Inferring gene networks from noisy under-determined experimental data

Alberto de la Fuente\(^1\)* and Dhaval Makhecha\(^2\)

\(^1\)Virginia Polytechnic Institute and State University, Virginia Bioinformatics Institute, Virginia Tech, Bioinformatics Facility I (0477), Blacksburg, VA 24061

\(^2\)Virginia Polytechnic Institute and State University, Department of Aerospace and Ocean Engineering, 215 Randolph Hall, Blacksburg VA, 24061

*Corresponding author:

Virginia Polytechnic Institute and State University

Virginia Bioinformatics Institute

Bioinformatics Facility I (0477)

Blacksburg, VA 24061

Phone: (540) 231-1791

Fax: (540) 231-2606

alf@vbi.vt.edu

Keywords: systems biology, multiple regression, gene networks, gene expression, functional genomics
ABSTRACT

Motivation: Previously an algorithm for gene network inference from gene expression perturbation data was proposed. Here the algorithm is extended by using regression with subset selection instead of matrix inversion as used in the original approach. Furthermore, this approach enables us to deal with under-determination; \textit{i.e.} when not all genes are perturbed.

Results: The performance of the algorithm is extensively evaluated on a set of data produced with gene network models at different levels of simulated experimental noise. Regression with subset selection outperforms the matrix inversion approach in the presence of experimental noise. The results on incomplete data sets show that the new method performs well at high number of perturbations, even when noise levels are high. At lower number of perturbations, while still being able to recover the majority of the connections, less confidence can be placed in the recovered edges.

Contact: alf@vbi.vt.edu
INTRODUCTION

Recently, great emphasis has been placed on inferring gene network structure from gene expression measurements using microarrays. Many strategies have been proposed, such as probabilistic models (Friedman et al., 2000; Friedman, 2004), time series analysis using linear models (D’Haeseleer et al., 1999; van Someren et al., 2000), partial correlation analysis (de la Fuente et al., 2004) and several perturbation approaches (de la Fuente et al., 2001; Wagner, 2001; de la Fuente et al., 2002; de la Fuente and Mendes, 2002; Kholodenko et al., 2002; Gardner et al., 2003; de la Fuente et al., 2004). Knowledge of gene networks will increase fundamental understanding of living cells, and has many practical purposes (Brazhnik et al., 2002), for example discovery of direct drug targets (Gardner et al., 2003; di Bernardo et al., 2005). For reviews about the current state of art in inferring networks one could refer to ((D’Haeseleer et al., 2000; Brazhnik et al., 2002; de Jong, 2002)).

Previously, a method named Regulatory Strength Analysis for gene network inference from perturbation data was proposed (de la Fuente et al., 2001; de la Fuente et al., 2002; de la Fuente and Mendes, 2002; de la Fuente et al., 2004). The central equation of the approach is

\[ \mathbf{R} \mathbf{O} = \mathbf{I} \quad \text{(Eq.1)} \]

Here \( \mathbf{R} \) is the regulatory strengths matrix that quantifies direct interactions between genes, which can be obtained by inverting the co-response matrix \( \mathbf{O} \). The co-response matrix is obtained experimentally by measuring the responses towards perturbations in the expression rates of individual genes. After the perturbation of gene \( i \) a column of matrix \( \mathbf{O} \) can be calculated as \( \Delta x_j / \Delta x_i \) for each of the \( j \) genes. When all gene expression rates have been
Inferring gene networks

perturbed $O$ will be square and $R$ (the gene network connectivity) can be obtained by
inverting $O$. Eq.1 thus is a means to calculate local interactions between genes from global
(co) responses to perturbations. The derivation of this framework has been described in detail
before (de la Fuente et al., 2001; de la Fuente et al., 2002; de la Fuente and Mendes, 2002; de
la Fuente et al., 2004). Here a short derivation is given for the sake of completeness.

We adapt a linear model in which the rate of change in expression level of each gene is
formulated as a linear function of the expression levels (mRNAs) of a specific set of input
genes and a perturbation term:

$$\Delta \left( \frac{dx_i}{dt} \right) = \sum_{j}^{n} a_{ij} \Delta x_j + \Delta u_i$$  \hspace{1cm} (Eq.2)

In steady state

$$0 = \sum_{j}^{n} a_{ij} \Delta x_j + \Delta u_i \hspace{1cm} \text{or} \hspace{1cm} \sum_{j}^{n} a_{ij} \Delta x_j = -\Delta u_i$$  \hspace{1cm} (Eq.3)

These relations can be written in matrix format

$$JX = -U$$  \hspace{1cm} (Eq.4)

Here $J$ is the Jacobian matrix, $U$ is a diagonal matrix containing perturbations $\Delta u_{ii}$ and $X$ is
a matrix containing responses $\Delta x_j$ towards the perturbations. This relationship was proposed
for inferring gene networks in (de la Fuente et al., 2001) (specifically in the form:

$$-J = \left( XU^{-1} \right)^{-1}$$

and is used in (Gardner et al., 2003) to infer gene networks from experimental perturbation data. The drawback is that the size of the perturbation should be
known to solve this equation. Measuring the size of the perturbation is experimentally
complicated, via indirect means (Gardner et al., 2003), if not impossible.
It is straightforward to obtain Eq.1 from Eq.4 in the following way: multiplying both sides with $U^{-1}$ (which is diagonal, having $1/\Delta u_{ii}$ on its diagonal elements) gives:

$$U^{-1}JX = -I$$  \hspace{1cm} (Eq. 5)

Pre-multiply both sides by $\text{diag}(X)$ (here, the $\text{diag}(\cdot)$ operator returns a diagonal matrix with elements equal to the diagonal elements of the matrix given as the argument) gives:

$$\text{diag}(X)U^{-1}JX = -\text{diag}(X)$$  \hspace{1cm} (Eq. 6)

$\text{diag}(X)$ contains the responses $\Delta x_{ii}$ of gene $i$ in experiment $i$ on its diagonal.

Post-multiplying both sides with the inverse of $-\text{diag}(X)$ gives

$$-\text{diag}(X)U^{-1}JX\text{diag}(X)^{-1} = I$$  \hspace{1cm} (Eq.7)

We define the matrices

$$R \equiv \text{diag}\left(J^{-1}\right)J = \text{diag}(X)U^{-1}J$$  \hspace{1cm} (Eq. 8)

(because rewriting Eq. 4 as $J^{-1} = -XU^{-1}$ gives $\text{diag}(J^{-1}) = -\text{diag}(X)U^{-1}$)

and

$$O \equiv -X\text{diag}(X)^{-1}$$  \hspace{1cm} (Eq. 9)

Substituting Eqs. 8 and 9 in Eq.7 we get back the original equation (Eq.1). $R$ is known as a regulatory strengths matrix, with elements $\frac{dx_i}{dv_i} \cdot \frac{dv_j}{dx_i}$. This is a measure of the effect of gene $j$ on gene $i$, partitioned as the effect that one gene has on the gene expression rate of the other (quantified by a Jacobian element) and the effect that the gene expression rate has on the gene (quantified by a global control (sensitivity) coefficient $\frac{dx_i}{dv_i}$). The regulatory
Inferring gene networks

strength matrix, a quantitative description of the gene network, can thus be experimentally obtained without the need to measure the perturbation sizes.

There three main concerns about using the linear quantitative modeling approach described above are

1) Experimental data is noisy
2) Gene-gene interactions are non-linear
3) The number of perturbation experiments is equal to the number of genes and all have to be done in order to infer the network

**ALGORITHM**

**How to deal with noisy data?**

Microarrays are known to suffer from a high noise to signal ratio. Therefore, the results obtained from direct inversion of matrix $O$ may not be reliable. Here an alternative way for solving for matrix $R$ is shown. In order to obtain matrix $O$ elements $\Delta x_j / \Delta x_i$ have to be calculated. Since dividing a noisy value by another noisy value will amplify the noise, the explicit calculation of matrix $O$ will be circumvented by using Eq. 6 instead of Eq. 1 in order to solve for $R$. Eq. 6 can be written as $RX = -diag(X)$.

**Regression with subset selection**

Instead of matrix inversion, Eq. 6 will be solved by subset selection and regression. Individual rows of matrix $R$ can be solved using
Inferring gene networks

\[ R_i \mathbf{X} = -\text{diag}(\mathbf{X})_i \]  
(Eq. 10)

where \( R_i \) are regulatory strengths of gene \( i \) towards its inputs and \( \text{diag}(\mathbf{X})_i \) is the \( i^{\text{th}} \) row of \( \text{diag}(\mathbf{X}) \). Gene networks are known to be sparse, since each gene has low number of connections and thus it can safely be assumed that most regulatory strengths are zero. Therefore there is no need to solve for all elements in row \( R_i \) but just for the non-zero regulatory strengths.

Consider Eq. 6 with a particular subset:

\[ R_{i \text{subset } k} \mathbf{X}_{\text{subset } k} = -\text{diag}(\mathbf{X})_i \]  
(Eq. 11)

\( R_{i \text{subset } k} \) contains a certain subset of size \( k \) of assumed non-zero regulatory strengths and \( \mathbf{X}_{\text{subset } k} \) contains the \( k \) corresponding rows of matrix \( \mathbf{X} \). If the subset size \( k \) is smaller than the number of perturbations \( n \) the system is over-determined, which can be solved approximately by Least Squares:

\[
\left( R_{i \text{subset } k} \right)^T = \left( \mathbf{X}_{\text{subset } k} \left( \mathbf{X}_{\text{subset } k} \right)^T \right)^{-1} \mathbf{X}_{\text{subset } k} (-\text{diag}(\mathbf{X})_i)^T
\]  
(Eq. 12)

Here \( T \) indicates the transpose. The problem with this approach is that it is unknown which of the elements in \( R_i \) are non-zero. Therefore, all possible subsets have to be tested and the subset that gives the best fit to the data is selected. Examples of quantitative indices for ‘best fit’ are residual sum of squares (RSS) and root mean squared error (RMS).

Efficient searching

Another problem with this approach is the fact that it is impossible to test all possible subset solutions for even networks of moderate size. In general \( \sum_{m=1}^{k_{\text{max}}} \frac{n!}{(n-m)!m!} \) models have to be
Inferring gene networks

tested, where $n$ is the number of genes, $m$ is the subset size and $k_{\text{max}}$ the maximum subset size considered. Unless very small values of $k_{\text{max}}$ are used, testing all possible subsets is computationally impossible for large networks. Therefore, we have to resort to strategies that make searching for the right model feasible. Many strategies have been suggested and tested on simulated datasets (van Someren et al., 2001). The best performance was shown by a method based on forward stepwise regression. First, all possible subsets of size $k=1$ are tested. Then, all possible subsets of size $k=2$ containing the best fitting input from the previous step are tested, the best subset from this step is selected and increased in size and tested and so forth, until a fixed subset size is reached. Instead of using the single best solution as a seed for the next step, one can also use the $D$ best solutions (van Someren et al., 2001). We employed the following variation of this strategy: first subsets of two genes were considered in which each gene was included in its own input subset. Given that each gene affects itself due to the mRNA degradation process (the more mRNA the faster the degradation) it is safe to assume that each gene appears in its own input subset. This assumption would be violated only if self-activation is present with exactly the same strength (but opposite in sign) of the degradation effect, yielding a zero effect of the gene on itself. The 5 best solutions are retained for the next step in which subsets of 3 are tested etc. The search is linear, each step testing only $Dn$ subsets per gene. Since the algorithm makes $k_{\text{max}}$ steps per gene, the complexity is: $n((k_{\text{max}} - 1)Dn)$. The value for $k_{\text{max}}$ was set to 20, assuming that no gene receives more than 20 inputs, to ease the computation.
**Inferring gene networks**

**How to deal with underdetermined data?**

In order to be able to solve the matrix equations (Eq.1 or equivalently Eq.6) the number of perturbations needed is equal to the number of genes in the network. However, in the previous section it is shown that with respect to subsets of inputs the data becomes over-determined. Here it is shown that it is possible to infer the network structure though not all gene expression rates have been perturbed. We start out by rewriting Eq.6 for a system in which not all gene expression rates have been perturbed. If only $p$ number of perturbations ($p<n$) are made we have the underdetermined system

$$R_{n\times n}X_{n\times p} = \begin{pmatrix} -diag(X)_{p\times p} \\ \mathbf{0}_{(n-p)\times p} \end{pmatrix}$$

(Eq. 14)

where $\mathbf{0}$ is a matrix with all zero elements. The perturbed genes are related by

$$R_{p\times n}X_{n\times p} = -diag(X)_{p\times p}$$

(Eq. 15)

which can be solved row by row as described above with the constrained that the size of the largest subset that can be selected is equal to the number of perturbations ($p$) minus 1.

For the unperturbed variables (for the sake of clarity of the following derivation, we switch to index notation):

$$\sum_{j} R_{ij} X_{jk} = 0$$

(Eq. 16)

$R_{ij}$ stands for the regulatory effect of gene $j$ on gene $i$ and $X_{ik}$ stands for the response of gene $i$ in experiment $k$ (gene expression of $k$ perturbed).

For the non-trivial solution of Eq. 16 consider the following: the regulatory strengths describing the connections into a certain gene are divided by the regulatory strength of that
gene on itself. This yields a regulatory strength matrix in which element in row $p$ in column $p$ is equal to one and all other elements in a row are scaled by element $p$.

This can be written as:

$$\sum_{j \neq i}^{n} \frac{R_{ij}}{R_{ii}} X_{jk} + \frac{R_{ii}}{R_{ii}} X_{ik} = 0 \quad \text{or} \quad \sum_{j \neq i}^{n} \frac{R_{ij}}{R_{ii}} X_{jk} = -X_{ik} \quad \text{(Eq. 17)}$$

For the unperturbed genes we can thus only solve for ratios between regulatory strengths rather than their absolute value. Again, these equations can be solved with the approach outlined above.

**Selecting genes to perturb**

The amount of information of the network that one gets out of a specific perturbation depends on factors like the topology of the network and where the perturbed gene is located in the network. When performing a set of perturbation experiments it is relevant to rationally select the perturbed genes in a way that maximizes the amount of information. An approach similar to the one described in (Tegner et al., 2003) is adapted. This approach starts by perturbing a randomly selected gene. Then, the gene that had the lowest response towards the first perturbation was perturbed next. The responses towards the two perturbations are summed and the gene with the lowest summed response is perturbed next and so forth until $p$ perturbations are made. It has been shown that this way of selecting genes to perturb is more informative then selecting genes randomly (Tegner et al., 2003).
RESULTS

In order to evaluate the above outlined strategy, it was implemented in MATLAB 7.0 and is applied to a set of artificially generated data. Data was simulated by solving Eq.4 for $X$ including an additional error term. Matrix $U$ is set to identity, since for the linear model the size of the perturbation is not important.

$$X = -J^{-1} + \epsilon$$  \hspace{1cm} (Eq. 18)

Ten random 100 by 100 Jacobian matrices were created. The number of non-zero elements in each row (number of connections into each gene) was determined by sampling from a Poisson distribution with a mean of 10, to give rise to random Erdős-Rényi networks (Erdős and Rényi, 1960). For each row, the particular non-zero elements were determined by sampling from a uniform distribution. The values for the coefficients were sampled from a normal distribution with mean of zero and standard deviation of 0.25, so that about 96% of the values are between -0.5 and 0.5 and the diagonal elements were set to –1. These values have been chosen in order to guarantee the so obtained Jacobian has all negative real eigenvalues and thus corresponds to a gene network in a stable steady state (which is a requirement of the linear approximation to be valid).

The error matrix in Eq.18 accounts for any deviation of the observed values in $X$ that is not accounted for by the linear model. In this way $\epsilon$ can be simply seen as experimental error, as well as the effects of non-linearity of genetic interactions. $\epsilon$ can be seen to contain the second and higher order terms of the Taylor expansion, $\frac{\partial^2 x_i}{\partial u_j^2} \Delta u_j^2 + \frac{\partial^2 x_i}{\partial u_j^3} \Delta u_j^3 + h.o.t.$, of
which only the first term is explicitly considered in our model. It is important to note that experimental error and non-linearity are somewhat mutually exclusive: the bigger the perturbation the smaller the experimental error will be relative to the measured responses, but the non-linear effects will be larger. On the other hand, small perturbations validate the linear approximation, but the responses will easily get lost in the experimental noise. Ideally, the size of the perturbations should be such that a good trade off is made between deviations from linearity and the measurement noise. To test three different scenarios all values in $\epsilon$ were sampled from a normal distribution with a mean of zero and standard deviation of 10%, 25% and 50% of the absolute value of the corresponding elements in $-$J$^{-1}$ (the purely linear responses).

The Regulatory Strength Analysis is a framework to infer and quantify genetic interactions. However, for the analysis to be quantitative, data of good quality should be used. Data with a high experimental noise level, which in addition displays non-linearity of interactions, will only yield a qualitative description. Therefore the algorithm is evaluated in a qualitative way by scoring the ability of the algorithm to predict the absence and presence of edges in the network, rather than evaluating how good the actual values of the actual regulatory strengths (as calculated by Eq. 8) are approximated. Two measures for qualitative evaluation are employed: the False Discovery Rate (FDR) and the Power. The FDR is expressed as the number of wrongly predicted edges divided by the total number of predicted edges, thus indicates how reliable the predictions made by the algorithm are. The Power is defined as the number of edges correctly inferred as a fraction of the total number of edges in the network, thus indicating how well the algorithm predicts what is to be predicted.
We first assume a complete dataset, meaning that all genes have been individually perturbed in experiments. In this case it is possible to solve the system in two ways 1) the original proposed way by inverting the $\mathbf{O}$ matrix (de la Fuente et al., 2001; de la Fuente et al., 2002; de la Fuente and Mendes, 2002; de la Fuente et al., 2004) and 2) the proposed regression with subset selection approach. Figure 1 shows the results on the analysis of the 10 networks, with 5 noise realizations each. These results were obtained by using a row dependent threshold below which elements in $\mathbf{R}$ were neglected, taken as 10% of the largest absolute value in a row.

The power of both approaches is very similar. Both approaches seem to discover an equal amount of the real edges in the networks, with almost maximum power at 10% noise, about 90% power at 25% noise and dropping to about 60% at the 50% noise level. However, when looked at the FDR it can be seen that the regression by subset selection approach outperforms the matrix inverse approach. While at the 10% noise level the differences are small, at the 25% noise level the FDR of the matrix inverse approach is about 60% meaning that the networks identified contain 6 erroneous connections out of every 10 identified, versus about 2 out of 10 edges in the networks inferred using the regression approach. At the 50% noise level about 4 out of 10 edges are wrong in the networks obtained by the regression approach, while 9 out of 10 are wrong in the networks obtained by the inverse approach! Although about the same amount of correct edges are found in both approaches, much more confidence can be placed in each of the edge found with the regression approach.
To test the performance on incomplete perturbation data, the algorithm was run on data subsets of 75, 50 and 25 perturbations (out of the full 100 total number of perturbations) with different levels of noise. The results are presented in Figure 2.

As expected the results are best for the case of having complete data. Except in the 25% data case, the power between cases with different fractions of complete data stayed quite constant, but the FDR increases with lower number of perturbations, making the networks discovered less reliable. Still, figure 2 shows that even when half of the total number of perturbations have been made the inference results are good at a noise levels of 25% and below.

DISCUSSION

Inferring gene networks is a daunting task, especially given that currently produced experimental microarray data contains high levels of technical noise. Previously, an algorithm for inferring gene networks from perturbation experiments was proposed. The algorithm required data consisting of measurements of gene expression levels of all genes in the network after single gene perturbations. It was shown that if all genes have been perturbed individually and their effects on the global gene expression measured, it is straightforward to work out the underlying gene network by calculating a matrix directly from gene expression data and taking the inverse of that matrix yielding a matrix with non-zero elements corresponding to interactions in the gene network. In this paper, that approach has been extended by using a regression with subset selection approach to be able to deal
with two relevant situations: 1) when experimental noise is present in the datasets, and 2) having an incomplete dataset, \textit{i.e.} when all perturbations have not (yet) been made.

The new approach deals much better with noisy data than the original matrix inverse approach especially at higher noise levels. While the power was very similar, the FDR of the inverse approach was much higher, making the discovered networks much more unreliable.

It was shown that the more perturbations made the better, giving best results for complete data. The power between cases with different fractions of complete data interestingly stayed quite constant (except in the 25\% data case), but the FDR increases with lower number of perturbations, making the networks discovered less reliable. At the higher noise level of 50\% the results are not satisfactory, but this was expected; if data is of such poor quality it becomes hard to infer anything. However, we could still identify correctly half of the edges in the networks, but on the other hand, half of the edges in the recovered networks were erroneous, making the results of application of this method to such highly noisy data quite unreliable. For a quite realistic noise level of 25\%, the overall results were good. Preliminary studies of larger networks of 250 genes show a similar relationship between the percentage of data and the power and FDR (results not shown).

Recently a similar noise study (di Bernardo \textit{et al.}, 2004) has been performed, but using Eq. 4 to infer the gene networks. The results of the current approach are better than their approach for the low noise levels, but their results were better for high noise levels. There are several reasons for these differences. One is the fact that in their approach there are two sources of noise. The first noise source, common to ours, is matrix $X$, consisting of the noisy expression values and the second is matrix $U$ consisting of noisy rate perturbation measurements. Also,
the way noise was simulated in their study was in proportion to the average absolute value in matrix $X$, in order to add the same amount of error to each element, while in the study presented in this paper noise was added proportional to the size of each element. We believe this way of adding noise more closely reflects noise in real experimental data, since large values will receive more absolute noise than low values.

The improved way of inferring gene networks using regulatory strengths presented in this paper is computationally much more expensive than the original approach, which required simply taking one matrix inverse. However, the results shown in this paper certainly emphasize the need for this computational approach when data is noisy and in the case of underdetermined data the matrix inverse approach obviously cannot be applied. The above-described algorithms were implemented in MATLAB 7. Solving for one network of 100 genes took about 950s on an AMD 3200+. For genome size gene networks supercomputing facilities and parallel implementation of the algorithm will most certainly be necessary.

**CONCLUSION**

The results shown here indicate that the Regulatory Strength Analysis now can be more confidently applied to noisy experimental data, even when not all necessary experiments have been done. Still, best results are obtained when all experiments are done, so that it should be the ultimate goal to do these experiments. Still, our results indicate that it is not necessary to wait until all experiments have been done to be able to infer networks (as was the case for the original approach using matrix inverse), but inferring can be started during
the iterative experimental creation of gene perturbations. However, during the early stages, having low numbers of perturbation experiments completed, the recovered networks are less reliable then at the later stages when more and more experiments have been carried out. Clearly, the reliability at each stage depends also on the amount of noise present in the data.

ACKNOWLEDGEMENTS

We would like to thank Valerie and Edwin Robinson for their continuing moral support.
REFERENCES


Inferring gene networks


Inferring gene networks


FIGURE LEGENDS

**Figure 1:** Testing the effects of noise levels of the regression by subset selection approach versus the matrix inverse approach. FDR and Power are shown for 5 noise realizations for each of 10 random gene networks of 100 genes and average number of 10 inputs per gene at noise levels of 10%, 25% and 50%. ‘Reg’ and ‘Inv’ indicate that the values are obtained using the regression and the matrix inverse approach, respectively.

**Figure 2:** Testing the effects of noise and incomplete data. The datasets contain 25%, 50%, 75% or 100% (all data) of the maximum number of single gene perturbations. Average FDR and Power are shown for 5 noise realizations for 10 random gene networks of 100 genes each with an average of 10 connections per gene at noise levels 0%, 10%, 25% and 50%.
FIGURES

Figure 1 – de la Fuente and Makhecha, 2005
Figure 2 – de la Fuente and Makhecha, 2005