International Journal of Biomathematics
Vol. 11, No. 3 (2018) 1850032 (13 pages)

DOI: 10.1142/S1793524518500328

World Scientific
www.worldscientific.com

Dynamics of gene regulatory networks with stochastic propensities

O. Akman*,¶, T. Comar†, A. L. Harris‡,
D. Hrozencik§ and Y. Li*

*Department of Mathematics, Illinois State University
Normal, IL, USA

†Department of Mathematics, Benedictine University
Lisle, IL, USA

‡Department of Physics, Illinois State University
Normal, IL, USA

§Department of Mathematics, Chicago State University
Chicago, IL, USA
¶oakman@ilst.edu

Received 2 February 2017
Accepted 2 January 2018
Published 7 March 2018

Gene regulatory networks (GRNs) control the production of proteins in cells. It is well-known that this process is not deterministic. Numerous studies employed a non-deterministic transition structure to model these networks. However, it is not realistic to expect state-to-state transition probabilities to remain constant throughout an organism's lifetime. In this work, we focus on modeling GRN state transition (edge) variability using an ever-changing set of propensities. We suspect that the source of this variation is due to internal noise at the molecular level and can be modeled by introducing additional stochasticity into GRN models. We employ a beta distribution, whose parameters are estimated to capture the pattern inherent in edge behavior with minimum error. Additionally, we develop a method for obtaining propensities from a pre-determined network.

Keywords: Molecular noise; network simulation; beta distribution.

Mathematics Subject Classification 2010: 92Bxx

1. Introduction

Genes operate in complex networks, which control fundamental cellular processes. To better understand the highly integrated regulatory mechanisms used in such processes, scientists and engineers alike have developed a variety of mathematical and computational models. The accuracy of such models is important for predicting the behavior of the system and evaluating potential experimental alterations of its

function. A vast literature exists describing the wide variety of approaches used to model these networks [7, 18].

Some of the typical attempts at modeling gene regulatory networks (GRNs) look at protein production as a two-step process: transcription followed by translation. Since DNA cannot pass through the membrane of the nucleus, the first step in protein production is *transcription*, the process by which DNA is copied onto messenger RNA (mRNA). The mRNA then passes through the nuclear membrane into the cellular cytoplasm where it is read by a ribosome and a protein is produced; this step is called translation. To model this process, mathematicians have used systems of differential equations, with two equations for each gene in the network: one describing the transcription process, and one describing the translation process. These equations model changes in the concentration of mRNA and changes in protein concentration, respectively. Such a model can be seen in [10]. These models, while generally valuable, have numerous parameters, requiring extensive experimental data for parameter estimation before yielding useful quantitative information about the network. For this reason, researchers have investigated whether they can obtain qualitative information about the dynamics of GRNs without the need for extensive experimental data for parameter estimation. This motivated Kauffman [12] to develop discrete Boolean GRN models. A discrete Boolean model of a GRN consists of a wiring diagram and rules for updating the states of the network. The wiring diagram is a graph with each gene being represented by a node and each biological interaction (activation or inhibition) being represented by a connection. Each gene is either off (state = 0) or on (state = 1). Given any initial state of the system, we can use the updating rules to explore the dynamics of the system. It has been shown that under the appropriate conditions, these discrete Boolean networks yield valuable qualitative information about the GRN in question [3, 6, 11, 15, 17, 20–23].

The models described above are deterministic, in that knowledge of the current system state and the updating rules allow one to know the system state at any time. There is significant evidence [4, 5, 8, 13, 24], however, that GRNs do not operate in this fashion due to possible external or internal noise. First, there is biological uncertainty. Second, there may be experimental noise due to the complex measurement process, ranging from hybridization conditions to microarray image processing techniques. Third, there may be interacting latent variables, such as proteins, environmental conditions, or other genes that are not measured [19]. The question then becomes: how do we account for these types of noise in our models?

Murugarra, *et al.* [16] propose using a stochastic model for regulation in the p53-Mdm2 network as a model to describe the dynamics of the tumor suppressor protein p53 and its negative regulator Mdm2 when DNA damage occurs. DNA damage caused by ionic irradiation decreases the level of nucleic Mdm2, which enables p53 to accumulate and to remain active, playing a key role in reducing the effect of the damage. They argue that stochasticity is needed at the biological function level because even ideal circumstances do not guarantee activation or degradation as expected, as there is a probability that the process will not occur due to possible

internal noise. In essence, the activation and deactivation processes between genes are subject to propensities that represent the frequency of the times that connections function as expected. Then, these propensities govern the probability that a given state transitions to another state. These transition probabilities are direct results of the propensities of the originating gene interactions. Section 2.1.1 illustrates with a two-gene network.

Although stochasticity in gene expression is generally believed to be detrimental to cell function since fluctuations in protein levels can corrupt the quality of intracellular signals, there are possible benefits where the stochasticity provides a mechanism for phenotypic and cell-type diversification. In fact, the role of increased or decreased noise on the evolutionary process is rather advantageous. For instance, switching between phenotypic states with varying growth rates might be an important factor in the phenomenon of persistent bacterial infections after treatment with antibiotics, [2, 14]. Stochastic transitions between distinct phenotypic states have also been observed in *S. cerevisiae* [1], where three feedback-control loops, one negative and two positive, form the galactose-utilization network. For certain types of cancer, there exists a key pair of oncogene and tumor suppressor genes. In seemingly identical cellular environments, certain cells become cancerous while their neighbors do not. Phenomena such as this require an approach that provides a possible explanation for how this occurs.

In this paper, we attempt to quantify the impact of internal noise in GRNs in terms of state-to-state transition variability. In the literature, several ways of modeling internal noise have already been tried. Early efforts to introduce randomization simulated a coin flip for each gene in order to determine whether or not to change its state. This approach overstates the random effects of noise in the system. Probabilistic Boolean networks (PBNs) are another way to introduce stochasticity into GRNs. In these models a single update function is generalized to two or more updating functions. With a fixed probability, a function is chosen and is used to update the nodes in the network. At each step, there is then a propensity that a given gene will not be updated according to the deterministic rule. This accounts for differences in the behaviors of various cells or molecules under the identical conditions. With this approach, one can produce a probabilistic state space showing the probabilities of transitioning between the 2^n different states of an n-gene network.

Although the approach used in [16] is a logical initial step, it has several shortcomings. First, it seems highly unlikely that activation/deactivation propensities will be constant. Second, due to variability both at the cellular and molecular levels, it is more likely that these propensities vary randomly around an average that describes the rate at which genes function according to an ideal deterministic structure [9, 25].

Here we present a method to quantify the transition variability under stochastic variations of propensity probabilities. This represents the most significant difference between our work and [16]. With this work, we lay the foundation for biologists to design experiments where edge variations are crucial. Furthermore, we hypothesize

that edge variability represents a previously overlooked aspect of biological stochasticity.

The organization of this paper is as follows. In Sec. 2, we develop a model describing the cellular level variability. The parameter estimation of the model is implemented in Sec. 3. Concluding remarks are given in Sec. 4.

2. Stochastic Gene Regulatory Networks

2.1. *Model development*

In this section, we develop a stochastic model for random propensity probabilities.

Let $G_1, G_2, \ldots, G_n$ represent genes in a regulatory network. Also let $x_1(t)$, $x_2(t), \ldots, x_n(t)$ be the state of each gene at time t, where $x_i(t) = 0$ if the gene is off at time t, and $x_i(t) = 1$ if the gene is on at time t.

Denote all possible states of gene G_i by set X_i, $i = 1, 2, \ldots, n$. In our case, $X_i = \{0, 1\}$, $i = 1, 2, \ldots, n$. Then the state space of this GRN, X, is the Cartesian product of each potential space of each gene, respectively. Mathematically, $X = X_1 \times X_2 \times \cdots \times X_n$.

Let $f_i : X_i \to X_i$ be the updating function for gene G_i, $i = 1, 2, \ldots, n$. The activation (degradation) propensity $p_i^{\uparrow}, p_i^{\downarrow}$ is the probability that the gene G_i will be activated (degraded) according to the updating rule. Note that $p_i^{\uparrow}, p_i^{\downarrow}$ in $[0, 1]$ because they are probabilities. Also, let $p_i^{\uparrow}$ and $p_i^{\downarrow}$ $(p_i^{\uparrow}, p_i^{\downarrow} \in [0, 1])$ be the activation and degradation propensities, respectively, for gene G_i.

Now denote the propensity matrix by

$$P = \begin{vmatrix} p_1 & \cdots & p_{2n-1} \\ p_2 & \cdots & p_{2n} \end{vmatrix},$$

where p_{2i-1} is the propensity of G_i's activation and p_{2i} is the propensity of G_i's deactivation, $i = 1, 2, \ldots, n$. Once regulatory rules and the propensities are given, then the transition matrix of the network is known. We denote transition matrix by T, which is a $2^n \times 2^n$ matrix.

2.1.1. *An example*

Let $n = 2$, $X = \{0, 1\} \times \{0, 1\}$, $F = \{f_1, f_2\} : X \to X$, where Table 1 represents the update rule for G_1 and G_2 and propensity matrix

$$P = \begin{vmatrix} 0.1 & 0.5 \\ 0.2 & 0.9 \end{vmatrix}.$$

Table 1 shows how the GRN would work if the system were deterministic, e.g., if the system were ever in state 01, then it would be in state 10 after one time step with 100% certainty. There is ample evidence, however, that the system does not work deterministically. Variability at the cellular and molecular levels require us to utilize stochastic elements to model a typical GRN. This is where the propensities are used. For example, for the system to move from state 01 to state 10, gene 1 would

Table 1. Example table for update rules.

G_1	G_2	f_1	f_2
0	0	0	0
0	1	1	0
1	0	0	1
1	1	1	0

Table 2. Transition matrix of two-GRN example.

		Output			
		00	01	10	11
Input	00	1	0	0	0
	01	0.81	0.09	0.09	0.01
	10	0.1	0.1	0.4	0.4
	11	0	0	0.9	0.1

need to be activated and gene 2 would need to be deactivated. The probability that gene 1 is activated is 0.1, its activation propensity and the probability that gene 2 is deactivated is 0.9, its deactivation propensity. To compute the probability of both occurring we would need to multiply these probabilities together. Then:

$$\Pr(01 \to 10) = 0.1 \times 0.9 = 0.09, \quad \Pr(01 \to 00) = (1 - 0.1)(0.9) = 0.81,$$

$$\Pr(01 \to 01) = (1 - 0.1)(1 - 0.9) = 0.09, \quad \Pr(01 \to 11) = (0.1)(1 - 0.9) = 0.01,$$

$$\Pr(10 \to 10) = (1 - 0.2)(1 - 0.5) = 0.4, \quad \Pr(10 \to 01) = 0.2 \times 0.5 = 0.1,$$

$$\Pr(10 \to 00) = 0.2 \times (1 - 0.5) = 0.1, \quad \Pr(10 \to) = (1 - 0.2) \times 0.5 = 0.4,$$

$$\Pr(11 \to 11) = 1 \times (1 - 0.9) = 0.1, \quad \Pr(11 \to 10) = 1 \times 0.9 = 0.9,$$

$$\Pr(00 \to 00) = 1 \times 1 = 1.$$

The transition matrix is given in Table 2. For the details of these calculations we refer the reader to [16].

In this study, we propose that $p_1, p_2, \ldots, p_{2n}$ be distributed according to a beta distribution $B(\alpha, \beta)$ and we assume that the regulatory rule for the network is known. Hence, an estimated network can be constructed by estimating the parameters, α and β, of the beta distribution.

3. Parameter Estimation

We now consider the beta distributions, $B(\alpha^{(ij)}, \beta^{(ij)})$, as random processes that $\{p_{ij}\}$ follow. Assuming the regulatory rule for the network is known, we choose $(\alpha^{(ij)}, \beta^{(ij)})$ to identify a distribution from which $\{p_{ij}\}$ is drawn. Hence the choice of α and β generates stochastic propensities, which in turn are used to calculate the edge probabilities of the network. Here the use of the beta distribution is appropriate

since its domain is $[0, 1]$, Additionally, it can accommodate a wide spectrum of propensity behaviors due to its broad range of shapes.

Our goal is to measure the variability hidden in a given network where the propensities reflect the average rate of stochastic activation/degradation behavior. We achieve this by simulating networks generated by a random set of propensities, as described above, in such a way that the difference (error) between the generated network and the fixed network is minimum. Forthwith a simulated network will be referred to as a *rebuilt network*, and the network generated by the fixed propensities will be referred to as a *fixed network*.

We define the error of estimation using the normalized difference between a fixed network and its corresponding rebuilt network

$$E = \frac{1}{m} \sum_{i,j=1}^{2^n} q_{ij}^2, \tag{1}$$

where q_{ij} is the element from the ith-row and jth-column of $T_0 - T$, T_0 and T are the transition matrices of fixed and rebuilt networks, respectively, and m is the total number of non-constant edges in the network. This error can be interpreted as the criterion for assessing the quality of the network estimation.

We illustrate our method with a two-GRN in Sec. 2.1.1. To ensure the validity of our method, we consider four cases that represent a wide range of possible gene activation/degradation behaviors. Such a network is depicted in Fig. 1.

- *High propensities*: All propensities are close to 1. This implies that the network is functioning efficiently. Every element is performing as expected most of the time.
- *Low propensities*: All propensities are close to 0. This implies that the network is functioning inefficiently. Every element is unlikely to perform properly.
- *Medium propensities*: All propensities are close to 0.5. This implies that every element's action is as likely to succeed as to fail.
- *Spread propensities*: Propensities have a wide range. This implies some elements are acting efficiently, while others are flawed.

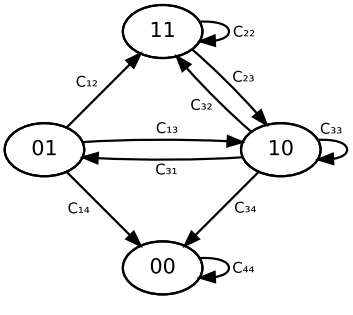

Fig. 1. Diagram of the example network from Sec. 2.1, including the nodes and edges.

With this approach, one can model, for instance, a network with high propensities using a left skewed beta distribution. Similarly, using a uniform distribution, one can model a network with spread propensities where propensity values are almost equally likely.

3.1. *Estimation of a network from a propensity matrix*

Starting with a given $n \times 2$ propensity matrix, we obtain the parameters of the $2n$ beta distributions whose means are equal to the known propensities p_{ij}. More specifically, a fixed variance v is chosen so that the fitted beta distribution parameters are obtained as given below:

$$\alpha = p_{ij}\big(p_{ij} - p_{ij}^2 - v\big)/v, \tag{2}$$

$$\beta = (1 - p_{ij})\big(p_{ij} - p_{ij}^2 - v\big)/v. \tag{3}$$

Then, for a predetermined propensity variance, we simulate k random variates from each distribution producing k networks. Using these simulated networks, we find the error of estimation of the kth-network using (1). Repeating this process sufficiently many times, we find an average error of estimation as a function of the chosen propensity variance. As an example, we choose four sets of propensities reflecting various gene behavior, as shown below. Particularly, the propensities represent high, medium, and low probability of activation/deactivation failure. As a test case, we also included a set where the propensities are chosen randomly from a uniform distribution. Implementing the process as explained above, we note that average error of estimation increases with increasing propensity variance, as expected:

$$\text{Network 1, Spread propensity} \quad \begin{vmatrix} 0.1 & 0.5 \\ 0.2 & 0.9 \end{vmatrix},$$

$$\text{Network 2, Medium propensity} \quad \begin{vmatrix} 0.3 & 0.4 \\ 0.2 & 0.6 \end{vmatrix},$$

$$\text{Network 3, Low propensity} \quad \begin{vmatrix} 0.1 & 0.05 \\ 0.15 & 0.1 \end{vmatrix},$$

$$\text{Network 4, High propensity} \quad \begin{vmatrix} 0.9 & 0.8 \\ 0.95 & 0.7 \end{vmatrix}.$$

Note that care must be taken in choosing the value of v since α and β are positive-valued parameters. Naturally, this limits the values of v to $v \geq p_{ij}(1-p_{ij})$.

Here we are interested in investigating the variability of different edges of a given network under varying propensities. Specifically, let $p_{ij} \sim B(\alpha^{(ij)}, \beta^{(ij)})$ with $E(P_{ij}) = p_{ij}$, where $\alpha^{(ij)}, \beta^{(ij)}$ represent the parameters of the beta distribution

from which p_{ij} is drawn. For notational simplicity, we will suppress the superscripts i, j, from here on.

In order to generate k networks, we set p_{ij} equal to the mean of the beta distribution

$$p_{ij} = \frac{\alpha}{\alpha + \beta}. \tag{4}$$

We also choose a value for the variance v and let

$$v = \frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)}. \tag{5}$$

Then the α and β estimates are obtained by solving the above equations. Given the estimated values of α and β, we randomly generate k networks. Using these k values for each edge, we calculate a variance that represents the variation of the network under the stochastic behavior of the propensities.

Figure 2 depicts 95% confidence intervals for each edge of the sample networks listed above. Here we note that:

- each edge displays a variability, and
- the variability differs from edge-to-edge.

The figure shows that edge variation depends on the propensities used in generating that particular network. Comparing the four different networks, we observe that when the edge values of two networks differ by an order of magnitude, the widths are also very different. In particular, the larger the edge value, the larger the width. However, when the edge values are of the same order of magnitude, the widths are similar.

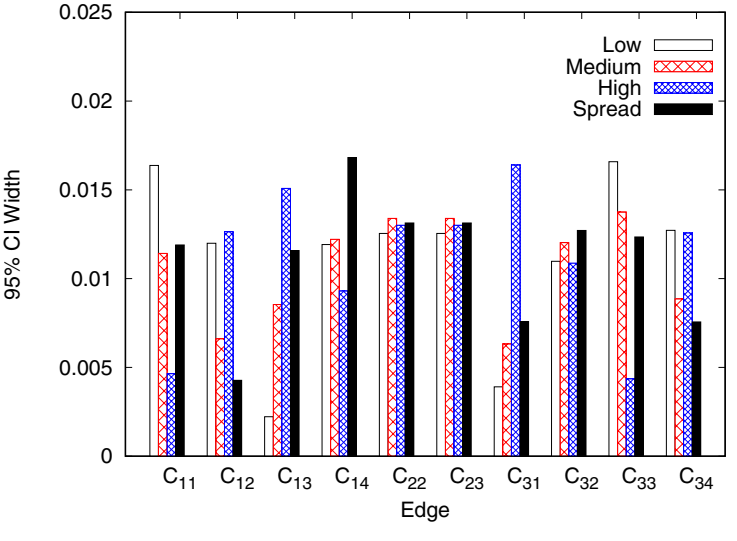

Fig. 2. 95% confidence interval width for each edge in the four-sample networks used here.

For example, compare the edge C_{13} for the high and low propensity networks. For the high propensity network, $C_{13} = 0.63$, while for the low propensity network, $C_{13} = 0.01$. Figure 2 shows their widths as 0.015 (high) and 0.002 (low). Alternatively, if one examines C_{22} for the high and low propensity networks, they have similar values of 0.3 and 0.9, respectively. In this case, their widths are also similar.

The edge variation estimated here extends the results obtained in [16]. The assumption of static propensities in [16] prohibits edge variation in their model, which in return restricts applicability of the model in realistic scenarios. For instance, the inherent variation due to varying propensity behavior potentially could be a contributing factor to the onset pathological conditions. Consequently, in medical studies, edges with high variation may be the natural candidates for faulty network performance and hence should be the focus for further investigation.

A reasonable next step is to investigate the impact of propensity variation on the individual edge variation. The variation of propensities in relation to the average variation per edge of an entire network is depicted in Fig. 3. As expected, the figure shows that increased variation in propensity leads to increased average network variation. We note that the relationship between the average network variation and the propensity variation is linear with positive slope, hence as the variation in each propensity increases, the overall network variance, and thus the average edge variation, should increase.

3.2. *Estimation of propensity matrix from network*

We now aim to find the propensity matrix that generates a given network using only the edge values of the network. In certain cases we may know the strength of

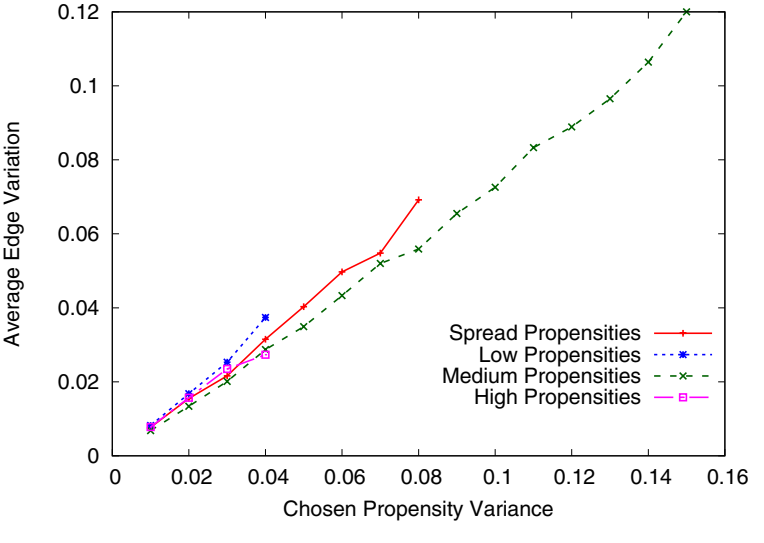

Fig. 3. Average edge variation as a function of the chosen propensity variance for the four-sample networks used here.

connections, however this does not necessarily provide us the variability of the edges. By simulating the given network we can construct an estimated network which provides us an approximate edge variability. This can be achieved by reconstructing the network, from which we can calculate the variability of the edges that may provide clues as to where the network is most likely to malfunction. Here we list the necessary steps to accomplish this:

(1) Begin with a fixed network.
(2) Simulate a rebuilt network as described below.
(3) Measure the overall variance of the rebuilt network from the fixed network as the sum of the squares of the individual edge differences.
(4) Repeat Step (3) to obtain the best rebuilt network that most closely resembles the fixed network. Obviously the lower the overall variance, the more closely the rebuilt network will resemble the fixed network.

Below we describe the network generation in detail.

3.2.1. *Network generation*

To begin, $2n$ beta distributions are generated using randomly-chosen values of alpha and beta. Then, a single value is chosen from each of the beta distributions to represent each of the activation and degradation propensities. Using these propensities, the transition matrix of the rebuilt network is obtained. This process is repeated N times, and the set of propensities with the smallest overall network variance is stored. Next, the entire process is repeated M times, resulting in an ensemble of M "best" propensity estimates. From this ensemble of M propensity values, an average value is calculated for each of the activation and degradation propensities. One way to determine if the method is successful is to compare the average propensities with the known values that created the fixed network. As examples, we use the same four fixed networks listed in Table 2. The average propensity values of the rebuilt networks as a function of overall network variance for each of the four trial networks are shown in Table 3.

Figure 4 depicts excellent agreement between the predicted and known propensity values. As expected, when the overall network variance increases, the predicted propensity values tend to deviate more from the fixed values. Currently, we have included all $M(=100)$ predicted propensity values in the calculated average, but one could improve the prediction by limiting the values used in the average to only those with an overall network variance below some small cutoff value. Recall that when the propensity values are low, the network is functioning inefficiently. We see this reflected in Fig. 3, in which the spread of the predicted propensities is large when the value of the propensity is small. Also, as verified by the simulation, the predicted propensities are close to the true values, indicating that it is possible to determine the propensities from the network edge values for spread, low, medium, and high propensity networks.

Table 3. Average values for the predicted propensities of the rebuilt networks that are shown in Fig. 4, along with their standard deviations. Also included are the known values of the propensities that created the fixed network.

		P_1	P_2	P_3	P_4
High	Ave	0.922	0.972	0.797	0.696
	SD	0.0472	0.0343	0.0341	0.0259
	Known	0.9	0.95	0.8	0.7
Medium	Ave	0.306	0.197	0.408	0.6
	SD	0.0704	0.0775	0.0633	0.0395
	Known	0.3	0.2	0.4	0.6
Low	Ave	0.0908	0.15	0.0275	0.101
	SD	0.0368	0.0342	0.0313	0.0251
	Known	0.1	0.15	0.05	0.1
Spread	Ave	0.0905	0.194	0.503	0.903
	SD	0.0513	0.0556	0.0504	0.0271
	Known	0.1	0.2	0.5	0.9

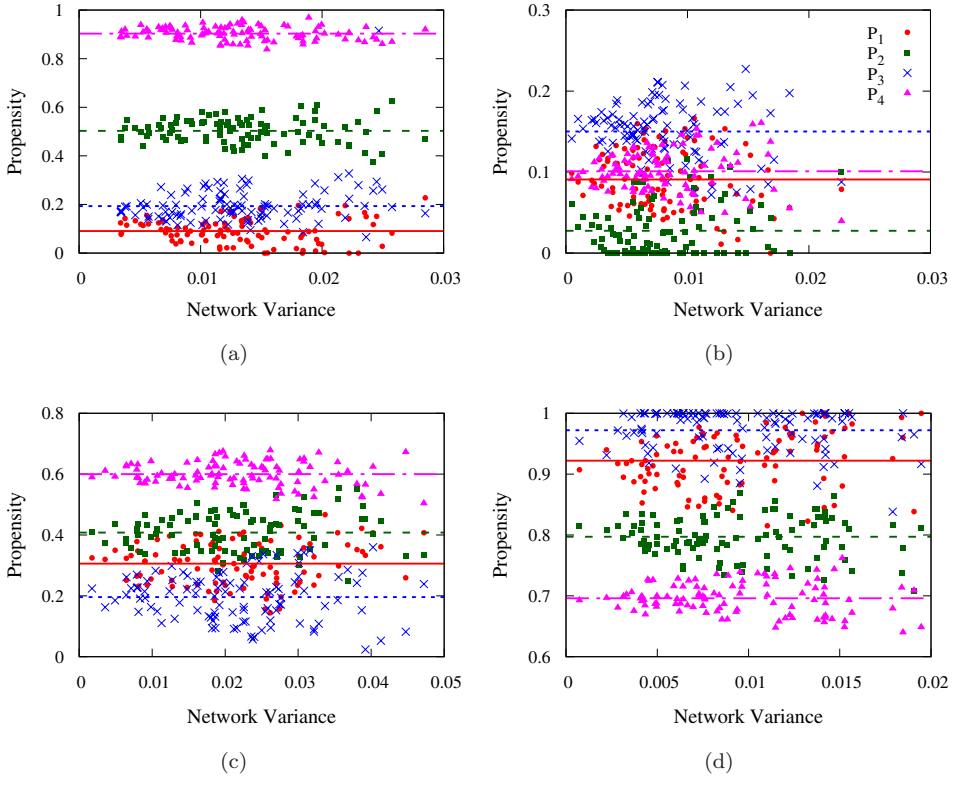

Fig. 4. Predicted propensity values for M rebuilt networks (points), and their average values (lines). Results are shown for the four networks discussed here: (a) spread propensity, (b) low propensity, (c) medium propensity, and (d) high propensity.

4. Conclusion

Building upon previous PBNs, our approach allows us to accommodate non-constant activation/degradation propensities. In particular, with this approach we provide a viable method of obtaining network variability starting from a set of fixed propensities. Moreover, our methods also allow us to reverse the process to arrive at propensities starting from the knowledge of network connection strengths. In fact, it is due to our approach that we were able to observe variations of different magnitude among the edges of a network.

Needless to say, creative use of statistical distributions is an essential component of successfully modeling non-constant gene propensities. This method alleviates the difficulty, due to a small number of propensities or connections, faced by using traditional statistical methods such as maximum likelihood or method of moments estimation.

Acknowledgments

The authors thank the anonymous referees whose comments helped improve the manuscript. The authors are also grateful to the Center for Collaborative Studies in Mathematical Biology of Illinois State University for its support, making this a successful collaboration. A. L. Harris gratefully acknowledges the support of the NSF under Grant No. PHY-1505217.

References

[1] M. Acar, A. Becskei and A. van Oudenaarden, Enhancement of cellular memory by reducing stochastic transitions, *Nature* **435**(7039) (2005) 228–232.
[2] A. Aertsen and C. W. Michiels, Stress and how bacteria cope with death and survival, *Crit. Rev. Microbiol.* **30**(4) (2004) 263–273.
[3] R. Albert and H. G. Othmer, The topology of the regulatory interactions predicts the expression pattern of the segment polarity genes, in *Drosophila melanogaster*, *J. Theor. Biol.* **223**(1) (2003) 1–18.
[4] A. Arkin, J. Ross and H. H. McAdams, Stochastic kinetic analysis of developmental pathway bifurcation in phage-infected, *Escherichia coli* cells, *Genetics* **149**(4) (1998) 1633–1648.
[5] D. C. Bennett, Differentiation in mouse melanoma cells: Initial reversibility and an on–off stochastic model, *Cell* **34**(2) (1983) 445–453.
[6] M. Daviditch and S. Bornholdt, The transition from differential equations to Boolean networks: A case study in simplifying a regulatory network model, *J. Theor. Biol.* **255**(3) (2008) 269–277.
[7] H. De Jong, Modeling and simulation of genetic regulatory systems: A literature review, *J. Comput. Biol.* **9**(1) (2002) 67–103.
[8] M. A. Dingemanse, P. A. Boer, A. F. Moorman, R. Charles and W. H. Lamers, The expression of liver-specific genes within rat embryonic hepatocytes is a discontinuous process, *Differentiation* **56**(3) (1994) 153–162.
[9] M. B. Elowitz, A. J. Levine, E. D. Siggia and P. S. Swain, Stochastic gene expression in a single cell, *Science* **297**(5584) (2002) 1183–1186.

[10] E. Gehrmann and B. Drossel, Boolean versus continuous dynamics on simple two-gene modules, *Phys. Rev. E* **82**(4) (2010) 046120.
[11] J. Gouzé, Positive and negative circuits in dynamical systems, *J. Biol. Sys.* **6** (1998) 11–15.
[12] S. Kauffman, Homeostasis and differentiation in random genetic control networks, *Nature* **224** (1969) 177–178.
[13] M. S. Ko, H. Nakauchi and N. Takahashi, The dose dependence of glucocorticoid-inducible gene expression results from changes in the number of transcriptionally active templates, *EMBO J.* **9**(9) (1990) 2835.
[14] K. Lewis, Programmed death in bacteria, *Microbiol. Molec. Biol. Rev.* **64** (2000) 503–514.
[15] F. Li, T. Long, Y. Lu, Q. Ouyang and C. Tang, The yeast cell-cycle network is robustly designed, in *Proc. Natl. Acad. Sci. USA* **101**(14) (2004) 4781–4786.
[16] D. Murrugarra, A. Veliz-Cuba, B. Aguilar, S. Arat and R. C. Laubenbacher, Modeling stochasticity and variability in gene regulatory networks, *EURASIP J. Bioinform. Syst. Biol.* **2012**(1) (2012) 5.
[17] E. Remy, P. Ruet and D. Thieffry, Graphic requirements for multistability and attractive cycles in a Boolean dynamical framework, *Adv. Appl. Math.* **41**(3) (2008) 335–350.
[18] T. Schlitt and A. Brazma, Current approaches to gene regulatory network modeling, *BMC Bioinform.* **8**(Suppl. 6) (2007) 1–22.
[19] I. Shmulevich, E. R. Dougherty and W. Zhang, From Boolean to probabilistic Boolean networks as models of genetic regulatory networks, *Proc. IEEE* **90**(11) (2002) 1778–1792.
[20] H. Siebert, Deriving behavior of Boolean bioregulatory networks from subnetwork dynamics, *Math. Comput. Sci.* **2**(3) (2009) 421–442.
[21] H. Siebert, Analysis of discrete bioregulatory networks using symbolic steady states, *Bull. Math. Biol.* **73** (2010) 873–898.
[22] D. Thieffry and R. Thomas, Qualitative analysis of gene networks, *Pac. Symp. Biocomput.* (1998), p. 77.
[23] R. Thomas, D. Theiffry and M. Kaufman, Dynamical behavior of biological regulatory networks. I. Biological role of feedback loops and practical use of the concept of the loop-characteristic state, *Bull. Math. Biol.* **57** (1995) 247–276.
[24] M. C. Walters, S. Fiering, J. Eidemiller, W. Magis, M. Groudine and D. I. Martin, Enhancers increase the probability but not the level of gene expression, in *Proc. Natl. Acad. Sci.* **92**(15) (1995) 7125–7129.
[25] G. W. Wright and R. M. Simon, A random variance model for detection of differential gene expression in small microarray experiments, *Bioinformatics* **19**(18) (2003) 2448–2455.