

Piecewise Linear Modeling of Gene-Protein Interaction Networks

Ronald L. Westra¹, Ralf L.M. Peeters¹, Goele Hollanders², Karl Tuyls³

1. Dept. Mathematics, Maastricht University, Maastricht, The Netherlands

2. Dept. Computer Science, Hasselt University, Hasselt, Belgium,

3. Dept. Computer Science, Maastricht University, Maastricht, The Netherlands

E-mail to: westra@math.unimaas.nl

Abstract. In this study we will focus on piece-wise linear state space models for gene-protein interaction networks. We will follow the dynamical systems approach with special interest for partitioned state spaces. From the observation that the dynamics in natural systems tends to punctuated equilibria, we will focus on piecewise linear models and sparse and hierarchic interactions, as for instance described by Glass, Kauffman, and de Jong. Next, the paper is concerned with the identification (a.k.a. reverse engineering and reconstruction) of dynamic genetic networks from microarray data. We will describe exact and robust methods for computing the interaction matrix in the special case of piecewise linear models with sparse and hierarchic interactions from partial observations. Finally, we will analyze and evaluate this approach with regard to its performance and robustness towards intrinsic and extrinsic noise.

Keywords: piecewise linear, robust identification, hierarchical networks, gene expression data, gene regulatory networks.

1 Introduction and problem statement

This paper is concerned with the identification of dynamic gene-protein interaction networks with intrinsic and extrinsic noise from empirical data, such as a set of microarray time series.

Prerequisite for the successful reconstruction of these networks is the way in which the dynamics of their interactions is modeled. The formal mathematical modeling of these interactions is an emerging field where an array of approaches are being attempted, all with their own problems and short-comings. The underlying physical and chemical processes involved are multifarious and hugely complex. This condition contrasts sharply with the modeling of inanimate Nature by physics. While in physics huge quantities of but a small amount of basic types of elementary particles interact in a uniform and deterministic way provided by the fundamental laws of nature, the situation in gene-protein interactions deals with tens of thousands of genes and possibly some million proteins. The quantities thereby involved in the actual interactions are normally very small, as one single protein may be able to (in)activate a specific gene, and thereby change the global state of the system. For this reason, gene regulatory systems are much

more prone to stochastic fluctuations than the interactions involved in normal anorganic reactions. Moreover, each of these interactions is different and involves its own peculiar geometrical and electrostatic details. There are different processes involved like transcription, translation and subsequent folding. Therefore, the emergent complexity resulting from gene regulatory networks is much more difficult to comprehend.

In the past few decades a number of different formalisms for modeling the interactions amongst genes and proteins have been presented. Some authors focus on specific detailed processes such as the circadian rhythms in *Drosophila* and *Neurospora* [10], [11], or the cell cycle in *Schizosaccharomyces* (Fission yeast) [14]. Others try to provide a general platform for modeling the interactions between genes and proteins. For a thorough overview consult de Jong (2002) in [2], Bower (2001) in [1], and others [6], [13].

We will focus on dynamical models, and not discuss static models where the relations between genes are considered fixed in time. In discrete event simulation models the detailed biochemical interactions are studied. Considering a large number of constituents, the approach aims to derive macroscopic quantities. More information on discrete event modeling can be found in [1].

2 Modeling gene-protein interactions as a piecewise linear system

The traditional approach to modeling the dynamical interactions amongst genes and proteins is by considering them as biochemical reactions, and thus representing them as 'rate equations'. The concept of chemical rate equations consists of a set of differential equations, expressing the time derivative of the concentration of each constituent of the reaction as some rational function of the concentrations of all the constituents involved. Though the truth of the underlying biochemical interactions between the constituents is generally accepted, a rate equation is not a fundamental law of Nature, but a statistical average over the entire ensemble of molecular collisions that contribute to an actual chemical reaction [22]. So, rate equations are statistical approximations that – under certain conditions – predict the average number of reactive collisions. The actual observed number will fluctuate around this number, depending on the details of the microscopic processes involved. In case of biochemical interactions between genes and proteins the applicability of the concept of rate equations is valid only for genes with sufficient high transcription rates. This is confirmed by recent experimental findings by Swain and Elowitz [5], [16], [18], [19].

From the above, we may conclude that modeling can only be successful for genes with sufficiently high transcription rates. Even in the optimal case, we would obtain a high-dimensional (reflecting the number of genes, RNAs, and proteins involved – so tens of thousands), non-linear, differential equation, that is subject to substantial stochastic fluctuations. Much more problematic is the fact that the precise details of most reactions are unknown, and therefore cannot be modeled as rate equation. This could be compensated by a well-defined parametrized generic form of the interactions, such that the parameters could be estimated from sufficient empirical data. A generic form based on rational positive functions is proposed by J. van Schuppen [23]. However, in the few cases where parts of such interaction networks have been described from

experimental analysis, like the circadian rhythms in certain amoeba [10], or the cell cycle in fission yeast [14], it is clear that such forms have a too extensive syntax to be of any practical use.

Let us for the moment forsake these problems, and consider the dynamics of gene-RNA-protein networks. When we assume a stochastic differential equation as model for the dynamics of the interaction network, the relation can be expressed as:

$$\dot{x} = f(x, u|\theta) + \xi(t) \quad (1)$$

Here $x(t)$, called the state-vector, denotes the N gene expressions and M RNA/protein densities at time t – possibly involving higher order time derivatives. $u(t)$ denotes the P controlled inputs to the system, such as the timing and concentrations of toxic agents administered to the system observed. $\xi(t)$ denotes a stochastic Gaussian white noise term. This expression involves a parameter vector θ , that contains the coupling constants between gene expressions and protein densities. We can consider this system as being represented by the state vector $x(t)$ that wanders through the (at least) $(N + M)$ -dimensional space of all possible configurations. In the formalism of dynamic systems theory, eventually x will enter an area of attraction, and become subject to the influence of an attractor. An attractor here can be an uniform convergent attractor, a limit cycle, or a 'strange attractor'. We can understand the entire space as being partitioned into cells, where such attractors – or their antagonists so-called repellers – reign. Thus, the behavior of x can be described by motion through this collection of cells, swiftly moving through cells of repellers, until they enter the basin of attraction of an attractor. Under the effects of external agents via the vector $u(t)$ or by stochastic fluctuations via $\xi(t)$ they can leave this cell, and start wandering again, thereby repeating the process. Now, a vital assumption is that in each cell the behavior is governed by specific (un)stable equilibrium points, and therefore it is possible to make a linear approximation of equation 1 in the cell with index l as:

$$\dot{x}(t) = F_l x(t) + G_l u(t) \quad (2)$$

In case of a uniform attractor the largest eigen-value of F_l will be negative, and in case of a uniform repeller the smallest eigen-value will be positive. We can now formalize the qualitative behavioral dynamics of gene-protein interactions as predominantly linear behavior near the stable equilibria – called the steady states, interrupted by abrupt transitions where the system quickly relaxes to a new steady state, either externally induced or by process noise.

In biology such behavior is frequently observed, as for instance in embryonic growth where the organism develops by transitions through a number of well-defined 'check points'. Within each such checkpoint the system is in relative equilibrium. There is an ongoing debate on mathematical modeling of cell division as *checkpoint mechanisms* versus *limit-cycle oscillators*, see [20]. We will follow the view of *piecewise linear behavior* (PWL, also known more appropriately as piecewise *affine* behavior). This approach corresponds to the piecewise linear models introduced by Glass and Kauffman [9], and the qualitative piecewise linear models described by de Jong et al. [2], [3].

3 The identification of *piecewise linear networks* by L_1 -minimization

Next, we will be concerned with the identification (a.k.a. *reverse engineering* or *reconstruction*) of piecewise linear gene regulatory systems from microarray data. The nature of our problem – few microarray experiments and lots of genes – implies that we are dealing with *poor data* (as opposed to *rich data*), where the number of measurements is *a priori* insufficient to identify all parameters of the system. One standard approach to circumvent this problem is by dimension reduction through the clustering of related genes. We consider the case where time series of genome-wide expression data is available. The case of the identification of a *simple* linear system is discussed in Peeters and Westra [15], [26], and Yeung et al. in [27]. In the following, we will be concerned with the identification of *piecewise* linear systems. Our aim is to obtain the gene-gene interaction matrix. This matrix can be interpreted as a connectivity matrix, and so directly relates to the graph of the gene regulatory network. With this network we are able to make statements like: ‘the expression of this gene causes that and that cluster of genes to alter their expression in this and this way’.

Let us in the following assume a dynamical input-output system Σ that switches irregularly between K linear time-invariant subsystems $\{\Sigma_1, \Sigma_2, \dots, \Sigma_K\}$. Let $S = \{s_1, s_2, \dots, s_{K-1}\}$ denote the set of – possibly unknown – switching times, i.e. the time instants $t = s_l$ that the system switches from subsystem Σ_l to Σ_{l+1} . Similarly as with the simple linear networks, we assume *Hankel matrices* $X = (x[1], x[2], \dots, x[M])$, and $U = (u[1], u[2], \dots, u[M])$ at M sampling times $T = \{t_1, t_2, \dots, t_M\}$, representing full observations of the N states and P inputs. The interval between two sample instants is denoted as $\tau_k = t_{k+1} - t_k$. In first instance we assume that the system is sampled on regular time intervals, i.e. that the sample intervals are equal to τ . Within one subsystem Σ_l the relation between the inputs $u(t)$ and outputs $y(t)$ is represented as a state-space system of first-order differential (for continuous time systems) or difference equations (for discrete time systems), using an auxiliary vector $x(t)$ spanning the so-called subspace.

Continuous time:

$$\dot{x}(t) = F_l x(t) + G_l u(t), \quad (3)$$

$$y(t) = H_l x(t) + J_l u(t). \quad (4)$$

Discrete time:

$$x[k + 1] = A_l x[k] + B_l u[k], \quad (5)$$

$$y[k] = C_l x[k] + D_l u[k]. \quad (6)$$

The relation between these is given by:

$$A_l = e^{\tau F_l}, \quad (7)$$

$$B_l = e^{\tau F_l} G_l. \quad (8)$$

with $x[k] = x(t_k)$.

3.1 Determination of the new state equilibrium points

Moreover, in each new state the new equilibrium point $\mu_l \in \mathbb{R}^N$ has also to be established. The linearization near μ_l can be written as:

$$\frac{\partial}{\partial t}(\mu_l + (x - \mu_l)) = F_l(x - \mu_l) + G_l u + \mathcal{O}(\|x - \mu_l\|^2) \quad (9)$$

which can be rewritten as: $\dot{x} = F_l x + \tilde{G}_l \tilde{u}$, with:

$$\tilde{G}_l = (G_l | -F_l \mu_l), \quad (10)$$

$$\tilde{u} = \begin{pmatrix} u \\ 1 \end{pmatrix}. \quad (11)$$

The reasoning is similar in the discrete case, and we obtain: $x[k+1] = A_l x[k] + \tilde{B}_l \tilde{u}[k]$. Therefore, we can follow the original formulation and, using \tilde{u} rather than u as input, estimate A_l and \tilde{B}_l , and using:

$$\tilde{B}_l = (B_l | -A_l \mu_l), \quad (12)$$

to compute μ_l and B . We will follow this approach, and from here on drop the *tilde*, and simply write B_l for $(B_l | -A_l \mu_l)$, and $u[k]$ for $\begin{pmatrix} u[k] \\ 1 \end{pmatrix}$.

3.2 General dynamics of switching subsystems

In the context of piecewise linear systems of gene regulatory systems, the dynamics is slightly different to the case of simple linear systems as in [15]. In our context we assume that we observe *all* N genes, and that there is no direct through-put. This means that $C_l = I$ and $D_l = 0$ for all l . Therefore, we can suffice with equation 5 corrected for the equilibrium point:

$$x[k+1] = A_l x[k] + B_l u[k]. \quad (13)$$

We furthermore assume that the system matrices in these equations are constant during intervals $[s_l, s_{l+1})$, and abruptly change at the transition between the intervals at $t = s_{l+1}$. We assume that on the time scale τ the system has relaxed to its new state. This means that we do not observe *mixed states*, which would severely complicate the problem of identification.

Finally, we define the *weights* w_{kl} , as the membership functions of observation k to subsystem Σ_l ; if observation $\{x[k], u[k]\}$ belongs to system Σ_l then $w_{kl} = 1$, if $\{x[k], u[k]\}$ does not belong to Σ_l then $w_{kl} = 0$. This definition allows for a *fuzzy* definition of weight, such that $w_{kl} \in [0, 1]$. *A priori*, we thus can state two constraints on w :

$$\forall_{k,l} w_{kl} \in [0, 1], \quad (14)$$

$$\forall_l \sum_l w_{kl} = 1. \quad (15)$$

To make sure that there'll be no frequently switchings between the different systems and we can talk about a perfect block-matrix, the weight matrix W will be optimized by adding an extra constraint:

$$\begin{aligned}
\sum_{k=1}^{M-1} |W_{k+1,1} - W_{k,1}| &= 1 \\
\sum_{l=2}^{K-1} \sum_{k=1}^{M-1} |W_{k+1,l} - W_{k,l}| &= 2 \\
\sum_{k=1}^{M-1} |W_{k+1,K} - W_{k,K}| &= 1
\end{aligned}
\tag{16}$$

This constraint takes care that the number of one-blocks (vector existing only of ones) will be as small as possible so that first system one, than system two, ... and finally system K will be activated. By consequence the first column of W (the first system, W_{k1}) starts with an one-block and jumps on the first switching time, S_1 , over into a zero-block (vector existing only of zeros). The last column of W (the last system, W_{kK}), behaves in the opposite way. It starts with zeros and jumps at the last switching time, S_{K-1} , over into ones and remains this. Between the first and the last system, all other systems, l , start with zeros and switch on switching time S_{l-1} over into an one-block. Until switching time S_l is reached, then the ones will be switched again into zeros.

The challenge in system identification is to estimate the relevant model parameters in piecewise linear dynamics from empirical observations. The success of this approach depends on the amounts of empirical data available – *rich* or *poor*, the validity of the mathematical model, the levels of process noise and measuring noise, and the nature of the sampling process. In case of regular sampling the discrete model 5 can be applied which leads to more straightforward techniques than the continuous model 3 that should be used in case of irregular sampling. In the following sections we will study a number of these conditions in more detail.

3.3 Identification of PWL models with *unknown* switching and *regular* sampling from *poor* data

The assumption that the switching times between the linear subsystems are completely known suits various experimental conditions, as for instance when toxic agents are administered. In many biological situations, however, the exact timing between subsystems is not known, as during embryonic growth and in many metabolic processes.

As an extension to the simple linear systems in case state derivatives are available When a sufficiently accurate record of estimates of the state derivatives $\dot{X} = \{\dot{x}[1], \dot{x}[2], \dots, \dot{x}[M]\}$ is available, we can simply rewrite this problem as a special case of the method described in the case of a simple linear problem as in [15]. In fact, by

exploiting the data $\mathcal{D} = \{X, U, \dot{X}\}$, the problem can be stated as a linear equation in terms of new matrices H_1 and H_2 as:

$$\dot{X} = H_1 X + H_2 U. \quad (17)$$

In this equation the matrices H_1 and H_2 relate to the – unknown – system matrices $\{A_1, B_1, \dots, A_K, B_K\}$ and ditto unknown weights $\{w_{kl}\}$ as:

$$\text{vec}(H_1) = W \cdot \text{vec}(A), \quad (18)$$

$$\text{vec}(H_2) = W \cdot \text{vec}(B). \quad (19)$$

The matrices A , B , and W are composed as follows:

$$A = \begin{pmatrix} A_1 \\ \dots \\ A_K \end{pmatrix}, \quad B = \begin{pmatrix} B_1 \\ \dots \\ B_K \end{pmatrix}, \quad W = w \otimes I_{N^2} = \begin{pmatrix} w_{1,1}I_{N^2} & \dots & w_{1,K}I_{N^2} \\ \dots & \dots & \dots \\ w_{M,1}I_{N^2} & \dots & w_{M,K}I_{N^2} \end{pmatrix}, \quad (20)$$

where \otimes is the Kronecker-product, and I_{N^2} is the $N^2 \times N^2$ identity matrix. Note that equation 17 is not anymore a linear problem, as the unknown matrices A , B , and W appear in a non-linear way in the equation. This equation is exactly of the type of simple linear networks as in [15]. Therefore, its solution method is fully applicable, so that an efficient and accurate algorithm is available for solving this problem in terms of H_1 and H_2 . However, now the problem has shifted to solving two additional non-linear equations:

$$W \diamond A = H_1, \quad (21)$$

$$W \diamond B = H_2. \quad (22)$$

where A , B , and W have to be solved from the known – i.e. computed – matrices H_1 and H_2 . The operation \diamond makes the relations in equations 21 and 19 explicit. This is an underdetermined system that can only be solved by additional information, such as assuming sparsity for A , and a block structure for W , such as the two constraints in equations 14 and 15.

This non-linear problem can thus be solved in terms of H_1 and H_2 , but not in terms of A , B , and W . It is a bilinear problem in terms of A and B for fixed W , otherwise it is a quadratic problem. As a quadratic programming problem this is not a well-posed problem, i.e. it has a nonsingular Jacobian at optimality and is ill-conditioned as the iterates approach optimality. Therefore, we follow a different approach and split the problem in two LP-problems that are well-posed. The approach is as follows: (i) initialize A , B , and W , (ii) perform the iteration:

1. Compute H_1 and H_2 , using the approach from Peeters and Westra [15] on equation 17,
2. Using fixed values for the weights W , compute A and B using equations 21, and 22,
3. Using fixed values for matrices A and B , compute the weights W using equations 14, 15, 21, and 22,

until: (iii) a cumulative weighted error criterion \mathcal{E} has converged sufficiently – or a maximum number of iterations has passed. A proper choice for the criterion function is:

$$\mathcal{E}(A, B, W|\mathcal{D}) = \sum_{k,l} w_{kl} \|A_l x[k] + B_l u[k] - \dot{x}[k]\|_2^2 \quad (23)$$

This problem can be solved by minimizing the quadratic L_2 -criterion subject to mentioned constraints, for instance by a gradient descent method. We can, however, formulate a different approach for solving this problem by defining an alternative criterion function \mathcal{E} , namely as a linear L_1 -criterion:

$$\mathcal{E}_1(A, B, W|\mathcal{D}) = \sum_{k,l} w_{kl} \|A_l x[k] + B_l u[k] - \dot{x}[k]\|_1 \quad (24)$$

This expression allows for an LP-formulation of the problem, in which \mathcal{E}_1 serves as the objective function. Thus, we can split the non-linear optimization problem as two separate LP-formulations that are successively applied in the iteration; (i) an LP-problem LP_1 for obtaining the system matrices A and B from minimizing objective function \mathcal{E}_1 with given weights w , subject to the constraints in equations 21 and 22; and (ii) an LP-problem LP_2 for obtaining the weights w from minimizing objective function \mathcal{E}_1 with given system matrices A and B , subject to the constraints in equations 14, 15, 21, and 22.

We will revisit this philosophy in the next Section, when reviewing the more realistic case when the state derivatives of the gene expressions are *not* available.

4 Numerical experiments and performance of the approach.

This approach resulted in an efficient and fast algorithm that is able to accurately estimate the gene-gene coupling matrix for tens of thousands of genes based on only several hundred genome wide measurements, and that is robust towards measurement noise. With increasing measurement noise or decreasing number of measurements the approach retains the strongest gene-gene coupling links - i.e. the largest modal value of the coupling matrix A - longest, see Figure 1. A basic assumption in the approach is the sparsity of the underlying gene-gene coupling matrix, represented by the number of non-zero entries per row. If this number grows above a certain threshold the performance of the approach is severely affected, see Figure 2a. A number of numerical experiments were performed with this approach. These controlled experiments consist of the comparison of reconstructed network with the - known - original network structure. They were all performed on a PC with an Intel pentium M processor of 1.73 GHz and 1 GB RAM memory under Windows XP professional, using Matlab 6.5 release 13 including the optimization toolbox. The Matlab routine `linprog` was used to solve LP problems; its default solution method is a primal-dual interior point method, but an active set method can optionally be used too. *For larger problems it turned out to be essential for obtaining reasonable computation times, that the LP problems were solved by application of the active set method on the dual problem formulation. Therefore this method was adopted throughout all the experiments.* In line with the definitions above,

we use the parameters N , M , K to quantify the size and complexity of the input. In addition, the sparsity of the interaction matrix A is measured by the number of nonzero entries per row and denoted by k (which should be much less than N). To quantify the quality of the resulting approximation A_{est} of A^* two performance measures are introduced: the number of errors N_e and the CPU-time T_c as clocked on the same platform.

1. *The number of errors N_e .*

Errors in the reconstruction are generated by the failure of the algorithm to identify the true non-zero elements of the original sparse vector x_0 . These errors stem from false positives and false negatives in the reconstructed vector x_d . Their numbers are added up to produce the total number of errors N_e .

2. *The CPU-time T_c .*

Using internal clocking, the time T_c required to perform the full computation was measured. As all numerical experiments are executed on the same platform under similar conditions, this provides a measure to compare problem instances.

The numerical experiments clearly demonstrate the range where the approach is effective. For relatively moderate noise levels and a high degree of sparsity i.e., a small number k of nonzero elements in the rows of matrix A - and not too many external stimuli p and switching times K , the approach allows one to reconstruct a sparse matrix with great accuracy from a relative small number of observations $M \ll N$. For example, a row of A with 30,000 components of which all but 10 are equal to zero, can be efficiently reconstructed from just 150 independent measurements, see Figure 2b. The sparsity property of A fits in nicely with the technique of L_1 -minimization, which automatically will always set many entries of the solution A^* to zero, whereas L_2 -regression would spread out the error over all components, thus creating many small components. Reconstruction of large networks from this approach is straightforward: each of the rows of the gene-gene interaction matrix can be computed independently from the same set of micro-array experiments.

What will happen if the number of genes and/or the number of measurements increases is presented by figures 3 and 4. The higher the number of genes and/or measurements, the higher the CPU-time, T_c , of the algorithm will be. When we observe the influence of the number of subsystems K , we can constatate the same, nl. an increasement of the number of subsystems influences the increasement of the computation time, in a positive way. This last part depends on the fact that the simple linear model will be used for the computation of each subsystem. Figure 5 shows us the rate of error in function of the number of measurements. When the number of subsystems is one, $K = 1$ and the number of genes is small, $N = 10$, (figure 5a) there are only a small number of measurements necessary to have an acceptable errorrate. When the number of subsystems increases (figure 5b) more measurements are necessary to have approximately the same errorrate. To be more exact, for $K = 1$ only 10 measurements and for $K = 3$, 30 measurements have to be available. This we can explain as follows: the available measurements will be devided over the subsystems. So if there are 3 subsystems and 30 measurements, each subsystem will have approximately 10 measurements. By keeping figure 5a in mind, we can conclude that this number of measurements is enough to have an acceptable low rate of error for each subsystem. And so the errorrate for the complete system will be small too. The last figure, figure 6, indicates respectively the

relation between the CPU-time and the errorrate with the number of subsystems. For both, the number of subsystems has a same impact. The more subsystems, the more possibilities there are for the identification of the weight matrix, W , which can be reflected as an exponential grow in function of K .

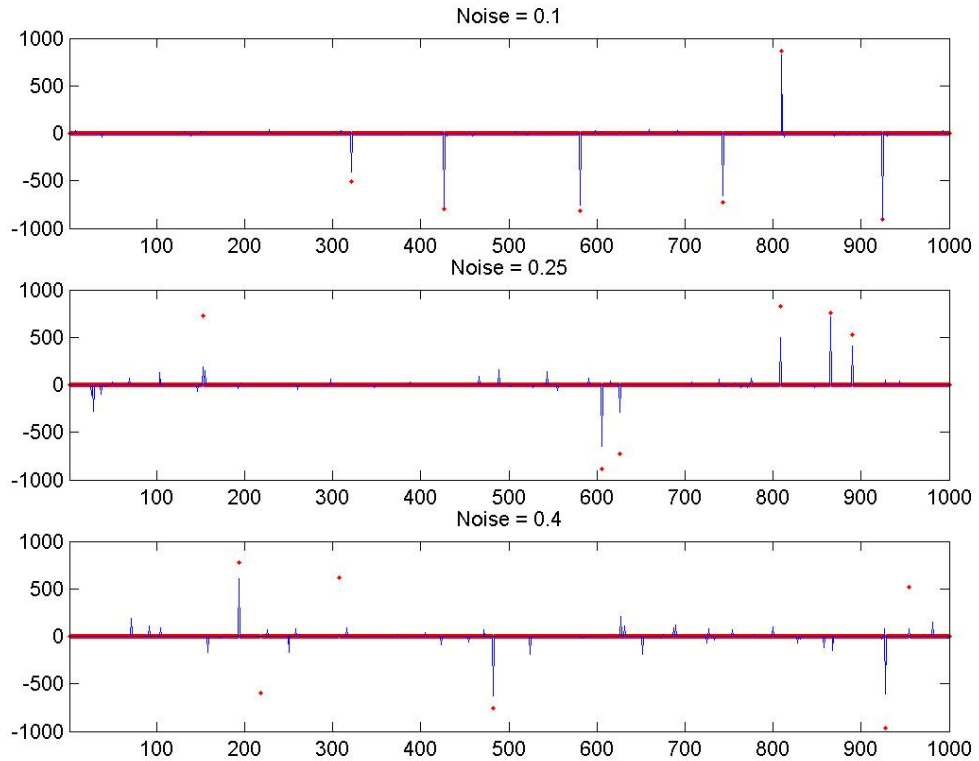


Fig. 1. The influence of increasing intrinsic noise on the identifiability. The plot shows the corresponding values of the gene-gene matrix $a \equiv \text{vec}(A)$, and increasing zero-mean Gaussian noise added to A . The red dots indicate the true value of a , and the blue line the reconstructed values a^* . For low noise levels, like 0.1, the non-zero values of a are recovered without exception. At noise level 0.4 only the largest modulus maxima values have a chance to be found.

Remark that the test results arise from computations on a normal PC as described above. When we do the same computations on a more powerful PC (dual XEON processor of 3.2 GHz and 4096 MB RAM memory) the time results will be much smaller and

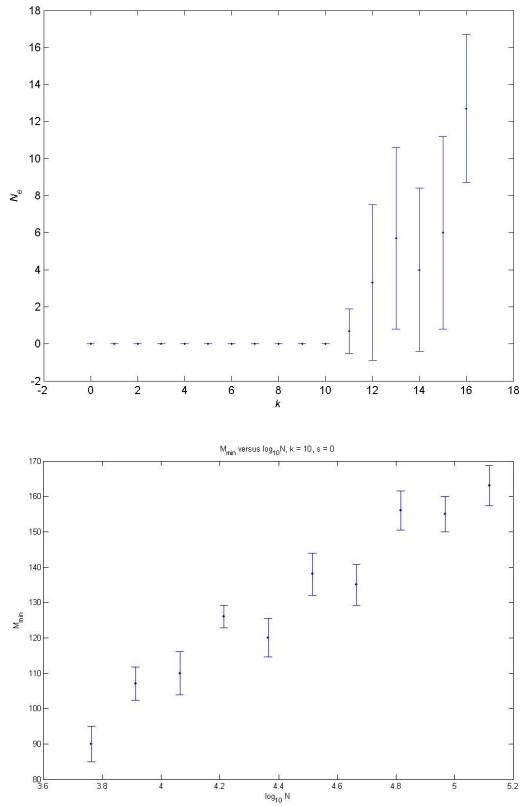


Fig. 2. *a*: Number of errors as a function of the number of nonzero entries k in x_0 , for $M = 150$, $K = 1$, $m = 5$, $N = 50000$, *b*: Dependency of the critical value M_{min} required to compute the matrix free of error versus the problem size N .

a higher number of genes can be taken into account. For example figure 7, which gives an presentation of the CPU-time in function of the number of genes, with $N = 50000$.

5 Discussion

In this work we have presented an approach for modeling and identification of gene regulatory networks from near genome wide expression profiles with a relative small amount of time instances using a piecewise linear state space model. The state space model is a rich and flexible metaphor from mathematical systems theory that applied to this case allows for hierarchical activation through master genes, representing the effects of multiple external inputs, hidden states such as none-observed genes or protein densities, and the effects of process and measurement noise. For this piecewise linear state space modeling we have presented an identification technique, based on a coupled

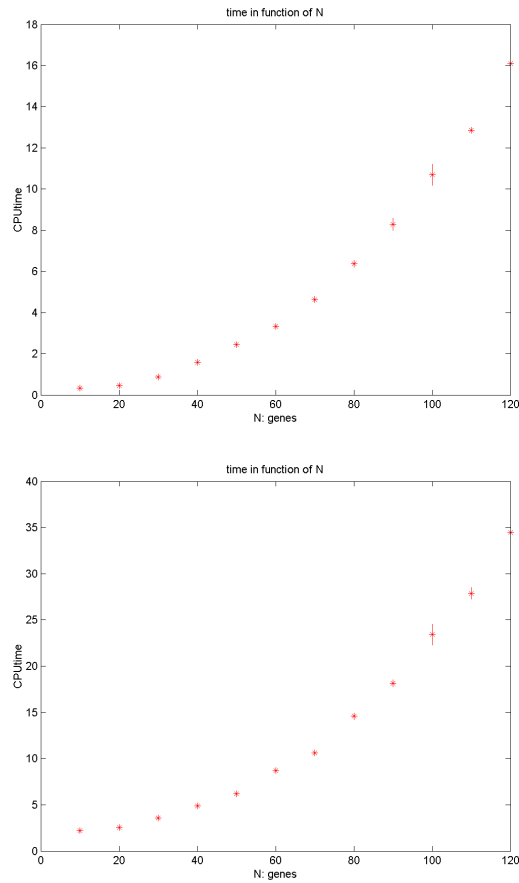


Fig. 3. CPU-time, T_c , as a function of the problem size N , for figure *a*: $M = 10$, $K = 1$ and for figure *b*: $M = 20$, $K = 3$.

set of two linear programming problems. This approach resulted in an efficient and fast algorithm that is able to accurately estimate the gene-gene coupling matrix for tens of thousands of genes based on only several hundred genome wide measurements, and that is robust towards measurement noise.

There remain a number of difficulties with regard to the system identifiability of this approach, i.e. the potential to reconstruct the interaction network from empirical data.

1. Due to the huge costs and efforts involved in the experiments, only a limited number of time points are available in the data. Together with the high dimensionality of the system, this makes the problem severely under-determined.
2. In the time series many genes exhibit strong correlation in their time-evolution, which is not per se indicative for a strong coupling between these genes but rather

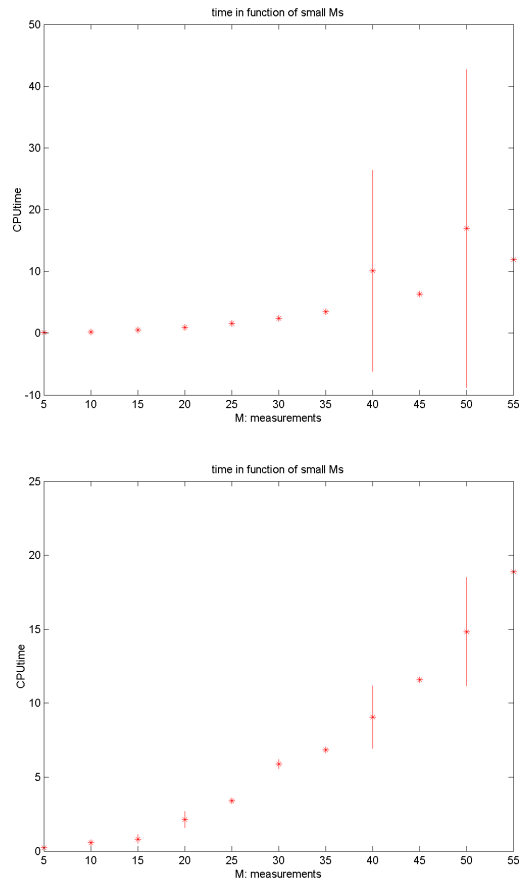


Fig. 4. CPU-time as a function of M , for figure a : $K = 1$, $N = 10$ and for figure b : $K = 3$, $N = 10$.

induced by the over-all dynamics of the ensemble of genes. This can be avoided by persistently exciting inputs.

3. Not all genes are observed in the experiment, and certainly most of the RNAs and proteins are not considered. therefore, there are many *hidden* states.
4. Because the identification techniques proposed below work on the rows, the hierarchical principle does not cause a problem, as the gene-gene interaction matrix is highly row-sparse but not column-sparse. In fact, the method utilizes the sparsity of the matrix as an implicit constraint, namely that the value of the components of the matrix should be zero.
5. Effects of stochastic fluctuations on genes with low transcription factors are severe and will obscure their true dependencies.

With this approach it is possible to reconstruct the steady states and the associated switching times of a metabolic processes from a set of micro-array experiments. In each

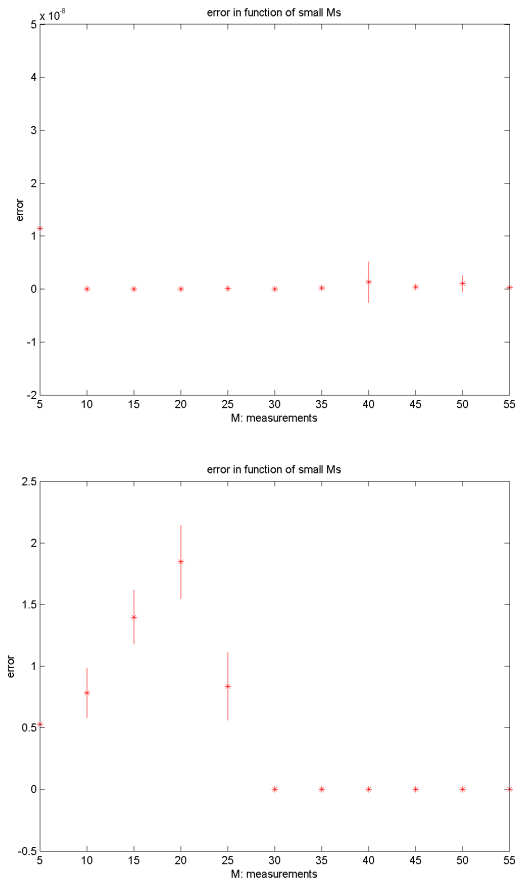


Fig. 5. Number of errors as a function of M , for figure *a*: $K = 1$, $N = 10$ and for figure *b*: $K = 3$, $N = 10$.

steady state the gene-gene interaction matrix defines the network topology. The microarray technique exhibits a strong increase in efficiency and a simultaneous decrease in associated costs. In the near future this will enable the registration of large time series of genome wide expression profiles and associated protein densities. The future availability of such data makes the further development of the mathematical modeling and associated identification of dynamic gene expression, as the approach presented here, an important condition for deducing and understanding the underlying interactions between genes and their environment.

References

1. Bower J.M., Bolouri H.(Editors), Computational Modeling of Genetic and Biochemical Networks, *MIT Press*, 2001. bibitemDavidson1999Davidson E.H. (1999), A View from the

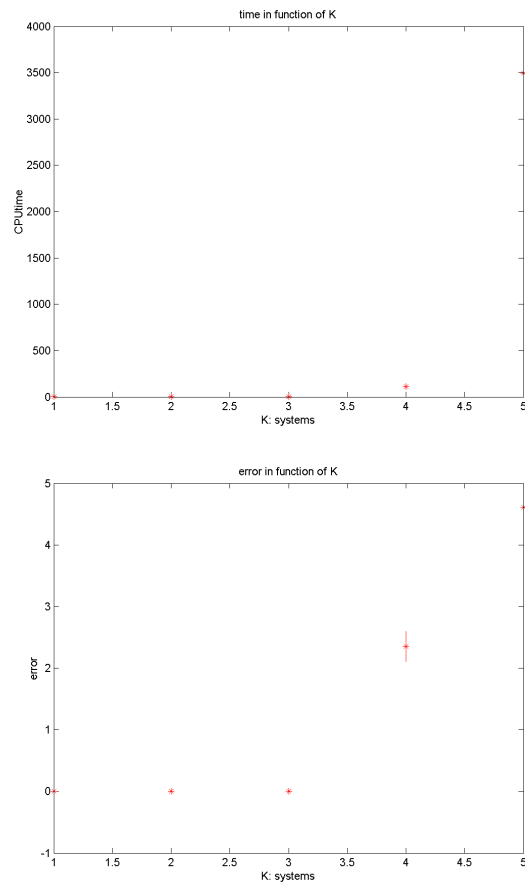


Fig. 6. *a*: CPU-time as a function of K and *b*: Number of errors in function of K with $N = 10$ and $M = 10$ for $K = 1$, $M = 20$ for $K = 2$, ...

Genome: Spatial Control of Transcription in Sea Urchin Development, *Current Opinions in Genetics and Development*, **9**, pp. 530 – 541.

2. de Jong H., Modeling and Simulation of Genetic Regulatory Systems: A Literature Review, *Journal of Computational Biology*, 2002, Volume 9, Number 1, pp. 67–103
3. de Jong H., Gouze J.L., Hernandez C., Page M., Sari T., Geiselman J., Qualitative simulation of genetic regulatory networks using piecewise-linear models, *Bull Math Biol.* 2004 Mar;66(2): pp 301–40.
4. D’haeseleer P., Liang S., Somogyi R., Genetic Network Inference: From Co-Expression Clustering to Reverse Engineering, *Bioinformatics*, vol. **16**, no. 8, 2000, pp. 707–726.
5. Elowitz M.B., Levine A.J., Siggia E.D., Swain P.S., Stochastic gene expression in a single cell, *Science*, vol. **297**, August 16, 2002, pp.1183–1186.
6. Endy, D, Brent, R. (2001) Modeling Cellular Behavior, *Nature* 2001 Jan 18; 409(6818):391-5.

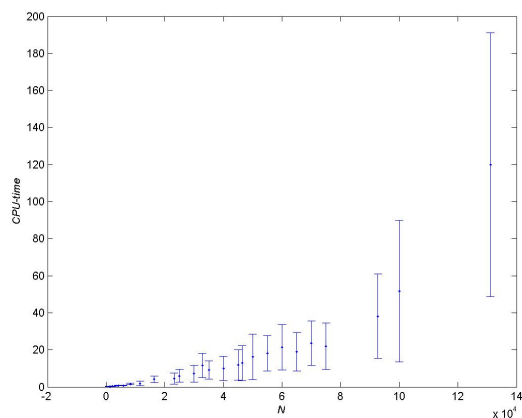


Fig. 7. CPU-time, T_c , as a function of the number of genes, with $M = 150$ and $N = 50000$.

7. Fuchs J.J. (2003), More on sparse representations in arbitrary bases, in: Proc. 13th IFAC Symp. on System Identification, Sysid 2003, Rotterdam, The Netherlands, August 27-29, 2003, pp. 1357–1362.
8. Fuchs J.J. (2004), On sparse representations in arbitrary redundant bases, IEEE Trans. on IT, June 2004.
9. Glass L., Kauffman S.A. (1973), The Logical Analysis of Continuous Non-linear Biochemical Control Networks, *J.Theor.Biol.*, 1973 Vol. 39(1), pp. 103–129
10. Goldbeter A (2002) Computational approaches to cellular rhythms. *Nature* 420, 238-45
11. Gonze D, Halloy J, and Goldbeter A (2004) Stochastic models for circadian oscillations : Emergence of a biological rhythm. *Int J Quantum Chem* **98**, pp 228–238.
12. Guthke R., Möller U., Hoffmann M., Thies F., Töpfer S., 2004, Dynamic network reconstruction from gene expression data applied to immune response, *Bioinformatics*, 2004, pp 2261
13. Hasty J., McMillen D., Isaacs F., Collins J. J., (2001), Computational studies of gene regulatory networks: in numero molecular biology, *Nature Reviews Genetics*, vol. 2, no. 4, pp. 268–279, 2001.
14. Novak B, Tyson JJ (1997) Modeling the control of DNA replication in fission yeast, *PNAS*, USA, Vol. 94, pp. 9147-9152, August 1997.
15. Peeters R.L.M., Westra R.L., On the identification of sparse gene regulatory networks, *Proc. of the 16th Intern. Symp. on Mathematical Theory of Networks and Systems (MTNS2004)* Leuven, Belgium July 5-9, 2004
16. Rosenfeld N, Young JW, Alon U, Swain PS, Elowitz MB, Gene regulation at the single-cell level, *Science* 307 (2005) pp 1962.
17. Somogyi R., Fuhrman S., Askenazi M., Wuensche A. (1997). The Gene Expression Matrix: Towards the Extraction of Genetic Network Architectures. *Nonlinear Analysis, Proc. of Second World Cong. of Nonlinear Analysis (WCNA96)* 30(3) pp 1815–1824.
18. Swain P.S., Efficient attenuation of stochasticity in gene expression through post-transcriptional control, *J Mol Biol* 344 (2004) pp 965.
19. Swain P.S., Elowitz MB, Siggia ED, Intrinsic and extrinsic contributions to stochasticity in gene expression, *PNAS* 99 (2002) pp 12795.

20. Steuer R. (2004), Effects of stochasticity in models of the cell cycle:from quantized cycle times to noise-induced oscillations, *Journal of Theoretical Biology* 228 (2004) 293-301.
21. Tegnér J., Yeung M.K.S., Hasty J., Collins J.J., Reverse engineering gene networks: Integrating genetic perturbations with dynamical modeling, *Proc. Nat. Acad. Science*, vol. **100**, no. 10, 2003, pp. 5944–5949.
22. van Kampen N. G. (1992), *Stochastic Processes in Physics and Chemistry*, Elsevier ScienceB. V., Amsterdam, (1992).
23. van Schuppen J.H. (2004), System theory of rational positive systems for cell reaction networks, CWI Report MAS-E0421, December 2004, ISSN 1386-3703
24. Verdult V., Verhaegen M., Subspace Identification of Piecewise Linear Systems, In *Proc. 43rd IEEE Conference on Decision and Control (CDC)*, pp 3838–3843, Atlantis, Paradise Island, Bahamas, December 2004.
25. Westra R.L., Peeters R.L.M. (2004), Modelling and identification of dynamical gene interactions: presentation, Workshop Intelligent Technologies for Gene Expression Based Individualized Medicine, 14th May 2004, Jena/Germany
26. Westra R.L.,(2005a), Piecewise Linear Dynamic Modeling and Identification of Gene-Protein Interaction Networks, Nisis/JCB Workshop reverse engineering, Jena, June 10, 2005.
27. Yeung M.K.S., Tegnér J., Collins J.J., Reverse engineering gene networks using singular value decomposition and robust regression, *Proc. Nat. Acad. Science*, vol. **99**, no. 9, 2002, pp. 6163–6168.