1 + \prod_{j=1}^{L} \left( \frac{TFA_j(t)}{k_{i,j}} \right)^{h_{i,j}}} \qquad \text{(E4)}$$

Microarray data are commonly expressed in terms of expression ratios, which are

$$\frac{mRNA_i(t)}{mRNA_i(0)} = \prod_{j=1}^{L} \left( \frac{TFA_j(t)}{TFA_j(0)} \right)^{h_{i,j}} \frac{1 + \prod_{j=1}^{L} \left( \frac{TFA_j(0)}{k_{i,j}} \right)^{h_{i,j}}}{1 + \prod_{j=1}^{L} \left( \frac{TFA_j(t)}{k_{i,j}} \right)^{h_{i,j}}} \qquad \text{(E5)}$$

Taking the logarithm of the above equation, we obtain

$$\log_{10}\left(Er_i(t)\right)=\sum_{j=1}^{L}h_{i,j}\log_{10}\left(TFAr_j(t)\right)+\log_{10}\left(\frac{1+\prod_{j=1}^{L}\left(\frac{TFA_j(0)}{k_{i,j}}\right)^{h_{i,j}}}{1+\prod_{j=1}^{L}\left(\frac{TFA_j(t)}{k_{i,j}}\right)^{h_{i,j}}}\right) \quad \text{(E6)}$$

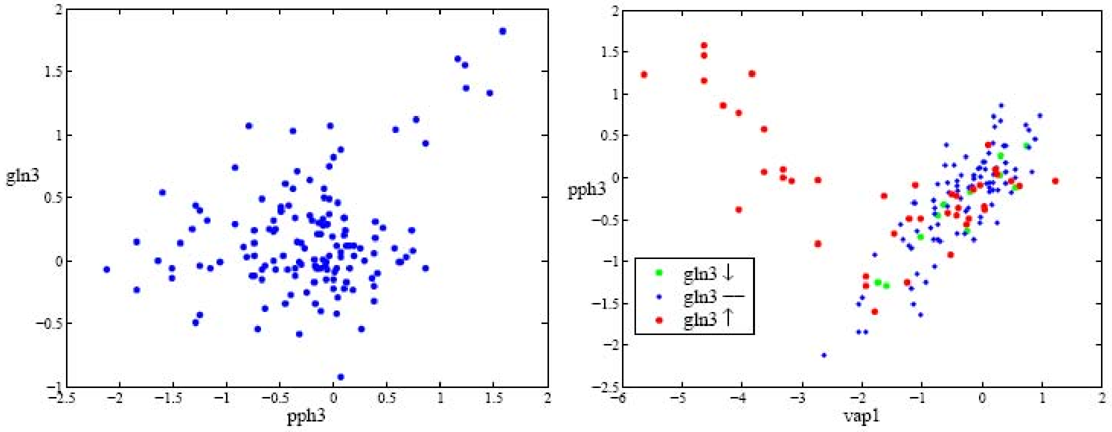
where $Er_i$ is the expression ratio of $mRNA_i$ and TFAr is the ratio of TFA. When $TFA_j(t)$ is in the neighborhood of $TFA_j(0)$, the second term on the right-hand side can be neglected, achieving linearization in the logarithmic space. Therefore, equation (8) in the main text can be regarded as a log-linearized form of the Hill equation.

# Chapter 2: GeneScreen: An Information Theoretic Exploratory Method for Learning Patterns of Conditional Gene Co-expression from Microarray Data

## 2.1 Introduction

DNA microarray technology has revolutionized the field of life science with the introduction of an experimental technique that allows the simultaneous monitoring of the expression levels of how different environmental conditions (including those that are drug induced) affect the regulatory program undertaken by the cell. It is important to recall that gene expression levels quantified in DNA microarray assays are estimates of the relative concentrations of the corresponding *m*RNA molecules. Thus, the expression levels constitute snapshots of the transcription process, which is a key but only one component of the overall complex mechanisms that determine the functioning and the physiological state of the cell. The outputs of the post-transcriptional and post-translational processes remain hidden. Still, one might hope that the analysis of a large amount of expression profiles would result in the capability of indirectly observing the results of such complex interactions.

Historically, biologists have favored simpler analysis tools, which are widely accepted mainly because of their straightforward biological interpretation: an example is given by correlation analysis and its extension to gene clustering by hierarchical agglomeration [1]. Such approaches are based on the simple concept that genes that show similar expression profiles are likely to be co-regulated or, in general, functionally related. On the other hand, because of the complex dynamics involved in cellular processes, gene expression levels are often characterized by nonlinear dependencies, which are not adequately described by a linear model. More importantly, a substantial percentage of interactions between the products of gene transcription occur post translation and therefore they are reflected only indirectly in the expression level of the corresponding gene species. For example, in a phenomenon common to many intracellular regulatory mechanisms, the expression level of a gene might show no substantial variations, while the activity levels of the corresponding protein (generated via the translation process) might vary considerably (based on its interactions with other proteins and/or signaling molecules). This results in variations in the expression levels of other genes, that the activated form regulates [2] [3]. If one only looked at pair-wise correlations of the gene expression profiles, then clearly there will be no correlation between the regulator gene and the regulated gene.



(a)

(b)

Figure 2.1. An example of functional relationship between two genes that is not discernible from their mutual expression patterns. (a) The protein phosphatase *pph3* is responsible for the activation of the transcriptional regulator *gln3*. The scatter plot of the expression profiles does not show particular evidence of a functional relationship between the two genes. This is essentially due to the fact that *pph3* exercises its control on the protein translated from *gln3* only post-transcriptionally. Therefore, the result of this co-activation mechanism is not reflected by the expression levels of *gln3*. (b) When the level of up- or down-regulation of *gln3* is used as an indicator variable, two separate patterns of co-regulation between the genes *vap1* and *pph3* are observed. The mechanism of interaction between *pph3* and *gln3* is elucidated by observing the secondary effects exercised on *vap1*.

Consider the following example. Figure 2.1(a) shows the scatter plot of the expression levels of two genes of the yeast *Saccharomices Cerevisiae*, collected during several whole-genome microarray experiments (*cfr.* Section IV). *gln3* is a transcription factor that is generally inactive unless activated by the protein phosphatase *pph3*, post-translationally: although the two genes are functionally related, this is not revealed by their mutual expression patterns. On the other hand, their functional relationship can be observed when considering their interaction with a third gene, in this case *vap1* (Figure 2.1(b)). The plot shows that *vap1* and *pph3* follow a pattern of co-regulation, when *gln3* is under-expressed or near the reference level. A negative pattern of correlation between the same two genes appears when *gln3* is up-regulated. Such opposite patterns of co-expression can be linked to significantly different, yet rather common, environmental conditions (in this case related to nitrogen abundance or deprivation). This simple case (further analyzed in Section IV), is a clear example of the fact that several functional associations between genes are due to the activities and interactions among their respective proteins, and that these dependencies are not reflected in pair-wise dependencies. However, when conditional co-expressions are observed, then some of these hidden functional associations might be unveiled.

Several attempts aiming at adapting well-established statistical learning frameworks to gene expression data, such as Bayesian Networks [4], Support Vector Machines (SVM) [5], K-means clustering, or Self-Organizing Maps (SOM) [6], have led to promising results, which are paving the road to the development of even more specialized machine learning frameworks, especially suited for biological data [7]. Many of these approaches aim at overcoming the limitations of first-order, linear methods by explicitly modeling non-linear or secondary interactions between genes.

When attempting to model large-scale systems like the one under consideration, one must pay close attention to the actual amount of information that the data can provide. Highly detailed models containing a large number of parameters require massive amounts of data in order to be estimated with a significant degree of accuracy. On the other hand, the amount of information carried by microarray data is inherently limited by at least two factors: first, the noise component can be significantly large; second, the sampling characteristics of time-course experiments are generally poor and characterized by frequent missing values. One of the current challenges is therefore that of determining what kind of high-order interactions can be systematically discovered by a data-driven learning approach, without exceeding the statistical limitations that are inherent to the available measurement data.

The proposed gene expression learning framework is based on the concept of *co-information*, a measure of statistical conditional dependence that is *non-parametric* and it is not restricted to linear models (Section II). The basic idea consists of estimating how the information content shared by a set of $\mathsf{M}$ nodes in the network (where each node is associated to an expression profile) varies upon conditioning on a set of $\mathsf{L}$ conditioning variables. In the simplest case the conditioning nodes are also represented by a separate set of expression profiles; however, the framework can be extended to include different types of descriptors, such as environmental conditions, or experimental settings. The algorithm is implemented as a combinatorial search method where for all possible selections of $\mathsf{M}$ expression profiles, the co-information score is evaluated as a function of the set of $\mathsf{L}$ conditioning variables (Section III). The combinations of $\mathsf{L}$ indicator variables and $\mathsf{M}$ conditioned nodes yielding the largest values of the objective function are stored in order to be further evaluated. In a second stage, the statistical significance of the co-information content of such clusters of nodes is estimated, thus allowing one to retain only those combinations that achieve a minimum *p*-value (details in Section IV). In order to limit the computational cost associated with the combinatorial approach, an efficient moment based approximation of the co-information measure is derived that overcomes the problem of estimating high-dimensional multi-variate probability density functions from the data (the details are provided in Appendix A).

The existence of patterns of co-dependency in gene expression data that can be systematically extracted by an unsupervised machine learning technique has not been clearly established; the results we obtained by analyzing a whole genome microarray assay of *Saccharomyces cerevisiae* [8] (described in Section IV) provide compelling evidence that conditional patterns of interactions between gene expression profiles appear frequently and they are characterized by a high level of statistical significance. Several of the discovered secondary patterns of co-regulation can be effectively associated to known (or partially known) biological mechanisms. On the other hand, not all significant patterns detected by the algorithm carry a straightforward biological interpretation. This is most likely due to the fact that such patterns of conditional co-expression are often mediated by additional hidden factors that are not directly measurable. Nonetheless, the proposed framework provides a valuable tool to biologists, being capable of highlighting interactions among genes whose biological significance can be elucidated through further experimental analysis.

A software implementation of our exploratory method (dubbed *GeneScreen*) has been incorporated in *BioSpice*[1] since its very first release. BioSpice is an integrated suite including a large collection of computational biology and modeling tools, developed within DARPA's BioComp program.

## 2.2 Conditional Co-expression Model

The idea behind the proposed approach consists of identifying groups of genes that are co-expressed *only conditionally on the expression level of other genes*. The problem can be broken down as follows: first, how do we choose a measure of statistical dependence that is capable of detecting conditional co-expressed genes; second, how can one identify a set of genes whose expression levels are indicator functions of a significant change in the transcriptional regulation mechanisms affecting a specific set of genes.

In order to detect patterns of conditional co-expression in the data, one must choose a measure of statistical dependence. Although conditional correlation is probably the simplest such measure, it is only suitable for detecting patterns of linear dependency [7]. A natural measure of statistical dependence that does not make any assumptions on the linearity of the model is given by the mutual information [9].

### *A. Conditional Mutual Information as a Measure of Conditional Co-expression*

Let us start by considering the definition of mutual information between two random variables $x_1$ and $x_2$ conditioned on a third random variable $y$:

$$I(x_1;x_2|y) \triangleq D\left(p(x_1,x_2|y)\,\middle\|\,p(x_1|y)p(x_2|y)\right), \qquad (1)$$

where $D$ is the Kullback-Leibler distance or relative entropy [9], defined as:

$$D(q||r) \triangleq E_q\left[\log\frac{q(u)}{r(u)}\right]. \qquad (2)$$

It can be shown [9] that the relative entropy is always non-negative and is zero if and only if $q = r$ almost everywhere. In the case of continuous random variables (1) can be expressed as:

$$I(x_1; x_2|y) = E_{p(x_1,x_2,y)}\left[\log\left(\frac{p(x_1, x_2|y)}{p(x_1|y)p(x_2|y)}\right)\right] \tag{3}$$

$$= \int_{-\infty}^{\infty} p(x_1, x_2, y)\log\left(\frac{p(x_1, x_2|y)}{p(x_1|y)p(x_2|y)}\right) dx_1 dx_2 dy.$$

This definition can be extended to the mutual information of $M$ random variables $\mathbf{x} = [x_1, \ldots, x_M]^T$ ,conditioned on a separate set of $L$ variables $\mathbf{y} = [y_1, \ldots, y_L]^T$ as follows:

$$\begin{aligned} I(\mathbf{x}|\mathbf{y}) &= E_{p(\mathbf{x},\mathbf{y})}\left[\log\frac{p(\mathbf{x}|\mathbf{y})}{\prod_{i=1}^{M} p(x_i|\mathbf{y})}\right] && (4) \\ &= \int_{-\infty}^{\infty} p(\mathbf{x}, \mathbf{y})\log\frac{p(\mathbf{x}|\mathbf{y})}{\prod_{i=1}^{M} p(x_i|\mathbf{y})} d\mathbf{x}d\mathbf{y}, && (5) \end{aligned}$$

This expression provides us with a measure of the expected mutual information of $\mathbf{x}$ conditionally on the value of $\mathbf{y}$. Evidently, when $\mathbf{x}$ and $\mathbf{y}$ are statistically independent, we have trivially that:

$$I(\mathbf{x}|\mathbf{y}) = I(\mathbf{x})\int_{-\infty}^{\infty} p(\mathbf{y})d\mathbf{y} = I(\mathbf{x}). \tag{6}$$

Recalling that we are after certain structure in the data that appears only under conditioning, this result prompts us with the idea of adopting the following cost function:

$$\boxed{\mathcal{L}(\mathbf{x}|\mathbf{y}) \triangleq I(\mathbf{x}|\mathbf{y}) - I(\mathbf{x})} \tag{7}$$

Clearly, we have that $L(x|y) = 0$ when $x$ and $y$ are independent. In this case, even if a cluster of genes possesses high information content, *i.e.* $I(x)$ is large, such structure appears regardless of the set of conditioning variables. On the other hand, $L(x|y)$ is a large positive number when the information content is significantly increased under conditioning. This is the case of interest in our framework. Notice that the quantity in (7) might assume negative values and it is not lower-bounded in general.

$$
\begin{aligned}
I(x_1;x_2|x_0) &= E_{p(x_0,x_1,x_2)}\left[\log\frac{p(x_1,x_2|x_0)}{p(x_1|x_0)p(x_2|x_0)}\right] && (9)\\
&= \int_{-\infty}^{\infty} p(x_0,x_1,x_2)\log\frac{p(x_1,x_2|x_0)p(x_0)^2}{p(x_1|x_0)p(x_2|x_0)p(x_0)^2}\,dx_0dx_1dx_2\\
&= \int_{-\infty}^{\infty} p(x_0,x_1,x_2)\log\frac{p(x_0,x_1,x_2)p(x_0)}{p(x_0,x_1)p(x_0,x_2)}\,dx_0dx_1dx_2\\
&= -H(x_0,x_1,x_2)-H(x_0)+H(x_0,x_1)+H(x_0,x_2), && (10)
\end{aligned}
$$

where:

$$H(x) = -\int p(x)\log p(x)dx, \qquad (11)$$

is the differential entropy of the random variable $x$. Therefore, recalling that

$$
\begin{aligned}
\mathcal{L}(x_1;x_2|x_0) = -H(x_0) - H(x_1) - H(x_2) + H(x_0,x_1)\\
+H(x_0,x_2) + H(x_1,x_2) - H(x_0,x_1,x_2). \qquad (12)
\end{aligned}
$$

From this expression we conclude that when considering a simple network with one conditioning node and two children nodes, the cost function (7) is indeed equal to the negative *co-information* between the three random variables. The general definition of co-information of $N$ random variables can be found in [10]:

$$\mathcal{C}(\mathbf{x}) = \sum_{E_j\subseteq E_N} q_j H(\mathbf{x}_{E_j}), \qquad (13)$$

where $E_j$ is the power set of $j$ and $q_j$ is the M¨obius inversion function, defined as:

$$q_j = -(-1)^{|E_j|} = \begin{cases} 1 & \text{if } |E_j| \text{ is odd} \\ -1 & \text{if } |E_j| \text{ is even} \end{cases}, \qquad (14)$$

and $|E_j|$ is the cardinality of $E_j$. The co-information provides a measure of the total information content shared by all the random variables, unlike the conventional mutual information which includes all the information shared by the variables two at a time.

Therefore, the maximization of (7) is equivalent to seeking clusters whose nodes yield the largest (in absolute value) negative co-information. Equivalently:

$$\max_{x_0,x_1,x_2} \mathcal{L}(x_1;x_2|x_0) = \min_{x_0,x_1,x_2} \mathcal{C}(x_0,x_1,x_2). \qquad (15)$$

Notice that expression (12) is not altered if we exchange the variables $x_0$, $x_1$, or $x_2$. Hence, it holds that:

$$\mathcal{L}(x_1;x_2|x_0) = \mathcal{L}(x_0;x_2|x_1) = \mathcal{L}(x_0;x_1|x_2) \qquad (16)$$

Thus, the information content of the sub-network does not change if we exchange one of the children nodes with the parent node.

## 2.3. METHOD

When designing a practical implementation of the algorithm seeking clusters that maximize the cost function (7), certain issues must be taken into account:

- A direct evaluation of the cost function (7) requires an estimate of the multi-variate probability density function (pdf) of the variables included in the cluster. Although for small dimensional problems methods for estimating directly the joint pdf have been developed [11], for higher dimensional problems the direct estimation of the pdf is usually not feasible.

- The noise level in the data might significantly limit the number of parameters that can be learned with a certain degree of accuracy.

- Current microarray experiments are affected by inherent limits in the number of expression samples that can be measured in a given interval of time. Hence, the sampling characteristics of any microarray assay are generally poor in the time-domain. This issue imposes a further limitation on the capability of estimating joint probability density functions.

- For a sub-network of a given size, finding high values of the cost function (7) requires a search through all possible combinations of nodes chosen among the set of genes included in the experiment. We will show that such a number of combinations can be quite large if the parameters of the search algorithm are not chosen properly, quickly yielding an intractable computational cost.

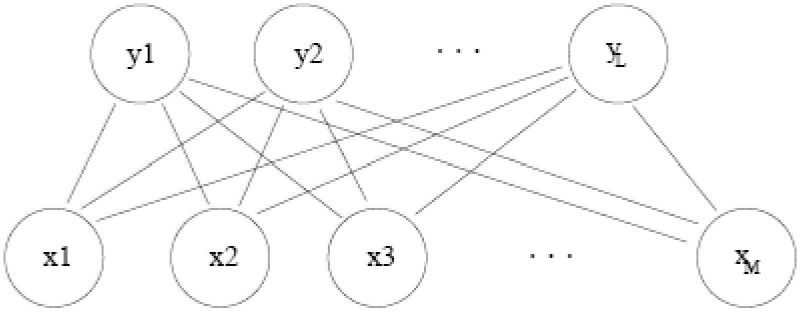


Figure 2.2. Cluster of genes composed of L conditioning genes and M children nodes. This cluster represents the generic sub-network explored to identify conditional structure.

*A. Combinatorial Search Approach*

The goal is to identify a list of sub-networks that yield large values of the cost function (7). Such a goal can be achieved by exhaustively selecting sub-networks (see Figure 2.2) consisting of all possible combinations of L genes as conditioning variables (which we will refer to as parent nodes), and all possible combinations of M genes among the remaining ones as conditioned variables (also known as children nodes), and by evaluating the corresponding value of the cost function (7). The cost of this combinatorial approach increases quite rapidly with the total number of genes assayed in the experiment, and it is a non-linear function of M and L. When a total number of N gene expression profiles are measured in the experiment, the total number of possible sub-networks with L parent nodes and M children nodes is given by the following expression:

$$\mathcal{K}(N,M,L) = \begin{pmatrix} N \\ L \end{pmatrix} \begin{pmatrix} N-L \\ M \end{pmatrix} \quad (17)$$

$$= \frac{N!}{M!L!(N-L-M)!}. \quad (18)$$

F

or example, when dealing with N = 2, 000 genes, a choice of L = 3 and M = 5 will result in $3.5 \cdot 10^{23}$ possible combinations! In general, for small values of M and L, we have that:

$$\mathcal{K}(N,M,L) \approx \mathcal{O}(N^{M+L}). \quad (19)$$

Hence, unless a technique is devised that allows for efficient pruning of non-informative clusters, the problem will be computationally tractable only for very small values of $M$ and $L$. In addition, as it will be discussed in more detail in the next section, for large values of $M$ and $L$ we will unavoidably incur the problem of having to estimate high-dimensional multivariate statistics of the data, thus requiring a significantly large number of samples in order to get a robust estimate. These constraints clearly suggest that a simple framework in which a sub-network involving only three genes (one parent node and two children nodes) should be the subject of an initial investigation and validation of the proposed approach. From the symmetric expression of the cost function given in (12), it is possible to show that the computational complexity associated with evaluating the co-information content of each possible sub-network, when $L = 1$ and $M = 2$ simplifies as:

$$\begin{aligned} \mathcal{K}(N,2,1) &= N(N-1)(N-2) + N(N-1) + N \\ &= N^3 - 2N^2 + N \qquad (20) \\ &= \mathcal{O}(N^3), \qquad (21) \end{aligned}$$

for a total number of $N$ genes assayed. As an example, when $N = 2,000$, approximately $8 \cdot 10^9$ possible combinations need to be considered, and the corresponding cost function evaluated. This kind of task can be completed in a reasonable amount of time by any modern off-the-shelf single-processor machine. It is also clear that the algorithm could be easily parallelized to run on clusters of processors, since the evaluation of the cost function for a given sub-network is an independent task.

### *B. Evaluating the Co-information*

The expression of the cost function given in (12) suggests that some kind of estimate of the multivariate joint probability density function of the three variables in the cluster is required in order to evaluate the corresponding entropies. However, considering that the typical experimental setting in DNA microarray assays results in a limited number of samples per gene, such poor sampling properties generally discourage the use of standard probability density function estimators such as parametric models or kernel methods. Therefore, in the design of a practical implementation of the principle (15), we opted for the use of a moment based approximation of the information theoretical quantities involved in the calculation of the cost function (the derivation of such an approximation is detailed in Appendix A).

## 2.4. RESULTS AND DISCUSSION

For a given microarray experiment all possible unique combinations of three genes are considered and the co-information is evaluated to assign a score to each such combination. The highest scoring clusters are recorded in order to be further evaluated. The actual software implementation includes a set of tools that are required in order to pre-process the expression data and perform a series of tasks which include pruning the set of genes according to a user defined criterion (*e.g.* their sample variance), correcting for outliers or accounting for missing values.

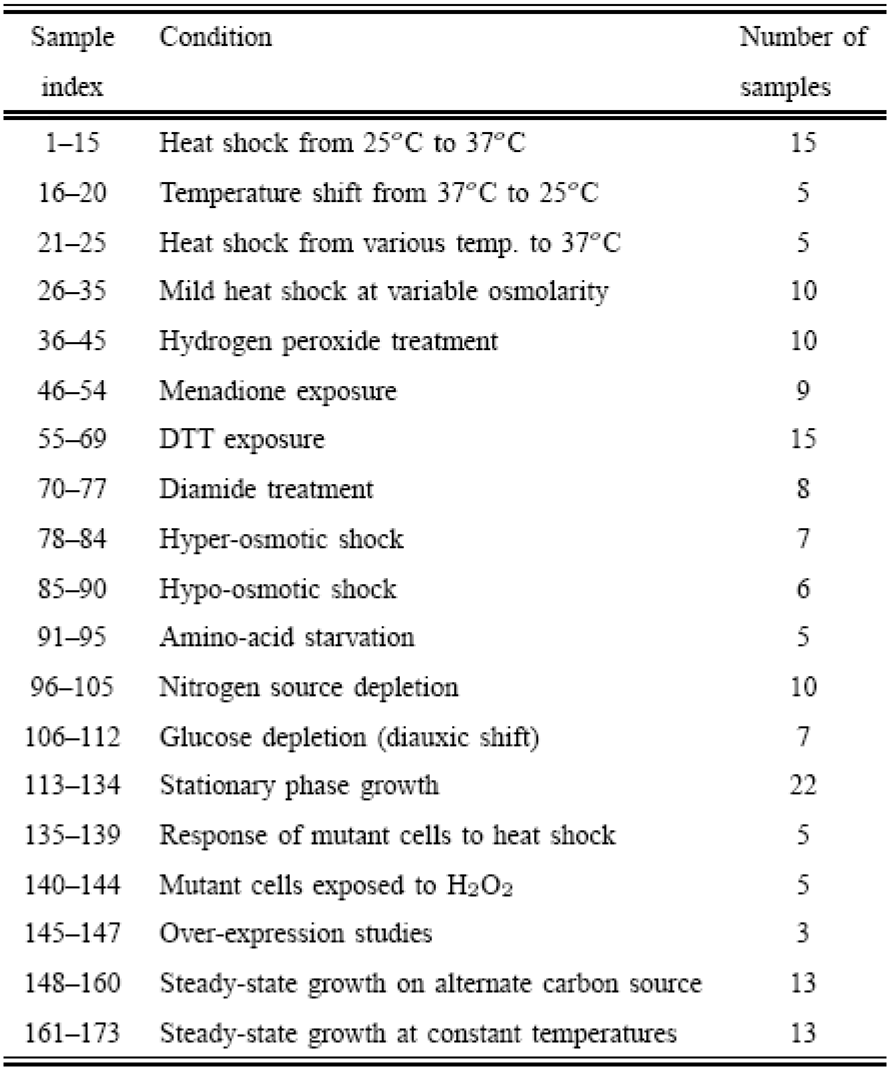
DNA microarray data is conventionally expressed as the logarithm (usually in base 10) of the ratio between the estimated expression level and a reference value. Therefore, a log-ratio value of zero indicates that the gene is expressed at levels close to the reference. For example, a reading of $0.3$ or above is equivalent to at least a 2-fold increase in the transcription level. Equivalently, when the log-ratio level is $-0.3$ or less, the gene shows at least a 2-fold decrease in the expression level.

Due to the small sample size available, the estimation of the set of conditional entropies is most efficiently accomplished by discretizing the expression levels of the parent node into three levels, according to whether the gene is down-regulated, close to the reference level (baseline), or up-regulated. The choice of the discretization levels is arbitrary and will, in general, affect the outcome of the exploratory analysis. Throughout our analysis, the default thresholds of $-0.3$ for *down-regulation* and $0.3$ for *up-regulation* were adopted. Such choice should not be considered as generally applicable but, rather, should be tailored to each specific dataset. In our case, the significance of a two-fold increase or decrease in the expression level was derived from the statistical analysis of the set of micro-array experiments reported in [8], which were used to validate the proposed exploratory method.

### *A. Analysis of Saccharomyces cerevisiae Expression Data*

In order to evaluate the effectiveness of the proposed approach in unveiling hidden dependencies between gene transcription levels, we considered a dataset composed of several experiments involving whole-genome assays of the gene expression levels of the yeast *S.cerevisiae*. The data[2] consists of a total of 6,152 genes with 173 sample points per gene. Table 2.1 provides a basic listing of the experimental conditions. A detailed description of the actual experimental conditions can be found in [8]. The dataset comprises a variety of experimental conditions, including temperature induced shock, exposure to various chemicals, amino acid starvation, nitrogen depletion and so on. The resulting large oscillations in the expression levels of several genes ensure that the dataset m provides enough variability to allow the consistent detection of specific patterns, in a statistically meaningful way. A preliminary analysis of the dataset suggested the removal of the sample points 113–134, (*cfr.* Table 2.1), which were collected over a considerably larger span of time (few days vs. few hours for the other experiments). We observed that in such experiments several genes were characterized by a different expression pattern, most likely associated with the fact that the yeast cells had reached a steady state and, as a consequence, the growth process had considerably decelerated. Therefore, such points were treated as outliers and were not included in our exploratory procedure.

TABLE 2. 1 EXPERIMENTAL CONDITIONS AND CORRESPONDING NUMBER OF MEASUREMENTS FOR THE *S.cerevisiae* DATASET.



| Sample index | Condition | Number of samples |
|---|---|---|
| 1–15 | Heat shock from 25°C to 37°C | 15 |
| 16–20 | Temperature shift from 37°C to 25°C | 5 |
| 21–25 | Heat shock from various temp. to 37°C | 5 |
| 26–35 | Mild heat shock at variable osmolarity | 10 |
| 36–45 | Hydrogen peroxide treatment | 10 |
| 46–54 | Menadione exposure | 9 |
| 55–69 | DTT exposure | 15 |
| 70–77 | Diamide treatment | 8 |
| 78–84 | Hyper-osmotic shock | 7 |
| 85–90 | Hypo-osmotic shock | 6 |
| 91–95 | Amino-acid starvation | 5 |
| 96–105 | Nitrogen source depletion | 10 |
| 106–112 | Glucose depletion (diauxic shift) | 7 |
| 113–134 | Stationary phase growth | 22 |
| 135–139 | Response of mutant cells to heat shock | 5 |
| 140–144 | Mutant cells exposed to $H_2O_2$ | 5 |
| 145–147 | Over-expression studies | 3 |
| 148–160 | Steady-state growth on alternate carbon source | 13 |
| 161–173 | Steady-state growth at constant temperatures | 13 |

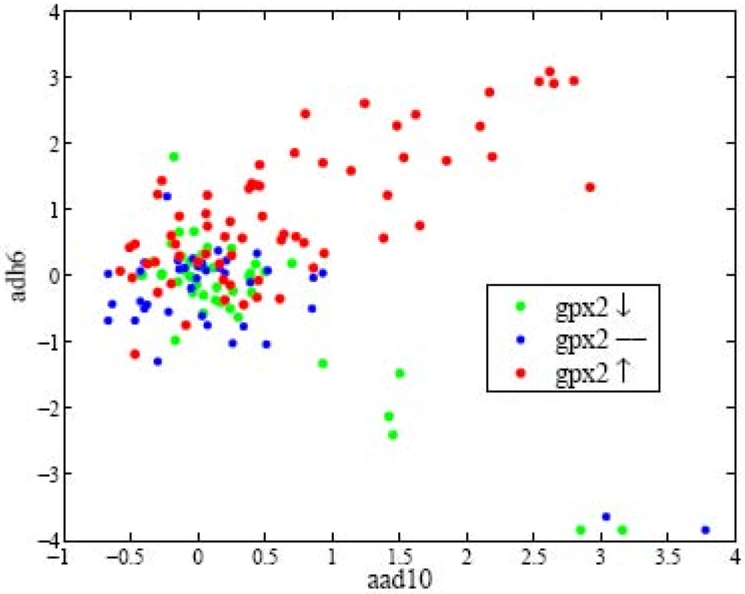


Figure 2.3. Co-expression pattern between the genes *aad10* and *adh6*, when gene *gpx2* is the conditioning node. The plot shows that when *gpx2* is up-regulated, *adh6* and *aad10* are in general positively correlated and above the reference level. On the other hand, when *gpx2* is down-regulated, a negative pattern of correlation appears between *adh6* and *aad10*. Such a conditional expression program could be explained by considering that *gpx2* was found to be considerably under-expressed in those experiments involving a depletion in sources of nitrogen. Therefore, in such conditions the enzyme translated from *adh6* lacks its primary activation mechanism, and its enzymatic role appears to be replaced by *aad10*.

Among all the possible triplets of genes whose score was evaluated, 3,124 resulted in a value of the co-information measure that was above a pre-selected significance threshold. Note that such a threshold simply limits the number of significant clusters that are logged during the procedure, and it does not affect the algorithm for scoring the triplets. Typically, the significance threshold is selected above a pre-defined value simply to limit the size of the output file generated by the procedure.

A selection of the conditional interaction patterns that were identified by the algorithm are described in detail below. For each cluster of genes whose conditional co-expression pattern appeared to be relevant, we evaluated separately the statistical significance of the co-information score by using the following procedure. For each conditioning node, we randomly permuted its sample points several times (at least 100 million permutations), and re-evaluated the score of the selected triplet of genes by using the permuted version of the conditioning variable. A *p*-value expressing the statistical significance of the interaction is obtained by counting the number of times that the score obtained with the scrambled values is larger than the score obtained when using the actual data. Such procedure is analogous to the bootstrapping approach for the computation of the statistical significance described in [12, Ch.16].

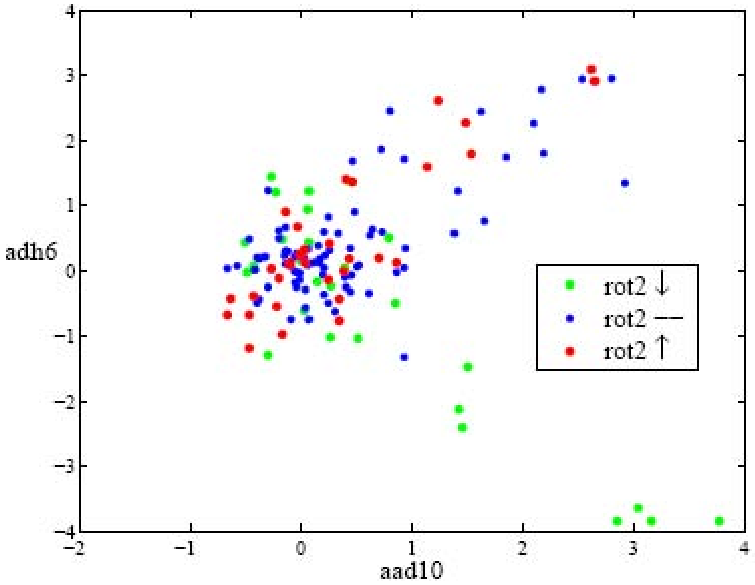


Figure 2.4. Co-expression pattern between the genes *aad10* and *adh6*, when gene *rot2* is the conditioning node. The expression levels of *aad10* and *adh6* switch from a positive to a negative correlation pattern when *rot2* is under-expressed.

The cluster of genes resulting in the highest value of the co-information cost function included *aad10* and *adh6*, which were found to be co-expressed conditionally on the expression levels of the genes *rot2*, *alg7*, and *gpx2* (Figures 2.3, 2.4, and 2.5). *aad10* is a putative alcohol dehydrogenase, *i.e.* an enzyme involved in alcohol degradation. The product of *adh6* is also an alcohol dehydrogenase, whose activity is NADPH dependent. Figure 2.3 shows that when *gpx2* (a glutathione peroxidase induced during glucose starvation) is up-regulated, *adh6* and *aad10* are in general positively correlated and above the reference level. On the other hand, when *gpx2* is down regulated, a negative pattern of correlation appears between *adh6* and *aad10*. Such a conditional expression program could be explained by considering that *gpx2* was found to be considerably under-expressed in those experiments involving a depletion in sources of nitrogen. Therefore, in such conditions the enzyme translated from *adh6* lacks its primary activation mechanism, and its enzymatic role appears to be replaced by *aad10*. Figures 2.4 and 2.55 show that the genes *rot2* (involved in normal cell wall synthesis) and *alg7* (responsible for protein glycosylation) also act as indicators of such alternative gene expression program, with their expression being repressed under the same conditions.

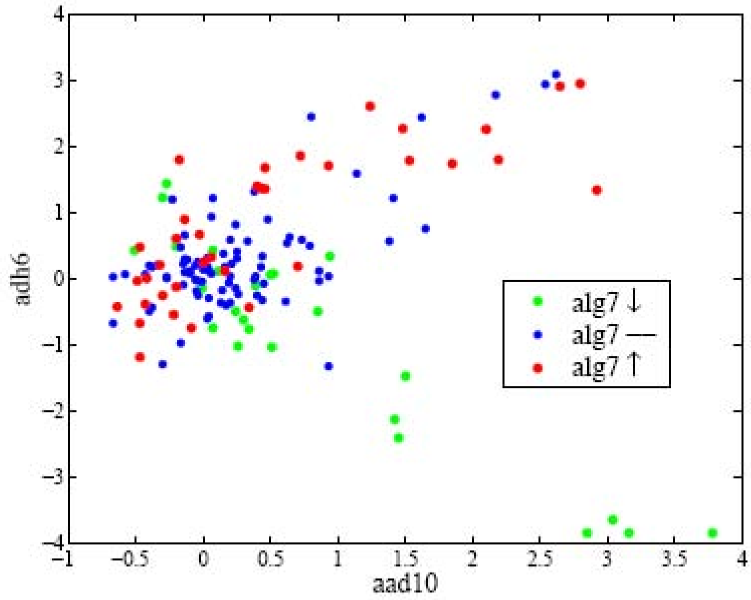


Figure 2.5. Co-expression pattern between the genes *aad10* and *adh6*, when gene *alg7* is the conditioning node. The expression levels of *aad10* and *adh6* switch from a positive to a negative correlation pattern when *alg7* is under-expressed.

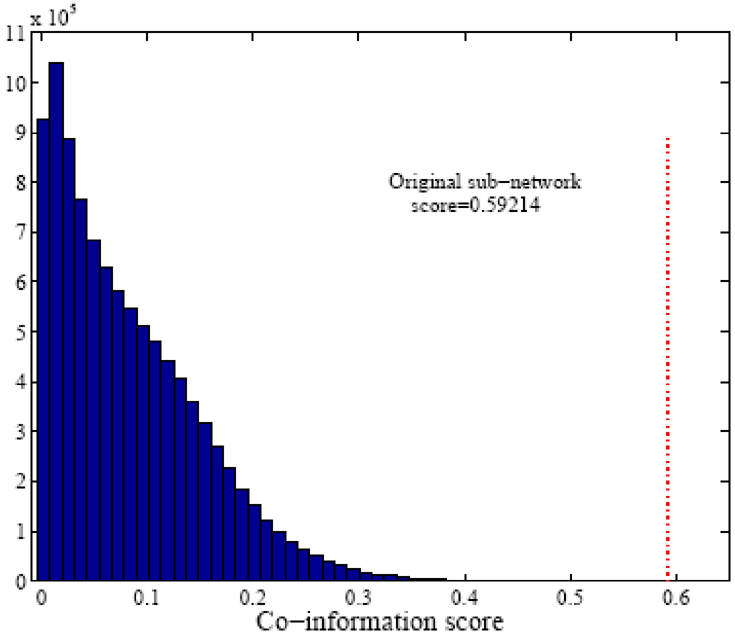


Figure 2.6. Statistical significance of the conditional co-expression pattern. The plot shows a histogram of the co-information values obtained by scoring the triplet *aad10*, *adh6*, and *gpx2*, when the samples of the latter are randomly permuted. The score of the actual sub-network is also shown for reference.

The conditional co-expression pattern involving *gpx2*, *aad10* and *adh6* was found to be significant at a p-value of less than $10^{-8}$. A histogram of the scores obtained by scrambling the values of the conditioning variable is shown in Figure 2.6, where the score obtained by using the actual sample points is also shown for comparison. Figure 2.7, shows the conditional co-expression pattern involving genes *sul1*, *sam4*, and *cwp1*. The gene *sul1* is one of two major mediators (*sul2* is the other one) of the sulfate transport pathway, being responsible for controlling the concentration of endogenous activated sulfate intermediates. Its activity is closely related to the one of *sam4*, the latter being involved in the metabolism of sulfur-containing aminoacids. The gene *cwp1*, is mainly involved in cell wall organization and biogenesis. *sam4* and *cwp1* appear to be in general negatively correlated, possibly due to the fact that their activity peaks in completely different stages of the yeast cell cycle. However, when *sul1* is over-expressed (signaling an increase in the concentration of activated sulfate compounds), the two genes appear to be positively correlated as well as generally under-expressed.

The last example (briefly mentioned in the introduction) involves the genes *gln3*, *vap1*, and *pph3*. This case is of particular interest since it demonstrates the method's capability of detecting certain types of regulatory interactions that could not be directly derived from simple correlation patterns. *gln3* is a transcription factor responsible for the regulation of nitrogen utilization. Its product is generally inactive unless activated by the protein phosphatase translated from *pph3*. Figure 2.1(a) shows a scatter plot of *gln3* vs. *pph3*: no pattern of co-expression appears when examining the expression profiles of these two genes. On the other hand, as it is shown in Figure 2.1(a), these genes are conditionally co-expressed. The plot shows that *vap1* (whose product is an amino-acid transport protein) and *pph3* are positively correlated, when *gln3* is under-expressed or near the reference level. An up-regulation of *gln3* results in the opposite correlation pattern for *vap1* and *pph3*. This mechanism can be explained considering that most of the expression levels responsible for such pattern are relative to conditions of either nitrogen depletion or amino-acid starvation. In such conditions, the expression level of *gln3* rapidly increases (concurrently with the level of its activator *pph3*), in order to repress the transcription of nitrogen demanding gene products. At the same time, *vap1* is strongly down-regulated due to the fact the amino-acid transport mechanisms are significantly slower during this stage.

In general, not all significant patterns of conditional interaction detected by the algorithm carry a straightforward biological interpretation. This is most likely due to the fact that such patterns of conditional co-expression are often mediated by several factors that are not directly measurable. Moreover, it is often the case that conditionally informative clusters include one or more genes whose biological role is only partially known or completely unknown. Despite such limitations, the proposed framework provides a valuable tool to biologists, being capable of highlighting patterns of interaction whose biological significance can be elucidated through further experimental analysis.

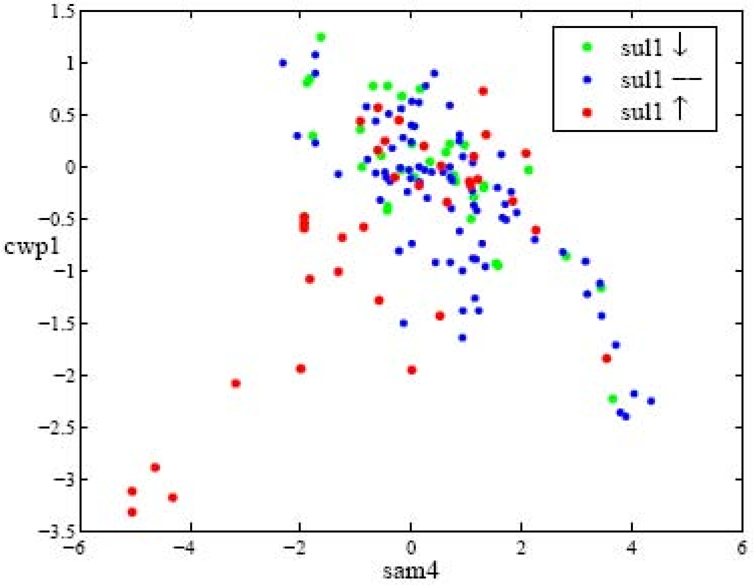


Figure 2.7. Co-expression pattern between the genes *sam4* and *cwp1*, when gene *sul1* is the conditioning node. *sam4* and *cwp1* appear to be in general negatively correlated, possibly due to the fact that their activity peaks in opposite stages of the yeast cell cycle. However, when *sul1* is over-expressed (signaling an increase in the concentration of activated sulfate compounds), the two genes appear to be positively correlated as well as generally under-expressed.

## 2.5 CONCLUSIONS

We introduced a novel method capable of detecting linear as well as non-linear patterns of conditional co-expression in gene expression measurements. Due to the significant computational cost associated with the proposed exploratory method, the derivation of an efficient technique for evaluating the co-information score played a key role in order to make the method computationally tractable. We applied the method to a whole genome micro-array dataset of the yeast *S. cerevisiae* and were able to detect several statistically significant patterns of conditional interaction between genes. This result proves unquestionably that such patterns of conditional co-expression appear indeed very frequently in the data, and raises the very important question of whether a general biological model capable of explaining such interactions can be devised.

### 2.6 References

[1] M.B. Eisen, P.T. Spellman, P.O. Brown, and D. Botstein, *Proc. Natl. Acad. Sci. USA*, 95:14863–14868, December 1998.

[2] K.C. Kao, Y.-L. Yang, R. Boscolo, C. Sabatti, V.P. Roychowdhury, and J.C. Liao, *Proceedings of the National Academy of Sciences (PNAS)*, 101(2):641–646, 2004.

[3] J.C. Liao, R. Boscolo, Y.-L. Yang, L.M. Tran, C. Sabatti, and V.P. Roychowdhury, *Proceedings of the National Academy of Sciences (PNAS)*, 100(26):15522–15527, 2003.

[4] N. Friedman, I. Nachman, and D. Pe´er, *Proc. Fifteenth Conf. on Uncertainty in Artificial Intelligence (UAI)*, pages 206–215, San Francisco, 1999.

[5] T. Hastie, R. Tibshirani, and J.H. Friedman, *The Elements of Statistical Learning*. Springer Verlag, New York, 2001.

[6] T. Kohonen, *Biological Cybernetics*, 43(1):59–69, 1982.

[7] Ker-Chau Li, *Proc. Natl. Acad. Sci. (PNAS) USA*, 99(26):16875–16880, December 2002.

[8] A.P. Gash *et al.*, *Mol. Biol. Cell.*, 11:4241–4257, 2000.

[9] T.M. Cover and J.A. Thomas, *Elements of Information Theory*, John Wiley & Sons, 1991.

[10] A. J. Bell, *Proc. 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 921–926, Nara, Japan, April 2003.

[11] B.W. Silverman, Chapman and Hall, New York, 1985.

[12] B. Efron and R. J. Tibshirani, *An Introduction to the Bootstrap*, Chapman & Hall, 1993.

# Chapter 3: A Software Package for cDNA Microarray Data Normalization and Assessing Confidence Intervals

## 3. 1 Introduction

The complimentary DNA, cDNA, microarray platform is a powerful molecular biology tool that can be used to monitor gene expression at a global level. Over the past few years, millions of data points have been generated using DNA microarray technology [1]. With this avalanche of information comes an increase in the complexity associated with its analysis. Two areas that are greatly impacted by this increase in complexity are: 1) data confidence and 2) data interpretation. This article addresses the first issue.

DNA microarray data are susceptible to many sources of systematic and random errors. Researchers must employ proper analytical techniques to control for systematic errors and assess the extent of random errors. Many of these techniques are rather cumbersome because of their complexity or lack of an intuitive user interface. Here, we will describe a software package, lcDNA (available from http://receptor.seas.ucla.edu/lcDNA), that is designed to assess the degree of confidence of microarray results. This program is specifically designed for use with two-channel arrays which utilize either spotted or *in situ* synthesized probes. lcDNA contains three core analytical components: i) data filtering, which removes outliers, ii) normalization between two channels, and iii) assessment of confidence intervals using a hierarchical Bayesian model to estimate the error distributions. The detailed methodologies have been published previously [2]. lcDNA can generate gene-specific confidence intervals with biologically independent measurements and technical replication; however, to fully exploit lcDNA's capabilities, calibration hybridizations should be incorporated into the experimental design.

Here the term gene is used to indicate a unique cDNA transcript and the term spot indicates the location where a gene is tethered to the array surface. In the case of replicate spotting, a given gene corresponds to multiple spots. The terms slide, array, and microarray are used interchangeably.

## 3.2 Sources of variations

The DNA microarray platform is still some distance from technical perfection. There are a number of factors that can contribute to variation in the data. See [3] for an overview of the sources of error in and analysis of cDNA microarray experiments. The variations can be roughly classified as spot-to-spot, slide-to-slide, and experiment-to-experiment variations [2, 4, 5]. To assess and control for these variations, spot, slide, and experimental replications are necessary.

Spot-to-spot variation can result from differences in the print tips, surface imperfections, surface contamination or damage, and inhomogenous hybridization.

Probe replication for each gene on a slide followed by quality filtering will reduce noise due to spot-to-spot variation. Slide-to-slide variation arises from slide manufacturing inconsistency, which is still a common problem for most spotted arrays. In order to account for this variation, it is necessary to perform technical replicates which are repeat hybridizations of the same labeled RNA mixture to different microarray slides (Fig. 3.1a). Experiment-to-experiment variation is meant to encompass biological variations, as well as procedural variations (error from sample processing). Biological variations arise from fluctuations in the environment, cell history, and intrinsic noise in the cell. Procedural variation is mainly due to labeling, which is a result of differences in the fluorescent properties of the dyes or incorporation efficiency. Normalizing the data reduces the amount of such variation, at the single slide level. Dye swapping (reverse labeling) should be used in replicate experiments to remove the variation arising from differences in dye properties.

### 3.3 Experimental Design

To overcome the error arising from systematic and biologic variation, we recommend that replicate spots, technical replicates, and experimental replicates are included in the experimental design (Fig. 3.1a). Technical replicates are performed by hybridizing the same labeled DNA mixture to multiple slides. The same experiment is then replicated multiple times. In addition, calibration hybridizations (Fig. 3.1b) are used to provide gene-specific prior information about slide-to-slide and culture-to-culture variations. In these experiments, the same RNA pool is divided into two aliquots, which are used to generate Cy3- or Cy5-labeled cDNA pools. The two labeled cDNA pools are mixed and then hybridized to two different slides. Since the two channels come from the same RNA pool, the variation between the slides and experiments can be readily identified. Such calibration experiments should be conducted under representative conditions where most of the genes of interests yield a signal. If no single condition fits this criterion, a mixture of RNA produced under different conditions can be used.

Once the calibration experiments are performed, the information could be used for all experiments subsequently, as long as slide-to-slide variation and experiment-to-experiment variation remain unchanged. Incorporation of calibration hybridizations into the experimental design increases the confidence level and the number of differentially expressed genes that are categorized as statistically significant.

### 3.4 Analytical Functions

The program contains three core analytical functions: 1) data filtering, 2) normalization, and 3) assessment of gene-specific confidence intervals. An overview of the functions is presented below. The data that is entered into lcDNA must be stored in text tab-delimited files that contain the following five columns: channel 1 intensity, channel 2 intensity, gene ID number, gene number, and gene name. Here, we use the tutorial data that are provided with lcDNA to illustrate some of the key features of the program. These data were generated using in-house *E. coli* genome chips printed on Corning GAPS microarray slides. Unless otherwise specified, the samples were labeled with the aminoallyl technique.

### 3.5 Data Filtering

*Eliminate Extremes:* This function rejects all spots that contain an intensity outside of a user specified range. The main purpose of this function is to eliminate the spots that contain values near the upper and lower limits of the measurable intensities. If cDNA microarray images are 16-bit the upper bound should be less than 65535 ($2^{16}$-1). The lower bound is the user-defined background level, typically generated from negative control spots.

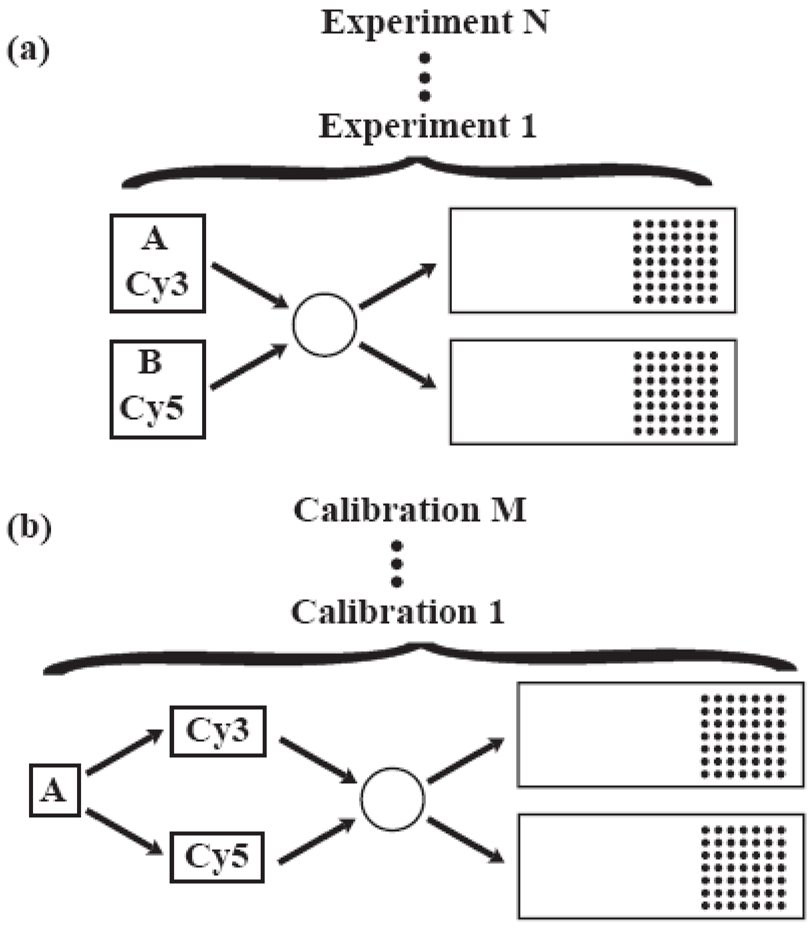


Figure 3.1. Components of the cDNA microarray experimental design supported by lcDNA. A and B are used to denote different sets of RNA, and Cy3 and Cy5 denote the commonly used fluorophores of the same names. (a) Technical replicates are repeat hybridizations of the same labeled RNA mixture to different microarray slides. They are

used to account for slide-to-slide variation. In most cases, two microarray slides per biologically independent measurement provide sufficient information about the variation. (c) Calibration hybridizations are performed by dividing a pool of RNA into two aliquots, labeling each aliquot with a separate fluorophore, mixing the labeled aliquots, and hybridizing the mixture to two or more microarray slides. Calibration hybridizations provide information about the effects of intrinsic and environmental factors on variations in gene expression levels.

1. *Quality Filtering:* A number of non-uniform errors can occur on the microarray surface. Some of the common sources of error include contamination by airborne particles, non-uniform printing, inhomogeneous hybridization, and surface damage. One method that is used to overcome these errors is printing multiple copies of each gene on the array and then comparing the measurements from the replicate spots. Because these problems are usually localized to a sector on the array, it is best to print the replicate spots in separate quadrants. In lcDNA, the coefficient of variation of a gene's intensity ratios between the two channels is used as the quality index (QI). For each gene, lcDNA calculates the QI and the average signal intensity across spots and dyes. lcDNA then ranks the genes by their average signal intensity. Next, lcDNA assigns each gene to a group that contains a specified number of genes with the closest mean rank (default of 50) and rejects the gene if its QI exceeds a certain percentile (default of 90). A quality plot with cutoff lines for various percentiles is shown in Fig. 3.2a, where the QI for each gene is plotted against the genes average intensity across all channels and spots. Low intensity spots are more likely to have a low quality index. This phenomenon occurs because weak signals are more strongly impacted by small fluctuations than strong signals.

### 3.6 Normalization

Normalization is required to remove systematic variation from the intensity readings. A significant source of variation arises from differences in dye properties, such as incorporation efficiency, quantum yield, and stability. Two common normalization techniques are total intensity and invariant set.

1. *Total Intensity (TI):* In this technique, the signal (S), from each channel, for each of the n spots on the array is divided by the sum of the signal intensities for the respective channel.

$$\tilde{S}_i = \frac{S_i}{\sum_{j=1}^{n} S_j} \qquad (1)$$

The underlying assumption for this method is that each cell, regardless of the experimental condition, has the same total amount of mRNA. If a gene in a test sample is expressed at a greater level than in a reference sample then at least one of the other genes must be expressed at a lower level in the test sample, relative to the reference sample.

Since this technique is linear, and researchers have shown that the systematic variation can be highly nonlinear [2, 4, 5], a nonlinear normalization technique may be better suited for microarray data.

2. *Rank Invariant (RI):* Although there is no clear consensus on which nonlinear normalization technique is the best, a number of researchers have recognized that the information provided by non-differentially expressed genes (invariant set) can be used to normalize the data [2, 4]. In this case, we assume that a set of genes is expressed at the same level in all samples and then use this set as a basis for normalization. lcDNA uses the RI method described in [2] to determine which genes belong to the invariant set. In the RI method, each gene is assigned two ranks (one for each signal channel; $r_{i,1}$ and $r_{i,2}$); in the case of replicate spotting the ranks are based on the average intensity. Next, the difference ($d_i = |r_{i,1} - r_{i,2}|$) between the two ranks is calculated. If the difference is less than a user specified maximum distance then the spot is considered invariant. When the number of expressed genes is large (>2000), lcDNA can perform an iterative version of the RI method to select a more conserved set of genes. After finding the invariant genes, lcDNA uses the Lowess procedure [7] to fit a normalization curve to the data.

*Comparison of TI and RI:* Because the systematic variation can be nonlinear, the RI method should be superior to the TI method. The expression data often contains a number of nonlinearities. In Fig. 3a, we present an MA plot of low quality data. An MA plot is a plot of each spot's $\log_{10}$ intensity ratio (M) versus its average $\log_{10}$ intensity (A) across the two channels. When dealing with expression data containing nonlinearities, RI normalization outperforms TI normalization (Fig. 3.3b). In Fig. 3.3a, for A greater than 5 the raw data is biased towards positive $\log_{10}$ ratios; while, for A less than 5 the bias is towards negative ratios. Instead of correcting the bias, TI normalization shifts the bias towards negative $\log_{10}$ ratios (Fig. 3.3b). Application of RI normalization removes this bias and shifts the center of the distribution to 0. However, not all expression data is strongly affected by nonlinear sources of error (Fig. 3.3c); if this is the case then the gain in performance by using a nonlinear normalization technique (RI) is not always substantial (Fig. 3d).

### 3.7 Assessment of Gene-Specific Confidence Intervals

Gene expression profiles are influenced by biological, environmental, and operational factors that vary from experiment to experiment. In order to realize an estimate of expression ratios, it is necessary to design the experiment and subsequent analytical procedures to account for sample-to-sample variations. lcDNA employs the hierarchical Bayesian model developed by [2] to account for these variations and estimate confidence intervals for cDNA microarray data.

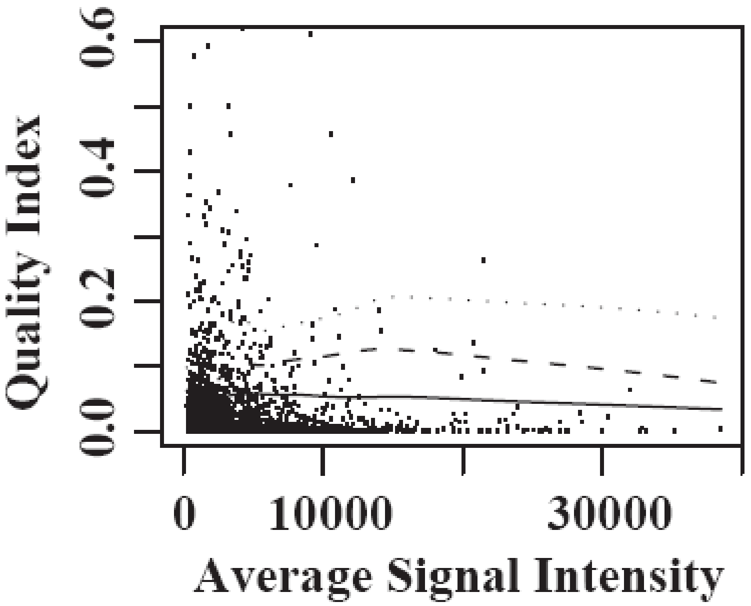


Figure 3.2. Quality filtering removes genes that exhibit a high intra-slide variability. Quality index (coefficient of variation of a gene's $\log_{10}$ intensity ratios) is plotted against the genes average signal intensity across the two channels on that particular slide for the data from lcDNA

### 3.8 The model

The variations in microarray data are described using the following hierarchical model (Tseng et al. 2000)

$$y_{gse} \sim N(\mu_{ge}, \tau_g^{\ 2}) \quad (2)$$

$$\mu_{ge} \sim N(\theta_g, \sigma_g^{\ 2}) \quad (3)$$

Here $y_{gse}$ is the normalized $\log_{10}$-ratio of gene g, slide s, and experiment *e*. As described above, $y_{gse}$ is affected by the slide-effect and uncontrollable variation between biological samples. For each biological sample, $y_{gse}$ is a sampling from a normal distribution of slide effect within the same sample, (2). $\mu_{ge}$ is the mean from tutorial slide R1S1. The quality curves were generated using a window size of 50 and percentiles of 90 (—), 95 (- - -), and 98 (· · ·), and different slides within this sample. $\tau_g^{\ 2}$ is the slide-to-slide variance for gene *g*. The within-experiment mean, $\mu_{ge}$, is in turn a sampling from a normal distribution of biological variation with a mean $\theta_g$ which measures the true log-fold-change of gene *g* and variance $\sigma_g{}^2$ which is the variance between samples. Note that only $y_{gse}$'s are observed data while $\tau_g{}^2$ , $\sigma_g{}^2$, and $\theta_g$ are unobserved parameters.

Under this model, $\theta_g$ is the unknown parameter of interest and the derived posterior distribution of $\theta_g$ is used to assess the expression level of gene *g*. If gene *g* is non-differentially expressed, then $\theta_g$ is distributed around 0. In general, to declare a gene differentially expressed means that $y_{gse}$'s deviate from 0 in the same direction and that the deviations are large compared to the magnitude of the posterior distribution of $\tau_g{}^2$ , $\sigma_g{}^2$ .

We use a Bayesian approach to incorporate prior knowledge generated from calibration experiments into the statistical analysis. The calibration experiments are used to construct prior distributions of unobserved parameters. The posterior distribution of the desired parameters is then computed to represent the combined inference of the parameters from the observed data and prior distribution. Since the posterior distributions of the parameters do not have closed form solution, a Markov chain Monte Carlo method (MCMC) is used to simulate the desired posterior distributions.

The minimal experimental design that this model can use to make sound estimates of expression ratio confidence intervals consists of two biologically independent measurements with technical replicates (Fig. 4a). Addition of two sets of calibration hybridizations, with technical replicates, will shorten the CIs (Fig. 3.4b). The yield of differentially regulated genes, at a 95% confidence level, increased from ~100 to over 1700 when the calibration hybridizations were incorporated into the analysis. Although operation of the current version (0.03) of lcDNA without technical replicates will not provide sound confidence interval estimates, the output will follow the same trend as that from experiments with technical replicates and could be useful as an exploratory exercise.

### 3.9 Discussion

Depending on the design of the experiment, the minimum number of biologically independent measurements that must be performed has been estimated to range from 6 to 25 [8]. Because of the high cost of microarray experiments, and a potentially limited availability of samples, models have been developed to estimate gene-specific confidence intervals [2, 4] when only a small number of samples are present. lcDNA employs the hierarchical Bayesian model described in [2] to estimate gene-specific confidence intervals.

With this error estimation model it is possible to construct confidence intervals for experimental designs that include biological and technical repeats (Fig. 3.4a).

If calibration hybridizations are utilized the confidence intervals will be shortened (Fig. 4b) and the yield of differentially regulated genes increases. For the data used here, the number of differentially regulated genes increased from ~100 to ~1700. The minimum number of arrays that are required to fully utilize the error estimation model is 8 (2 biological replicates and 2 calibration hybridizations, with a technical replicate for each.). When performing a time course study, or repeatedly using the same strain or cell line, the number of required arrays can be reduced by recycling the calibration hybridization information. Because the calibration hybridizations provide prior information about slide-to-slide and experiment-to-experiment variations, these data can be used as long as the same batch of slides and similar experimental conditions are used.

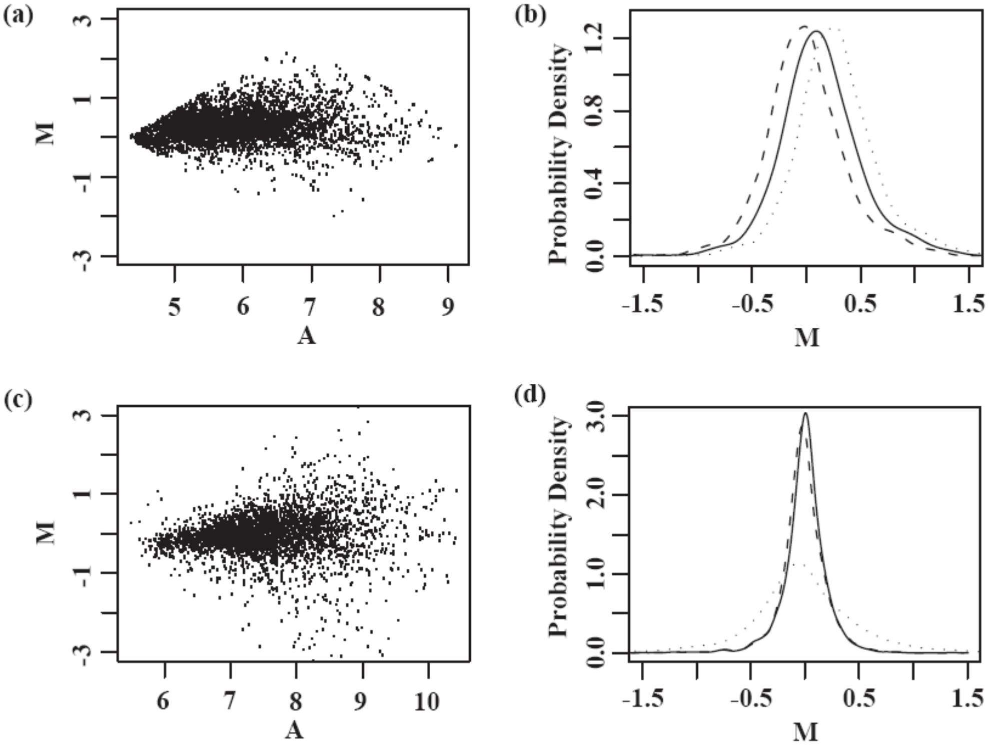


Figure 3.3 MA plots provide information about the type of normalization strategy that should be used for the data. For each spot, $M = \log_{10}$ (Channel 2/Channel 1), $A = ½\log_{10}$ (Channel 1*Channel 2). (**a**) and (**b**) show the MA plot and the probability density of M for a low quality slide before and after normalization. Because the un-normalized data (· · ·) from (**a**) contains nonlinear characteristics, rank invariant normalization (——) outperforms total intensity normalization (- - -). (**c**) and (**d**) show the MA plot and the probability density of M for slide R1S1 from the tutorial data The un-normalized data from (**c**) does not contain any significant nonlinear features, hence, the difference between rank invariant and total intensity normalization is not large. The data presented in panels (**a**) and (**b**) are from an experiment performed using the direct labeling approach, whereas data in (**c**) and (**d**) are from an experiment performed using the aminoallyl labeling method.

If one suspects that biological noise increased significantly in a different set of conditions, new calibration hybridizations should be performed. If the same calibration data are used then the minimum number of slides required for a series of n different experiments is (4+4n).

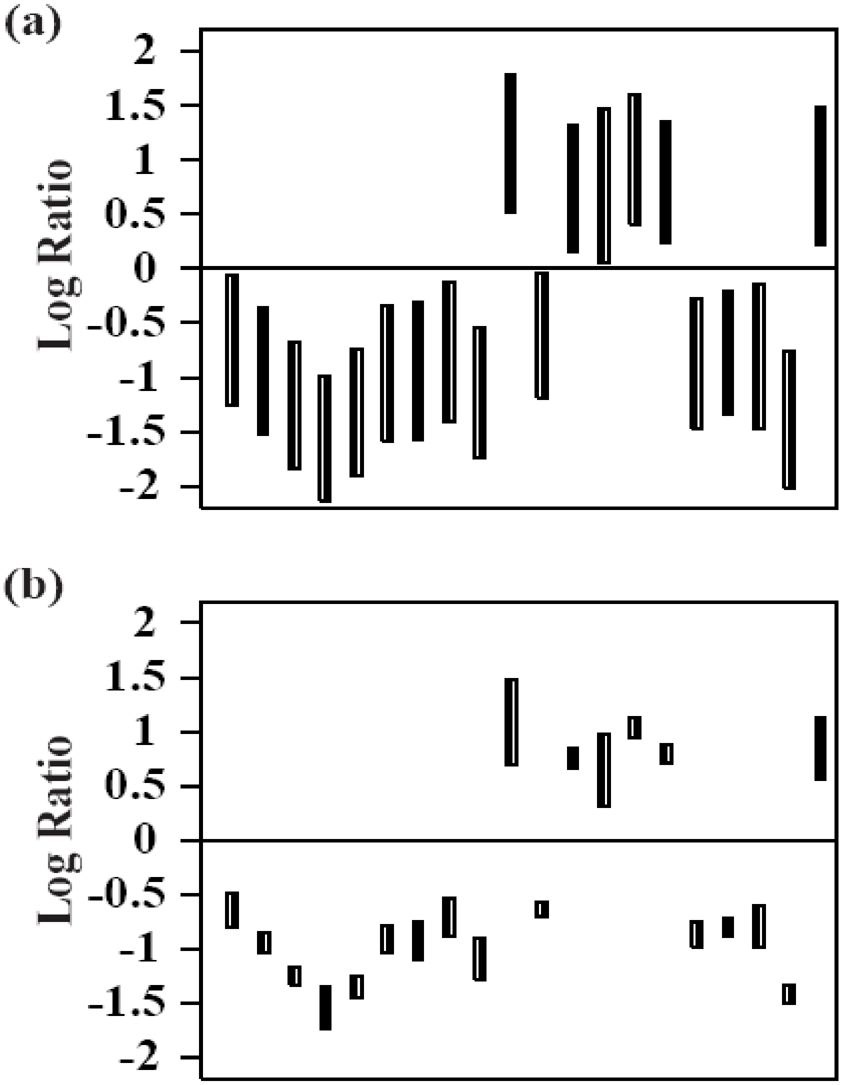


Figure 3.4. (**a**) 95 % confidence intervals of the $\log_{10}$ expression ratios for a set of two biologically independent measurements with technical replicates (tutorial slides: R1S1 and R1S2; R2S1 and R2S2). (**b**) Incorporation of a set of two biologically independent calibration hybridizations (tutorial slides: C1S1 and C1S2; C2S1 and C2S2) shortens the confidence intervals.

In addition to estimating confidence levels, lcDNA contains tools for data filtering, normalization across two channels, and data visualization. The Quality Filtering function can be used to reduce the amount of error arising from non-uniform errors on the slide surface. After the filtering is complete, the data can be visualized with an MA plot. If the data appear to contain an anomaly, such as the straight edge on the upper left side of Fig. 3a, then the filtering stringency should be increased, by decreasing the percentile, or the array should be discarded. Another way to visually assess the quality of the data is to plot the normalized $\log_{10}$ ratios obtained from one slide against its technical and biological replicates. If the measurements were perfect replicas then all points would fall on the 45 degree line. Perfection, however, is not within our grasp and the data will exhibit some degree of scatter. If the data appears to strongly deviate from the 45 degree line then one, or both, of the slides should be discarded.

Since the early development of the cDNA microarray, this technology has been evolving, [9]. Improvements have been made in the quality of array surfaces, probe quantification, and labeling and hybridization technologies. Eventually, the array fabrication and experimental protocols will reach a state where the operational variations will be dominated by biological variations. Once the technology matures to this point, it will no longer be necessary to include technical replicates in the experimental design. In order to prepare for this occurrence, we are in the process of augmenting lcDNA with the ability to make statistically sound estimates in the absence of technical replicates.

### 3.10 Methods

lcDNA is an open source program for analyzing cDNA microarray data. The lcDNA user interface was designed using the Tcl/Tk package (www.tcl.tk), version 8.4, with the BLT (http://sourceforge.net/projects/blt/), version 2.4z, and [incr Tcl] (http://incrtcl.sourceforge.net), version 3.2, extensions. BLT provides advanced data handling and graphical representation functions. [Incr Tcl] provides object-oriented programming capabilities. The computationally intense portions of the code were written in C and C++ and embedded into the Tcl code using mktclapp (http://www.hwaci.com/sw/mktclapp/). Standalone executables are available for the Microsoft Windows and GNU/Linux platforms (http://receptor.seas.ucla.edu/lcDNA/) running on Intel x86 compatible computers. The computational portions (quality filtering, normalization, and assess expression) of lcDNA are part of the DARPA BioSPICE program (www.biospice.org). The source code can be downloaded at the lcDNA website. The current version (0.03) has been developed with support from DARPA BioComp and is distributed under the DARPA BioCOMP OPEN SOURCE LICENSE version 1.0 (www.biospice.org).

### 3.11 References

[1] Miles, M.F., Nature Reviews Neuroscience, 2:441-443, 2001

[2] Tseng, G.C., Oh, M.K., Rohlin, L., Liao, J.C., and Wong, W.H., Nucleic Acids Research, 29:2549-2557, 2001

[3] Liao, J.C. and Sabattii C., ASM News, 68:432-437, 2002

[4] Dudoit , S., Yang, Y.H., Callow, M.J., and Speed, T.P., Statistica Sinica, 12:111-139, 2002

[5] Yang, Y.H., Dudoit, S., Luu, P., Lin, D.M., Peng, V., Ngai, J., Speed, T.P., Nucleic Acids Research, 30: e15, 2002

[6] Schuchhardt, J., Beule, D., Malik, A., Wolski, E., Eickhoff, H., Lehrach, H., and Herzel, H., Nucleic Acids Research, 28: e47, 2000

[7] Cleveland, W.S., The American Statistician, 35:54, 1981

[8] Simon, R.M. and Dobbin, K., BioTechniques, 34:S16-S21, 2003

[9] Schena, M., Shalon D., Davis R.W., and Brown P.O., Science, 270:467-470, 1995