33rd Annual International Conference of the IEEE EMBS
Boston, Massachusetts USA, August 30 - September 3, 2011

An Application of Monotone Functions Decomposition to the Reconstruction of Gene Regulatory Networks

H. Chang, G. Richard, A.A. Julius, C. Belta, and S. Amar

Abstract— **We describe the reconstruction of a gene regulatory network involved with the Toll-like Receptor signaling pathways. By applying our recent identification algorithm to a time series gene expression dataset, we identify regulatory interactions between genes and construct discrete-time piecewise affine regulatory functions. Our validation shows that our model predicts the expression levels of the genes involved in the network with good accuracy.**

I. INTRODUCTION

The integration of multiple networks is an effective way of improving the accuracy of computational models. Recently, we have described the integration of the signaling network of the Toll-Like Receptor (TLR) signaling pathways [10], the metabolic network of mouse [13], and a gene regulatory network we created and its application to infection responses (Fig. 1) [12]. We describe in this paper the reconstruction of this gene network.

One of the most important challenges in systems biology is the identification of gene regulatory networks. Previous works in the field have largely been based on the analysis of gene expression data [1], [4]. This problem has been studied with control systems approaches. An identification method using the structure of piecewise affine (PWA) dynamical systems has been reported in [3], [11]. A model for identification of sparse networks using Hill functions has been developed in [2]. One of the authors has identified sparse networks based on genetic perturbation data, assuming that the dynamics can be locally described as a linear system [8], [14].

Recently, we have presented a new identification method based on monotone functions decomposition [7]. This approach assumes that each regulatory function is continuous, non-negative, and monotone. Monotonicity is a natural assumption since the notions of gene activation and repression are not well defined otherwise. The reconstruction algorithm detects regulatory relations between genes, and constructs a mathematical model for the network in the form of a discrete-time PWA system.

We apply our reconstruction procedure to the gene expression dataset from [1]. This dataset contains time series expression values for bone-marrow dendritic cells exposed to five different stimuli. In this paper we use the expression values obtained after exposure to lipopolysaccharide. Gene expression levels were measured twice at every time point of the experiment. We use the first set to identify a gene regulatory network and validate the identification procedure with the second set. Our validation shows that the reconstructed network predicts gene expression level with great accuracy.

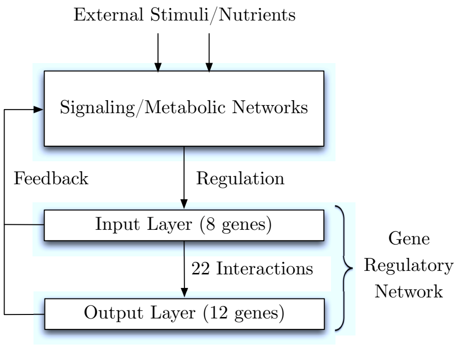

Fig. 1. Schematic representation of the integrated model from [12].

The remainder of this paper is organized as follows. In Sec. II we provide a short description of our reconstruction algorithm. Then, in Sec. III, we illustrate the reconstruction of the gene regulatory network. In Sec. IV we provide a validation of the suggested network.

II. RECONSTRUCTION ALGORITHM

In this section we briefly review our gene network reconstruction algorithm from [7]. The algorithm detects regulatory interactions between genes and constructs discrete-time PWA regulatory functions.

We assume that we are given experimental data for genes in a set $\mathcal{G}$ as time-series expression data at $N+1$ time points in the form $x_{g,n}$ where $g \in \mathcal{G}$ and $0 \leq n \leq N$. We compute the *time-difference expression data* as $q_{g,n} \triangleq x_{g,n+1} - x_{g,n}$, $g \in \mathcal{G}$, $0 \leq n \leq N-1$. The goal of the algorithm is to construct a mathematical model for the gene network dynamics that is compatible with the gene expression data. We focus on a particular class of discrete time systems of the following form

$$x_g(n+1) = x_g(n) + \sum_{k \in \mathcal{G}_g^R} f_{g,k}(x_k(n)) - \lambda_g x_g(n), \quad (1)$$

where $x_g(n)$ denotes the concentration of mRNA expressed from gene $g \in \mathcal{G}$ at time step n, $\lambda_g \geq 0$ is its decay rate, $\mathcal{G}_g^R$

This work was partially supported by grant NIH-NIDCR R01DE 015989-05A2S1.

H. Chang and C. Belta are with the Dept. of Mechanical Engineering, Boston University, Boston, MA 02215, USA {hjchang,cbelta}@bu.edu G. Richard is with the Program in Bioinformatics, Boston University, Boston, MA 02215, USA grichard@bu.edu A.A. Julius is with the Dept. of Electrical, Computer and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY 12180, USA agung@ecse.rpi.edu S. Amar is with the Dept. of Periodontology and Oral Biology, Boston University, Boston, MA 02215, USA samar@bu.edu

978-1-4244-4122-8/11/$26.00 ©2011 IEEE

is the set of regulators for gene g, and $f_{g,k}(\cdot)$ is a function describing the regulation of gene g by gene k.

It is assumed that each function $f_{g,k}(\cdot)$, $g \in \mathcal{G}$ and $k \in \mathcal{G}_g^R$, is continuous, non-negative, and monotone. Monotonicity is a natural assumption, since the notions of gene activation and repression are not well defined otherwise.

The first step in the identification of a model in the form (1) for our gene network is the construction of the regulatory sets $\mathcal{G}_g^R$, for all $g \in \mathcal{G}$. We start by sorting $x_{g,\cdot}$ in ascending order. We denote the sorted experimental data by $\hat{x}_{g,\cdot}$, $g \in \mathcal{G}$. The sorting process corresponds to the construction of a bijection $\sigma_g : \{0, \ldots, N\} \to \{0, \ldots, N\}$ such that $x_{g,n} = \hat{x}_{g,\sigma_g(n)}$ and $\hat{x}_{g,n} \leq \hat{x}_{g,n+1}$. Let $\Delta^g_{k,n} \triangleq f_{g,k}(\hat{x}_{k,n+1}) - f_{g,k}(\hat{x}_{k,n})$ for $0 \leq n \leq N-2$ and $k \in \mathcal{G}_g^R$.

In order to verify that $\mathcal{G}_g^R$ is a set of regulators for gene g, we use the following theorem.

Theorem 1 (Theorem 2 of [7]): $\mathcal{G}_g^R$ is a set of activators for gene $g \in \mathcal{G}$ such that the available experimental data is compatible with the model from (1) if and only if the following polyhedral set

$$q_{g,n} = -\lambda_g x_{g,n} + \sum_{k \in \mathcal{G}_g^R} \left(f_{g,k}(\hat{x}_{k,0}) + \sum_{l=0}^{\sigma_g(n)-1} \Delta^g_{k,l} \right)$$

$$\text{for } n \in \{0, \cdots, N-1\},$$

$$\Delta^g_{k,l} \geq 0 \text{ for } k \in \mathcal{G}_g^R \text{ and } l \in \{0, \cdots, N-2\},$$

$$\lambda_g \geq 0,$$

$$f_{g,k}(\hat{x}_{k,0}) \geq 0 \text{ for } k \in \mathcal{G}_g^R$$

$$f_{g,k}(\hat{x}_{k,0}) + \sum_{l=0}^{N-2} \Delta^g_{k,l} \geq 0 \text{ for } k \in \mathcal{G}_g^R,$$

is non-empty, where $f_{g,k}(\hat{x}_{k,0})$, $\Delta^g_{k,l}$, and λ_g for $g \in \mathcal{G}$, $k \in \mathcal{G}_g^R$, and $l \in \{0, \ldots, N-2\}$ are the variables. $x_{g,n}$ and $\hat{x}_{g,n}$ for $g \in \mathcal{G}$ and $n \in \{0, \ldots, N\}$ correspond to the experimental gene expression data.

The above theorem is based on the fact that a gene k activating g corresponds to an increasing function $f_{g,k}$. To accommodate other combinations of activator/repressor genes in the set $\mathcal{G}_g^R$, one can simply change the sign constraints for $\Delta^g_{k,l}$ and get the corresponding equivalent forms of Theorem 1.

Computationally, checking the non-emptiness of the polyhedral set in Theorem 1 involves solving a Linear Programming problem (LP) with a trivial objective function (*i.e.* 0) and the polyhedral set as constraints. From an implementation viewpoint, it is more efficient to reformulate the LP as a Linear Quadratic (LQ) programming problem based on the definition of slack variables $\epsilon := (\epsilon^f, \epsilon^\Delta)$ (see [7] for a detailed LQ formulation). MATLAB is used with the CVX [6] package to solve the LQ programming problem. A small ϵ implies that the genes in $\mathcal{G}_g^R$ correctly explain the expression data for gene g.

Finally, it is important to note that, as a by-product of Theorem 1, we get numerical values for the decay rates λ_g and the regulation functions $f_{g,k}$ at the time points corresponding to the experimental data. Given that the gene expression data is over the same time points, the latter can be easily converted to a finite number of values for each function $f_{g,k}(x_k)$. By linear interpolation of these values we construct a piecewise linear model of the form given in (1).

III. APPLICATION OF THE ALGORITHM

In this section we apply the algorithm from Section II to the experimental data from [1] to construct a mathematical model for a sparse gene network that interacts with the TLR signaling pathways. Gene expression levels were measured twice 0.5, 1, 2, 4, 6, 8, 12, 16, and 24 hours after exposure to the stimulus. We denote as $\mathcal{T}_1$ and $\mathcal{T}_2$ the first and second measurements, respectively. We employ $\mathcal{T}_1$ for the purpose of network identification in this section and $\mathcal{T}_2$ to validate the performance of the reconstructed network in Section IV.

A. Selection of gene pool

The first step for the construction of the gene network is the selection of the set of genes $\mathcal{G}$. This is, in fact, an iterative process. By using the KEGG database [9] and the TLR network, we selected a set of 17 genes that are directly regulated by transcription factors from these signaling pathways. This set of genes form the “input layer” of our gene network. Similarly, we identified a set of 49 “output layer” genes that code for proteins involved in the TLR pathways.

The identification procedure require genes whose expression values vary in time. To this end we employ the first measurement of the expression data ($\mathcal{T}_1$) and discretize it following the process detailed in [5]. We compute for each of the 66 genes previously selected their average expression level over the time series. Then, we check whether each data point lies above or below the average expression level ± 0.5 in base 2 logarithm. Thus, each point is labeled as highly-expressed, lowly-expressed, or undetermined if the expression is above, below, or within these thresholds, respectively. We consider that genes with only undetermined values are not suitable for the identification procedure, since their expression levels have very little variation.

Fig. 2 shows two examples of this discretization procedure. Gene expression data for BIRC2 and TNF are plotted with star-marks. The solid, dotted, and dashed lines correspond to the average, upper threshold, and lower threshold levels, respectively, based on the discretization previously described. All the expression values of BIRC2 lie between the dotted and dashed line, hence rendering that gene unusable. Several expression values of TNF are located above the dotted line or below the dashed line, indicating that this gene is suitable for further analysis.

This procedure reduces the input layer to 8 genes and the output layer to 31 genes. We denote as $\mathcal{G}_I$ and $\mathcal{G}_O$ these reduced input and output layers, respectively. Our main goal for the gene network reconstruction is to connect each gene from the input layer to at least one gene from the output layer through regulatory interactions, possibly using some other intermediate genes. If each gene in the input-layer regulates at least one gene in the output-layer the procedure of this section stops. Otherwise, intermediate genes are added and

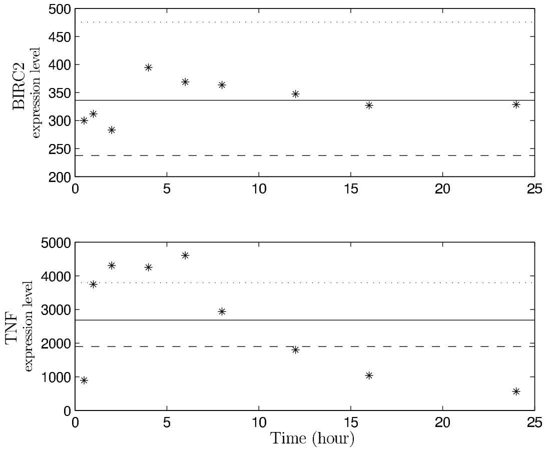

Fig. 2. Examples using the discretization of [5] for the data from [1]. The data of the genes BIRC2 and TNF is plotted with star-mark in the top and bottom graphs, respectively. The solid, dotted, and dashed lines correspond to the average, upper threshold, and lower threshold levels, respectively, for each cases. We consider gene TNF but not gene BIRC2, due to the extent of the variation of the time-series expression data.

the identification procedure is reiterated. In this case the intermediate genes are chosen through searches in KEGG and biomedical literature.

It is important to note that we only consider regulation of the output genes by (sets of) input genes. There are two main motivations for this assumption. First, the feedback from the output genes to the input genes is already captured implicitly by the TLR signaling network. Second, our plan is to use a Bayesian (probabilistic) approach for the gene network in the near future, and cycles are not allowed in Bayesian networks.

B. Application of the algorithm

We apply our identification method to $\mathcal{G} := \mathcal{G}_I \cup \mathcal{G}_O$ to search connections from the input layer to the output layer genes. We limit to two genes the set of regulators for each output gene. This assumption is biologically reasonable and also reduces the computational load.

We have to solve for each gene g in $\mathcal{G}_O$ a total of 128 $(= 2^1 C^8_1 + 2^2 C^8_2)$ LQ problems, since any combination of the 8 genes in $\mathcal{G}_I$ can regulate g, and since every gene in each regulatory set can be an activator or a repressor. To determine the best candidate regulator set, a simple method would be to select the set that corresponds to the smallest error value. However, our calculation showed that the smallest errors were very close to each other (less than 10^{-4} difference). To select a set of regulators, we proceeded to count the number of occurrences of each gene k as a regulator of the same type for gene g in the union of all sets of regulators with comparably small error. No regulator set with an error value higher than 10^{-2} was considered. We accepted k as a regulator of g if it was present in more than 45% of the candidates in this set.

C. Result of the network reconstruction

In the second iteration of the algorithm, we added two intermediate genes to the set $\mathcal{G}$. However, there was no improvement in the error values and we stopped the iteration. Fig. 3 shows the result of the reconstruction of the gene regulatory network. We can see that each gene from the input layer is connected to at least one gene from the output layer. A total of 12 output genes are linked to the input-layer.

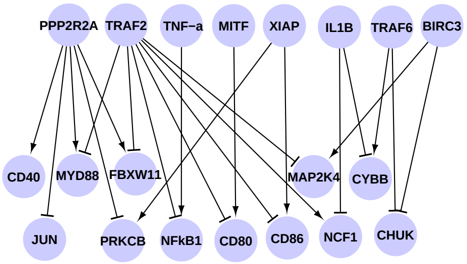

Fig. 3. Reconstructed gene regulatory network. The 8 genes in the upper region and the 12 genes in the lower region correspond to the genes in the input-layer and the output-layer, respectively.

In addition to the regulatory interactions showed in Fig. 3, we obtain discrete-time models for the genes in $\mathcal{G}_O$ controlled by genes in $\mathcal{G}_I$ as by-products of solving the LP formulation. By virtue of the LP formulation in Theorem 1 we can automatically construct the continuous non-negative monotones functions $f_{g,k}(\cdot)$ for $k \in \mathcal{G}^R_g$ as PWA functions. This is achieved by linearly connecting the data in the $(x, f_{g,k}(x))$ plane. Thus we can obtain discrete-time PWA models for 12 genes in $\mathcal{G}_O$.

Note that the connections in our gene network do not necessarily imply direct relations in terms of biological regulations since we only depend on the computational method described in Section II. A connection in the network shows a causal relation inferred from gene expression data.

IV. Validation of the Model

In this section we validate the performance of the model constructed in Section III. The second expression measurement ($\mathcal{T}_2$) is employed throughout this section.

A. Validation based on one-step predictions

For each gene $g \in \mathcal{G}_O$, we compare the experimental expression values with predictions made with our model. The experimental values $x_{g,n}$ and $x_{k,n}, k \in \mathcal{G}^R_g$ from $\mathcal{T}_2$ are used in (1) to generate $x_g(n+1)$. The predicted value is then compared with $x_{g,n+1}$.

Table I provides the relative error values between experimental and predicted expressions $(|x_g(n) - x_{g,n}|/x_{g,n})$. About 70% of all relative errors are below 0.20 and only 7% of them are higher than 0.50. The average relative error obtained for all predictions reaches 0.26. These results indicate that our PWA model generally predicts expression values correctly.

TABLE I

RELATIVE ERROR VALUES BETWEEN EXPERIMENTAL AND PREDICTED EXPRESSIONS AT EACH TIME STEP. EXPERIMENTAL VALUES AT TIME STEP n ARE USED IN (1) TO COMPUTE PREDICTIONS AT TIME $n+1$.

Time steps	CD80	CD86	CHUK	CYBB	JUN	MYD88	NCF1	NFκB1	PRKCB	CD40	MAP2K4	FBXW11
1 (1 h)	0.46	0.12	0.15	0.08	0.29	0.04	0.03	0.01	0.33	6.57	0.35	0.85
2 (2 h)	0.44	0.39	0.23	0.18	0.55	0.17	0.25	0.20	0.43	0.78	0.72	0.52
3 (4 h)	0.21	0.30	0.01	0.19	0.12	0.18	0.29	0.13	0.11	0.13	0.19	0.28
4 (6 h)	0.25	0.10	0.20	0.07	0.02	0.18	0.09	0.09	0.12	0.06	0.04	0.36
5 (8 h)	0.21	0.08	0.09	0.03	0.28	0.10	0.02	0.11	0.06	0.12	0.00	0.41
6 (12 h)	0.31	0.18	0.06	0.21	0.13	0.05	0.30	0.09	0.18	0.04	0.03	0.19
7 (16 h)	0.18	0.01	0.12	0.21	0.03	0.16	0.09	0.05	0.28	0.93	0.15	0.04
8 (24 h)	0.07	0.12	0.13	0.16	0.34	0.02	0.05	0.12	0.01	0.26	0.18	0.05
Average	0.27	0.16	0.12	0.14	0.22	0.11	0.14	0.10	0.19	1.11	0.21	0.34

B. Validation based on time course simulations

We perform simulations with (1) to test the performance of our model. We consider a simulation time step of 30 min. For each gene $g \in \mathcal{G}_O$, we initialize $x_g(0)$ with $x_{g,0}$ (*i.e.* at 0.5 hour) and similarly all $x_k(0)$ with $x_{k,0}, k \in \mathcal{G}_g^R$. Values for $x_g(n+1)$ are determined with (1) using $x_g(n)$ and $x_k(n)$. Values for $x_k(n)$ are reinitialized at each time step from the experimental data. If no experimental value exists – we only have measurements at 0.5, 1, 2, 4, 6, 8, 12, 16, and 24 hours after stimulus – we derive a value by linear interpolation.

Fig. 4 shows the simulations obtained for every gene of the output layer. Simulated and experimental data (from $\mathcal{T}_2$) are plotted with dotted lines and star-marks, respectively. In most cases, the simulation follows closely the trend given by the experimental data. The model is able to capture accurately the dynamic of gene expression.

V. CONCLUSIONS

We described the reconstruction of a gene regulatory network related to the TLR pathways. We applied our recent identification algorithm to a gene expression dataset, and determined regulatory interactions between genes. The connections in our gene network implies causal relations conjectured from gene expression data. Predictions made with our model agree with experimental data.

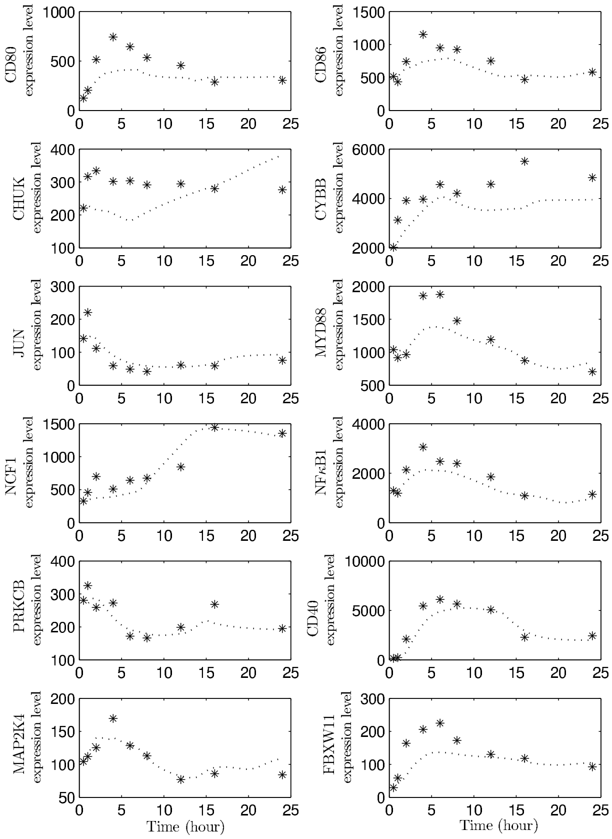

Fig. 4. Time course simulations of the network models constructed in Section III. Simulated and experimental data (from $\mathcal{T}_2$) are plotted with dotted lines and star-marks, respectively.

REFERENCES

[1] I. Amit et al. Unbiased reconstruction of a mammalian transcriptional network mediating pathogen responses. *Science*, 326(5950):257–263, October 2009.

[2] E. August and A. Papachristodoulou. Efficient, sparse biological network determination. *BMC Systems Biology*, 3(1):25, 2009.

[3] S. Drulhe et al. The switching threshold reconstruction problem for piecewise-affine models of genetic regulatory networks. *IEEE Transactions on Automatic Control*, 53(Special Issue):153 –165, 2008.

[4] J. Faith et al. Large-scale mapping and validation of Escherichia coli transcriptional regulation from a compendium of expression profiles. *PLoS Biol*, 5(1):e8, Jan 2007.

[5] N. Friedman et al. Using bayesian networks to analyze expression data. *Journal of Computational Biology*, 7:601–620, 2000.

[6] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 1.21. http://cvxr.com/cvx, February 2011.

[7] A. Julius and C. Belta. Genetic regulatory network identification using monotone functions decomposition. *18th IFAC World Congress*, Milan, Italy, 2011.

[8] A. Julius et al. Genetic network identification using convex programming. *IET Systems Biology*, 3(3):155–166, 2009.

[9] M. Kanehisa et al. KEGG for linking genomes to life and the environment. *Nucleic Acids Res*, 36(Database issue):D480–4, 2008.

[10] F. Li et al. Identification of potential pathway mediation targets in Toll-like receptor signaling. *PLoS Comput Biol*, 5(2):e1000292, 2009.

[11] R. Porreca et al. Structural identification of piecewise-linear models of genetic regulatory networks. *Journal of Computational Biology*, 15(10):1365–1380, 2008.

[12] G. Richard et al. Integration of large-scale metabolic, signaling, and gene regulatory networks with application to infection responses. submitted to *50th IEEE CDC and ECC*, 2011.

[13] M. Sigurdsson et al. A detailed genome-wide reconstruction of mouse metabolism based on human recon 1. *BMC Syst Biol*, 4(1):140, Oct 2010.

[14] M.M. Zavlanos et al. Identification of stable genetic networks using convex programming. In *Proc. of the 2008 IEEE American Control Conference*, pages 2755 –2760, 2008.