



## A Bayesian connectivity-based approach to constructing probabilistic gene regulatory networks

Xiaobo Zhou<sup>1</sup>, Xiaodong Wang<sup>2</sup>, Ranadip Pal<sup>1</sup>, Ivan Ivanov<sup>1</sup>, Michael Bittner<sup>3</sup> and Edward R. Dougherty<sup>1,4,\*</sup>

<sup>1</sup>Department of Electrical Engineering, Texas A&M University, College Station, TX 77843, USA, <sup>2</sup>Department of Electrical Engineering, Columbia University, New York, NY 10027, USA, <sup>3</sup>Translational Genomic Research Institute, Phoenix, AZ 85004, USA and <sup>4</sup>Department of Pathology, University of Texas M.D. Anderson Cancer Center, Houston, TX 77030, USA

Received on July 16, 2003; revised on April 7, 2004; accepted on May 3, 2004  
Advance Access publication May 14, 2004

### ABSTRACT

**Motivation:** We have hypothesized that the construction of transcriptional regulatory networks using a method that optimizes connectivity would lead to regulation consistent with biological expectations. A key expectation is that the hypothetical networks should produce a few, very strong attractors, highly similar to the original observations, mimicking biological state stability and determinism. Another central expectation is that, since it is expected that the biological control is distributed and mutually reinforcing, interpretation of the observations should lead to a very small number of connection schemes.

**Results:** We propose a fully Bayesian approach to constructing probabilistic gene regulatory networks (PGRNs) that emphasizes network topology. The method computes the possible parent sets of each gene, the corresponding predictors and the associated probabilities based on a nonlinear perceptron model, using a reversible jump Markov chain Monte Carlo (MCMC) technique, and an MCMC method is employed to search the network configurations to find those with the highest Bayesian scores to construct the PGRN. The Bayesian method has been used to construct a PGRN based on the observed behavior of a set of genes whose expression patterns vary across a set of melanoma samples exhibiting two very different phenotypes with respect to cell motility and invasiveness. Key biological features have been faithfully reflected in the model. Its steady-state distribution contains attractors that are either identical or very similar to the states observed in the data, and many of the attractors are singletons, which mimics the biological propensity to stably occupy a given state. Most interestingly, the connectivity rules for the most optimal generated networks constituting the PGRN are remarkably similar, as would be expected

for a network operating on a distributed basis, with strong interactions between the components.

**Availability:** The appendix is available at <http://gspsnap.tamu.edu/gspweb/pgrn/bayes.html>. username: gspweb password: gsplab.

**Contact:** [edward@ee.tamu.edu](mailto:edward@ee.tamu.edu)

**Supplementary Information:** <http://gspsnap.tamu.edu/gspweb/pgrn/bayes.html>

### 1 INTRODUCTION

In considering connectivity in transcriptional networks, we are frequently drawing inferences about the way that a particular class of molecules, the transcription factors, are operating. The transcription factors are protein molecules that directly interact with genomic DNA to mediate transcription of RNAs. Historically, it has been determined that a relatively small set of these molecules, which themselves must be transcribed to be effective, can provide a very rich menu of control operations by virtue of their ability to have their control behavior modified by direct chemical modification; combinatorial association with other transcription factors to form singular molecular complexes with unique or overlapping functional specificities; direct interactions with other types of factors that modulate gene transcription; and by directing the production of the ensemble of transcription factors present in any given cell. Control by this class of agents is therefore not via a centralized hierarchic control acting to produce a large set of specific control agents, but by a very decentralized set of interactions among the entire set of all classes of control agents.

The more general attributes of biological systems place a variety of constraints on how this system must operate, and therefore, expectations about what kinds of general patterns one expects to find in the patterns of transcriptional profiles. Although specific quantitative studies on this general

\*To whom correspondence should be addressed.

behavior have not been carried out, transcription profiling to date clearly shows trends that are in agreement with biological expectations. At the most basic level, the need for cells of all types to carry out basic metabolic functions and to produce common, basic architectural elements would argue for an underlying pattern of similar transcription patterns for genes that participate in common functions within all the cells of a multicellular organism. This is readily seen in profiles as a large cohort of genes whose transcription products are present at fairly comparable levels in a wide variety of normal and even cancerous cells. Layered above this level of generic infrastructure are the requirements for producing the gene products associated with the various specific functions of different cell types. Since differentiated cells must maintain their particular phenotypic characteristics over long periods of time as they perform their duties within particular tissues, one would expect considerable similarity across individuals in the transcription patterns exhibited by cells of a particular tissue. This has been found to be true, and has been studied most aggressively in the area of oncology, where transcription patterns specific to the tissue of origin and the cancer phenotype can be readily demonstrated (Bittner *et al.*, 2000). These biological behaviors strongly suggest that transcriptional regulation operates as a very highly determined system, capable of maintaining itself within tight operational boundaries. To achieve this goal using decentralized control, one would expect that the elements of the system must exhibit a high degree of mutual reinforcement, since in a decentralized system of control, both the measurement of congruence to the goal state and the ability to move the component parts of the system toward the goal state must be decentralized.

This study constructs hypothetical regulatory networks based on the observed behavior of a set of genes whose expression patterns vary across a set of melanoma samples exhibiting two very different phenotypes with respect to cell motility and invasiveness. The method of construction optimizes the regulatory connectivity in a Bayesian framework so that operation of the hypothetical regulatory network produces state vectors similar to the original observations. The connection scheme within a network describes the predictor–target relationships among the constituent genes. The central question being asked is whether network construction based on optimizing connectivity will lead to regulation consistent with biological expectations. A key expectation is that the hypothetical networks should produce a few, very strong attractors, highly similar to the original observations, mimicking biological state stability and determinism. Under the assumption that we are sampling from the steady state, this biological state stability means that most of the steady-state probability mass is concentrated in the attractors and that real-world attractors are most likely to be singleton attractors consisting of one state, rather than cyclic attractors consisting of several states. Under the assumption that we are sampling from the steady state, a key criterion for checking the validity

of a designed network is that much of its steady-state mass lies in the states observed in the sample data because it is expected that the data states consist mostly of attractor states (Kim *et al.*, 2002). Another central point is that, since it is expected that the biological control is distributed and mutually reinforcing, interpretation of the observations should lead to a very small number of connection schemes compatible with the observations. In essence, the particular interactions arising between the various genes necessary for their actions to reinforce movement towards a stable attractor should result in the data only being consistent with a tremendously constrained set of gene-to-gene connections.

It is important to recognize at the outset that there are inherent limitations to the design of dynamical systems from steady-state data. Steady-state behavior constrains the dynamical behavior, but does not determine it. In particular, it does not determine the basin structure. Thus, while we might obtain good inference regarding the attractors, we may obtain poor inference relative to the steady-state distribution. This is because if the basin of an attractor is small, it is easy for the network to leave the basin when a gene is perturbed; if the basin is large, then it is difficult to leave the basin. In the other direction, a small basin will be less likely to ‘catch’ a network perturbation than a large basin. The consequence of these considerations is that the steady-state probabilities of attractor states depend on the basin structure. Building a dynamical model from steady-state data is a kind of overfitting. It is for this reason that we view a designed network as providing a regulatory structure that is consistent with the observed steady-state behavior. Given our main interest is in steady-state behavior, this is reasonable in that we are trying to understand dynamical regulation corresponding to steady-state behavior.

Heretofore, there have been a number of approaches to modeling gene regulatory networks, in particular, Bayesian networks (Friedman *et al.*, 2000), Boolean networks (Kauffman, 1993) and probabilistic Boolean networks (PBNs) (Shmulevich *et al.*, 2002a,b), the latter providing an integrative view of genetic function and regulation. The original design strategy for PBNs in Shmulevich *et al.* (2002a) is based on the coefficient of determination between target and predictor genes. Although the Boolean framework leads to computational simplification and elegant formulation of relations, the model extends directly to genes having more than two states. This view has been considered by Zhou *et al.* (2003), where the key factor for network construction is a non-linear regression method based on reversible-jump Markov chain Monte Carlo (MCMC) annealing. The method is a hybrid technique that combines clustering with Bayesian analysis. A natural alternative would be a fully Bayesian approach that utilizes statistical optimality.

In this paper, we propose a fully Bayesian approach to construct probabilistic gene regulatory networks based essentially on optimizing network topology. Each network configuration

is represented by a directed graph whose vertices correspond to genes and whose directed edges correspond to input–output relationships between predictor and target genes. Given the possible sets of parent genes for all genes, the corresponding predictors are estimated. Then an MCMC method is employed to search for the best set of networks. The probabilities associated with the probabilistic network follow naturally from the Bayesian scores of the chosen network configurations.

All figures and tables can be accessed in the Appendix on the companion website.

## 2 PROBABILISTIC GENE REGULATORY NETWORKS

We define a gene regulatory network (GRN) to consist of a set of  $n$  genes,  $g_1, g_2, \dots, g_n$ , each taking values in a finite set  $V$  (containing  $d$  values), a family of regulatory sets,  $R_1, R_2, \dots, R_n$ , where  $R_k$  contains the genes that determine the value of gene  $g_k$ , and a set of functions,  $f_1, f_2, \dots, f_n$ , governing the state transitions of the genes. The value of gene  $g_k$  at time  $t + 1$  is given by  $g_k(t + 1) = f_k[g_{k1}(t), g_{k2}(t), \dots, g_{kn}(t)]$ , where  $R_k = \{g_{k1}, g_{k2}, \dots, g_{kn}\}$ . For a Boolean network,  $V = \{0, 1\}$ .

A probabilistic gene regulatory network (PGRN) consists of a set of  $n$  genes,  $g_1, g_2, \dots, g_n$ , each taking values in a finite set  $V$  (containing  $d$  values), and a set of vector-valued network functions,  $\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_r$ , governing the state transitions of the genes. Mathematically, there is a set of state vectors  $S = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m\}$ , with  $m = d^n$  and  $\mathbf{x}_k = (x_{k1}, x_{k2}, \dots, x_{kn})$ , where  $x_{ki}$  is the value of gene  $g_i$  in state  $k$ . Each network function  $\mathbf{f}_j$  is composed of  $n$  functions  $\psi_{j1}, \psi_{j2}, \dots, \psi_{jn}$ , and the value of gene  $g_i$  at time  $t + 1$  is given by  $g_i(t + 1) = \psi_{ji}[g_1(t), g_2(t), \dots, g_{i-1}(t), g_{i+1}(t), \dots, g_n(t)]$ . The choice of which network function  $\mathbf{f}_j$  to apply is governed by a selection procedure. Specifically, at each time point a random decision is made as to whether to switch the network function for the next transition, with a probability  $q$  of a switch being a system parameter. If a decision is made to switch the network function, then a new function is chosen from among  $\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_r$ , with the probability of choosing  $\mathbf{f}_j$  being the selection probability  $c_j$ . In other words, each network function  $\mathbf{f}_j$  determines a GRN and the PGRN behaves as a fixed GRN until a random decision (with probability  $q$ ) is made to change the network function according to the probabilities  $c_1, c_2, \dots, c_r$  from among  $\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_r$ . In effect, a PGRN switches between the GRNs defined by the network functions according to the switching probability  $q$ . A final aspect of the system is that at each time point there is a probability  $p$  of any gene changing its value uniformly randomly among the other possible values in  $V$ . Since there are  $n$  genes, the probability of there being a random perturbation at any time point is  $1 - (1 - p)^n$ . The state space  $S$  of the network together with the set of network functions, in conjunction with transitions between the states and network functions, determine a

Markov chain. The random perturbation makes the Markov chain ergodic, meaning that it has the possibility of reaching any state from another state and that it possesses a long-run (steady-state) distribution.

As a GRN evolves in time, it will eventually enter a fixed state, or a set of states, through which it will continue to cycle. In the first case the state is called a singleton or fixed-point attractor, whereas, in the second case it is called a cyclic attractor. The attractors that the network may enter depend on the initial state. All initial states that eventually produce a given attractor constitute the basin of that attractor. The attractors represent the fixed points of the dynamical system that capture its long-term behavior. The number of transitions needed to return to a given state in an attractor is called the cycle length. Attractors may be used to characterize a cell's phenotype (Kauffman, 1993). The attractors of a PGRN are the attractors of its constituent GRNs. However, because a PGRN constitutes an ergodic Markov chain, its steady-state distribution plays a key role. Depending on the structure of a PGRN, its attractors may contain most of the steady-state probability mass.

## 3 BAYESIAN CONSTRUCTION OF PROBABILISTIC REGULATORY NETWORKS

We represent a GRN by a pair  $\mathcal{N} = (G, \Gamma)$ .  $G$  is a directed graph whose vertices correspond to the genes,  $x_1, x_2, \dots, x_n$ .  $\Gamma$  represents the predictors for each gene, i.e.  $\Gamma = \{f^{(1)}, f^{(2)}, \dots, f^{(n)}\}$ . Each predictor  $f^{(i)}$  is in turn parameterized by  $\{J^{(i)}, \Theta_j^{(i)}\}$  as to be discussed shortly. The problem of constructing a PGRN is the following: given a set of gene expression measurements  $\mathcal{O} = \{\underline{x}_1, \dots, \underline{x}_n\}$  where  $\underline{x}_i = [x_{i1}, \dots, x_{iN}]^T$  is the observation vector of node  $x_i$ , find a set of networks  $\{\mathcal{N}_k\}_{k=1}^K$  that best match  $\mathcal{O}$ .

We approach the problem in a Bayesian context relative to network topology by searching for the networks with the highest a posteriori probabilities

$$P(G|\mathcal{O}) \propto P(\mathcal{O}|G)P(G), \quad (1)$$

where  $P(G)$ , the prior probability for the network  $G$ , is assumed to take a uniform distribution on all possible topologies. Note that

$$P(\mathcal{O}|G) = \int p(\mathcal{O}|G, \Gamma)p(\Gamma)d\Gamma. \quad (2)$$

To make the problem tractable, we impose a conditional independence assumption on  $p(\mathcal{O}|G, \Gamma)$ :

$$p(\mathcal{O}|G, \Gamma) \triangleq p(\underline{x}_1, \dots, \underline{x}_n|G, \Gamma) = \prod_{i=1}^n p(\underline{x}_i|U_i, f^{(i)}), \quad (3)$$

$$p(\Gamma) \triangleq p(f^{(1)}, \dots, f^{(n)}) = \prod_{i=1}^n p(f^{(i)}), \quad (4)$$

where  $p(f^{(i)})$  is a prior density for the predictor  $f^{(i)}$  and  $U_i$  denotes the observations corresponding to the parent nodes of  $x_i$  in  $G$ . Assuming node  $i$  has  $v(i)$  parent nodes,  $U_i = [x_{i1}, x_{i2}, \dots, x_{iv(i)}]$ . Since the computation of Equation (2) is prohibitive, we choose to plug in the estimated parameters  $\Gamma$  instead of integrating them out, that is, we approximate  $P(\mathcal{O}|G)$  by

$$P(\mathcal{O}|G) \approx p(\mathcal{O}|G, \hat{\Gamma})p(\hat{\Gamma}) = \prod_{i=1}^n p(x_i|U_i, \hat{f}^{(i)})p(\hat{f}^{(i)}), \quad (5)$$

where  $\hat{f}^{(i)}$  is the designed optimal predictor.

Within this framework, we propose a fully Bayesian approach to construct PGRNs by searching over the space of all possible network topologies and picking those with the highest Bayesian scores. Given a network configuration  $G$ , we calculate the predictors  $\Gamma$  associated with it as well as the corresponding Bayesian score  $P(\mathcal{O}|G)$  in Equation (5). The Bayesian approach controls the complexity of  $G$  because  $P(\mathcal{O}|G)$  is approximated by the product of Equation (5) and the complexity of  $p(x_i|U_i, \hat{f}^{(i)})$  is controlled by the minimum description length (MDL)-based reversible jump MCMC algorithm (see Zhou *et al.*, 2003). We then move to the next network configuration by implementing a reversible jump MCMC step. The process repeats a sufficient number of iterations. Finally, the PGRN is formed by choosing those networks with the highest Bayesian scores. Summarizing the algorithm, we generate an initial directed graph  $G^{(0)}$ , say, by clustering, and compute  $P(\mathcal{O}|G)$  according to Equation (5); and then, for  $j = 1, 2, \dots$ , we calculate the predictors  $\Gamma^{(j)}$  corresponding to  $G^{(j)}$ , compute the Bayesian score  $P(\mathcal{O}|G)$  according to Equation (5), and pick  $G^{(j+1)}$  via an MCMC step. We next elaborate on each of the above steps.

### 3.1 Calculating predictors

Based on the parent set  $X_G^{(i)}$  of node  $i$  in the directed graph  $G$ , we want to estimate the predictors  $f^{(i)} : X_G^{(i)} \rightarrow y_i$  from the observations (in fact,  $y_i$  should be  $x_i$ , but to describe the functions clearly we let  $y_i$  stand for  $x_i$ ). For notational convenience, we consider a general formulation,  $f : X \rightarrow y$ . We postulate a multivariate-input, single-output mapping

$$y_t = f(\mathbf{x}_t) + n_t, \quad (6)$$

where  $\mathbf{x}_t \in \mathbb{R}^d$  is a set of input variables,  $y_t \in \mathbb{R}$  is a target variable,  $n_t$  is a noise term and  $t \in \{1, 2, \dots\}$  is an index variable over the data. The learning problem involves computing an approximation to the function  $f$  and estimating the characteristics of the noise process given a set of  $N$  input-output observations,  $\tilde{\mathcal{O}} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N; y_1, y_2, \dots, y_N\}$ . Typical examples include regression, prediction and classification.

We consider the following family of predictors:

$$\mathcal{M}_0 : y_t = b + \beta^T \mathbf{x}_t + n_t, \quad (7)$$

$$\mathcal{M}_J : y_t = \sum_{j=1}^J a_j \phi(\|\mathbf{x}_t - \boldsymbol{\mu}_j\|) + b + \beta^T \mathbf{x}_t + n_t, \quad (8)$$

$$1 \leq J \leq J_{\max},$$

where  $J_{\max} \triangleq \lfloor N/10 \rfloor$  is an upper-bound on the number of nonlinear terms in Equation (8);  $\phi$  is a radial basis function (RBF);  $\|\cdot\|$  denotes a distance metric (usually Euclidean or Mahalanobis);  $\boldsymbol{\mu}_j \in \mathbb{R}^d$  denotes the  $j$ -th RBF center;  $a_j \in \mathbb{R}$  is the  $j$ -th RBF coefficient;  $b \in \mathbb{R}; \beta \in \mathbb{R}^d$  are the linear regression parameters; and  $n_t \in \mathbb{R}$  is assumed to be i.i.d. noise. Depending on our a priori knowledge about the smoothness of the mapping, we can choose different types of basis functions. Here, we use the Gaussian basis function,  $\phi(\varrho) = \exp(-\varrho^2)$ . We express the model (8) in vector-matrix form as

$$\underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}}_{\triangleq \mathbf{y}} = \underbrace{\begin{bmatrix} 1 & x_{1,1} \cdots x_{1,d} & \phi(\|\mathbf{x}_1 - \boldsymbol{\mu}_1\|) \cdots \phi(\|\mathbf{x}_1 - \boldsymbol{\mu}_J\|) \\ 1 & x_{2,1} \cdots x_{2,d} & \phi(\|\mathbf{x}_2 - \boldsymbol{\mu}_1\|) \cdots \phi(\|\mathbf{x}_2 - \boldsymbol{\mu}_J\|) \\ \vdots & \vdots & \vdots \\ 1 & x_{N,1} \cdots x_{N,d} & \phi(\|\mathbf{x}_N - \boldsymbol{\mu}_1\|) \cdots \phi(\|\mathbf{x}_N - \boldsymbol{\mu}_J\|) \end{bmatrix}}_{\triangleq \mathbf{D}} \times \underbrace{\begin{bmatrix} b \\ \beta_1 \\ \vdots \\ \beta_d \\ a_1 \\ \vdots \\ a_J \end{bmatrix}}_{\triangleq \boldsymbol{\alpha}} + \underbrace{\begin{bmatrix} n_1 \\ n_2 \\ \vdots \\ n_N \end{bmatrix}}_{\triangleq \mathbf{n}}, \quad (9)$$

that is

$$\mathbf{y} = \mathbf{D}\boldsymbol{\alpha} + \mathbf{n}. \quad (10)$$

The noise variance is assumed to be  $\eta$ . We assume here that the number  $J$  of RBFs and the parameters  $\boldsymbol{\Theta}_J \triangleq \{\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_J, \eta, \boldsymbol{\alpha}\}$  are unknown. Given the dataset  $\tilde{\mathcal{O}}$ , our objective is to estimate  $J$  and  $\boldsymbol{\Theta}_J$ , where  $\boldsymbol{\Theta}_J \in \mathbb{R}^m \times \mathbb{R}^+ \times \Omega_J$  with  $m = 1 + d + J$ ; that is  $\boldsymbol{\alpha} \in \mathbb{R}^m, \eta \in \mathbb{R}^+$  and  $\boldsymbol{\mu} \in \Omega_J$ , where

$$\Omega_J = \left\{ \boldsymbol{\mu} = (\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_J); \mu_{j,i} \in \left[ \min_{1 \leq l \leq N} x_{l,i} - \varphi, \max_{1 \leq l \leq N} x_{l,i} + \varphi \right], \right. \\ \left. j = 1, \dots, J; i = 1, \dots, d \right\}. \quad (11)$$

Here we set  $\varphi = 0.5$  in our experiments. We consider the Bayesian inference of  $J$  and  $\boldsymbol{\Theta}_J$  based on the joint

posterior distribution  $p(J, \Theta_J | \tilde{\mathcal{O}})$ . Our aim is to estimate this joint distribution from which, by standard probability marginalization and transformation techniques, one theoretically expresses all posterior features of interest. It is not possible to obtain these quantities analytically, since this would require the evaluation of very high dimensional integrals of nonlinear functions in the parameters. An MCMC technique is used to perform the above Bayesian computation. From model (10), given  $J, \mu_1, \dots, \mu_J$ , the least-squares estimate of  $\alpha$  is given by

$$\alpha = (\mathbf{D}^T \mathbf{D})^{-1} \mathbf{D}^T \mathbf{y}. \quad (12)$$

An estimate of  $\eta$  is then given by

$$\eta = \frac{1}{N} (\mathbf{y} - \mathbf{D}\alpha)^T (\mathbf{y} - \mathbf{D}\alpha) = \frac{1}{N} \mathbf{y}^T \mathbf{P}_J^* \mathbf{y}, \quad (13)$$

$$\text{where } \mathbf{P}_J^* \triangleq \mathbf{I}_N - \mathbf{D}(\mathbf{D}^T \mathbf{D})^{-1} \mathbf{D}^T. \quad (14)$$

Based on the MDL criterion, we can impose the following a priori distribution on  $J$  and  $d$  (Andrieu et al., 2001),

$$P(J, d) \propto \exp \left[ - \left( J + \frac{d+1}{2} \right) \log N \right]. \quad (15)$$

Assuming the noise samples are i.i.d. Gaussian, it can then be shown that the joint posterior distribution of  $(J, \mu_1, \dots, \mu_J)$  is given by (Andrieu et al., 2001),

$$p(J, \mu_1, \dots, \mu_J | \tilde{\mathcal{O}}) \propto [(\mathbf{y}^T \mathbf{P}_J^* \mathbf{y})^{-N/2}] P(J, d). \quad (16)$$

Hence the maximum a posteriori (MAP) estimate of these parameters is obtained by maximizing the right-hand side of the Equation (16). This can be done by using the reversible jump MCMC algorithm (Andrieu et al., 2001).

### 3.2 Calculating Bayesian scores

Consider node  $x_i$  with observation  $\mathcal{O}_1 = \{x_i, U_i\}$ . From the above discussion on the form of  $f^{(i)}$ , it follows that  $f^{(i)}$  is parameterized by  $J_i, \mu_{i,1}, \dots, \mu_{i,J_i}(\mathbf{D}_i), \alpha_i, \eta_i \mu d_i$  and

$$p(x_i | U_i, f^{(i)}) = (2\pi \eta_i)^{-N/2} \exp \left( -\frac{1}{2\eta_i} \|x_i - f^{(i)}(U_i)\|^2 \right). \quad (17)$$

Here we assume the prior of  $f^{(i)}$  to be  $P(J_i, d_i)$ . According to Equation (5) and following the derivation in (Andrieu et al., 2001), after straightforward computation, the  $P(\mathcal{O}|G)$

is approximated by

$$\begin{aligned} P(\mathcal{O}|G) &\approx \prod_{i=1}^n p(x_i | U_i, f^{(i)}) p(f^{(i)}) \\ &= \prod_{i=1}^n p(x_i | U_i, f^{(i)}) P(J_i, d_i) \\ &= \prod_{i=1}^n (2\pi \eta_i)^{-N/2} \exp \left( -\frac{1}{2\eta_i} \|x_i - \mathbf{D}_i \alpha_i\|^2 \right) P(J_i, d_i) \\ &\propto \prod_{i=1}^n (x_i^T \mathbf{P}_{J_i}^* x_i)^{-N/2} \exp \left[ - \left( J_i + \frac{d_i + 1}{2} \right) \right], \end{aligned} \quad (18)$$

where Equation (18) follows from Equation (16) and the fact that the predictor  $f^{(i)}$  obtained by the MCMC procedure satisfies Equations (12) and (13), and  $\mathbf{P}_{J_i}^* \triangleq \mathbf{I}_N - \mathbf{D}_i [\mathbf{D}_i^T \mathbf{D}_i]^{-1} \mathbf{D}_i^T$ . Here  $\mathbf{D}_i$  is similar to the  $\mathbf{D}$  in Equation (10).

### 3.3 Searching networks via MCMC

The space of all possible network topologies is huge. To find the networks with highest scores, we again resort to the MCMC strategy. Given the current network topology  $G$ , define its neighborhood,  $\aleph(G)$ , to be the set of graphs which differ by one edge from  $G$ , i.e. we can generate  $\aleph(G)$  by considering all single-edge additions, deletions and reversals (Giudici and Castelo, 2003). Let  $q(G'|G) = 1/|\aleph(G)|$ , for  $G' \in \aleph(G)$ , and  $q(G'|G) = 0$  for  $G' \notin \aleph(G)$ . Sample  $G'$  from  $G$  by a random single edge addition, deletion or reversal in  $G$ . Then the accept ratio is given by

$$R = \frac{q(G|G')P(G'|G)}{q(G'|G)P(G|G)} = \frac{P(\mathcal{O}|G')}{P(\mathcal{O}|G)},$$

where  $P(\mathcal{O}|G)$  and  $P(\mathcal{O}|G')$  can be obtained from Equation (5). In this study, 20 000 MCMC iterations are implemented with the first 10 000 being the burn-in period. Another 10 000 networks with configurations are kept to construct the PGRN.

### 3.4 Constructing probabilistic gene regulatory networks

We pick the networks with the highest Bayesian scores (although this could be modified by prior knowledge). After selecting the  $K$  graphs  $\{\mathcal{N}_k\}_{k=1}^K$  with the highest scores out of the 10 000 generated by the MCMC technique, we then construct PGRNs. For each network  $\mathcal{N}_k$ , the predictors and the corresponding parent sets for each gene are given according to the previous discussion. The probabilities to select networks

are defined by their Bayesian scores,

$$\bar{q}_k = \prod_{i=1}^n p(f_k^{(i)} | \tilde{\mathcal{O}}) \propto \prod_{i=1}^n (x_i^T \mathbf{P}_{J_i^k}^* x_i)^{-m/2} \\ \times \exp \left[ - \left( J_i^k + \frac{d_i^k + 1}{2} \right) \right], \quad k = 1, \dots, K,$$

where the corresponding predictors  $f_k^{(i)} : X_k^{(i)} \rightarrow x_i$  can be parameterized by  $J_i^k, \boldsymbol{\mu}_{i,1}^k, \dots, \boldsymbol{\mu}_{i,J_i^k}^k(\mathbf{D}_i^k), \boldsymbol{\alpha}_i^k, \eta_i^k, d_i^k, k = 1, \dots, K$ , as described in the above discussion. Finally, the probability  $q_k$  is defined as its normalized Bayesian score value, i.e.

$$q_k \triangleq \frac{\bar{q}_k}{\sum_{j=1}^K \bar{q}_j}, \quad k = 1, \dots, K.$$

#### 4 STEADY-STATE ANALYSIS OF PGRNS

We are keenly interested in the long-run behavior of a PGRN. If we fix a network, so that we have a gene regulatory network but not a PGRN, its dynamic behavior results from the expression state of each gene at step  $t + 1$  being predicted by the expression levels of its predictor genes at step  $t$  in the same network. The flow of the resulting finite-state Markov chain is depicted by

$$S^{(t)} = [x_1^{(t)}, x_2^{(t)}, \dots, x_n^{(t)}] \\ \rightarrow S^{(t+1)} = [x_1^{(t+1)}, x_2^{(t+1)}, \dots, x_n^{(t+1)}].$$

If the networks runs sufficiently long, then it will settle into one of its attractor cycles. As discussed by Shmulevich *et al.* (2002b) and applied by Kim *et al.* (2002), if the network involves random perturbations, meaning that at each transition there is a small probability  $p$  that a gene will switch its value randomly rather than being controlled by the rule  $\binom{i}{k}$ , then the Markov chain is ergodic and therefore possesses a steady-state distribution. This means there exists a probability distribution  $\boldsymbol{\pi} = (\pi_1, \pi_2, \dots, \pi_M)$  such that for all states  $i, j \in \{1, 2, \dots, M\}$ ,  $\lim_{r \rightarrow \infty} P_{ij}^r = \pi_j$ , where  $P_{ij}^r$  is the  $r$ -step transition probability from initial state  $i$  to target state  $j$ . Perturbations allow the chain to jump out an attractor cycle.

The steady-state issue was studied in the context of PBNs (Shmulevich *et al.*, 2002a), in which case a different network transition model was selected at each time point. As noted by Shmulevich *et al.* (2002c) and studied extensively by M. Brun, E. R. Dougherty and I. Shmulevich (submitted for publication), an approach consistent with the stability of biological systems is to assume a small probability  $q$  for a network change at any time point. If a change is made, then the current state of the Markov chain is treated as an initial state and the Markov chain moves forward in this new selected network. Here we assume both perturbation and probabilistic network switching in the context of PGRNs, with  $p = q = 0.001$ .

Finally, following Kim *et al.* (2002), we use the Kolmogorov–Smirnov statistic to diagnose whether a chain converges after a burn-in of  $T = 1\,000\,000$  iterations. Given Markov chain samples  $S^{(T+1)}, S^{(T+2)}, \dots, S^{(T+N)}$ , we want to compare the distributions of the two halves of these samples,  $S^{(T+1)}, S^{(T+2)}, \dots, S^{(T+N/2)}$  and  $S^{(T+1)}, S^{(T+N/2+1)}, \dots, S^{(T+N)}$ . Here we set  $N = 10\,000\,000$  and the Kolmogorov–Smirnov statistic (Robert and Casella, 1999) is defined as the maximum value of the absolute difference between two cumulative distributions. After the Markov chain converges, we continue  $100\,000\,000$  iterations to study the steady-state behavior.

#### 5 TESTING THE ALGORITHM ON SYNTHETIC NETWORKS

Before applying the design algorithm to a melanoma dataset, we will examine its steady-state properties on two synthetic PBNs for which we know the attractor structure. As explained in the Introduction section, we are particularly interested in designing PBNs that capture singleton attractors. Therefore, we do not want to apply the algorithm to arbitrarily random synthetic PBNs because these can possess a wide variety of attractor structures (Kauffman, 1993). Thus, we are presented with the following search problem: given a set of states earmarked to be singleton attractors and an upper bound on the number of predictors for each gene, find a Boolean network satisfying these conditions that contains no other singleton attractors and no cyclic attractors. There is an implicit consistency requirement for this search: if  $f$  predicts the value of gene  $g$  and has essential variables  $x_1, x_2, \dots, x_m$  (which without loss of generality we consider it to be first  $m$  variables) out of the full cohort  $x_1, x_2, \dots, x_n$ ,  $m < n$ , and if  $f(a_1, a_2, \dots, a_n) = 1$  for some particular set of values of  $x_1, x_2, \dots, x_n$  then  $f(a_1, a_2, \dots, a_m, x_{m+1}, \dots, x_n) = 1$  for any values of  $x_{m+1}, \dots, x_n$ . An analogous statement holds for  $f(a_1, a_2, \dots, a_n) = 0$ . This is a computationally intensive search. An algorithm has been developed to accomplish it; nonetheless, it remains burdensome and therefore we limit our synthetic examples to  $n - 6$  genes.

To test the algorithm we consider two synthetic PBNs consisting of two Boolean networks each. We label the PBNs as  $N_1$ , with constituent Boolean networks  $B_{11}$  and  $B_{12}$ , and  $N_2$ , with constituent Boolean networks  $B_{21}$  and  $B_{22}$ . The constituent networks possess equal probabilities, and the switching and perturbation probabilities are given by  $q = p = 0.001$ . The design algorithm has been applied in the following manner to both  $N_1$  and  $N_2$ : take a sample of size 60 from the steady-state distribution, randomly seed the design algorithm 10 times and in each case take the highest scoring network, and form the designed PBN by using the 10 resulting networks as its constituent networks.

For PBN  $N_1$ , the attractor-basin structures for  $B_{11}$  and  $B_{12}$  are shown in Figures 1 and 2, respectively.  $B_{11}$  has

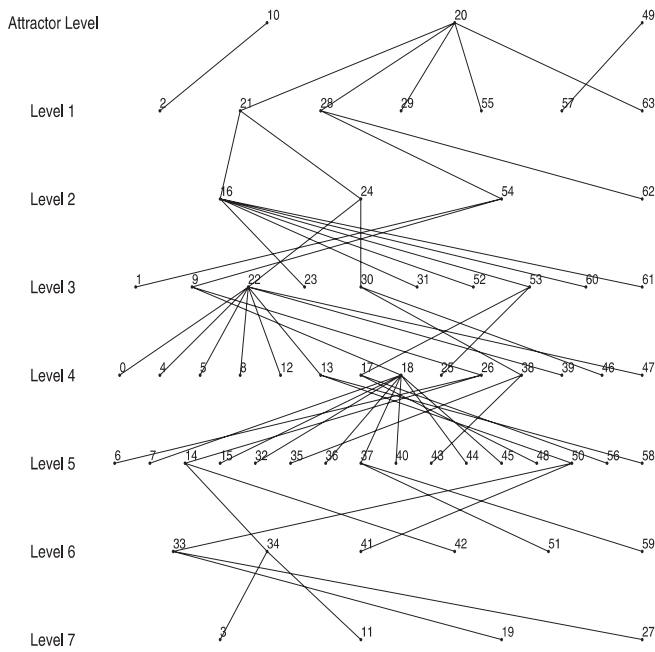


Fig. 1. Network  $B_{11}$

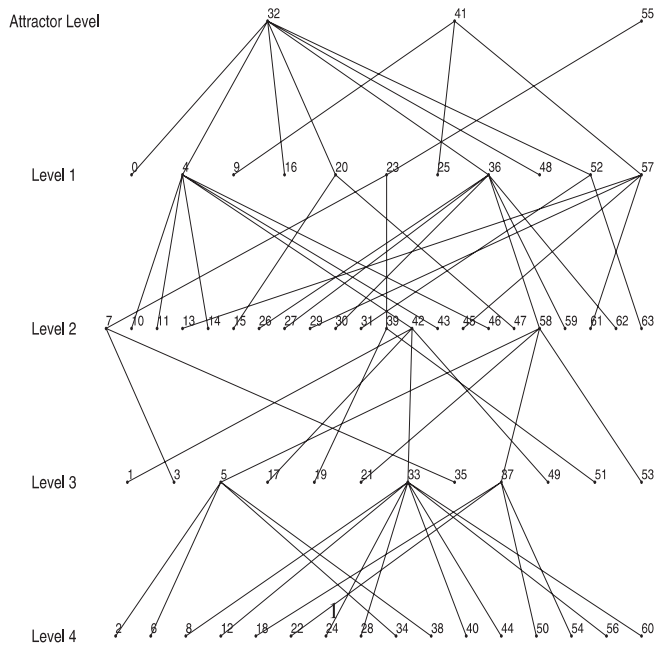


Fig. 2. Network  $B_{12}$

three singleton-attractor states: 10, 20 and 49.  $B_{12}$  also has three singleton-attractor states: 32, 41 and 55. In the simulated steady-state distribution for  $N_1$ , attractors 20 and 32 carry most of the steady-state mass, with  $P(20) = 0.49$  and  $P(32) = 0.50$ . The other attractors carry negligible mass. Note the extremely small basins of attractors 10, 49 and 55. Supplementary Table 1 shows the attractors, both singleton

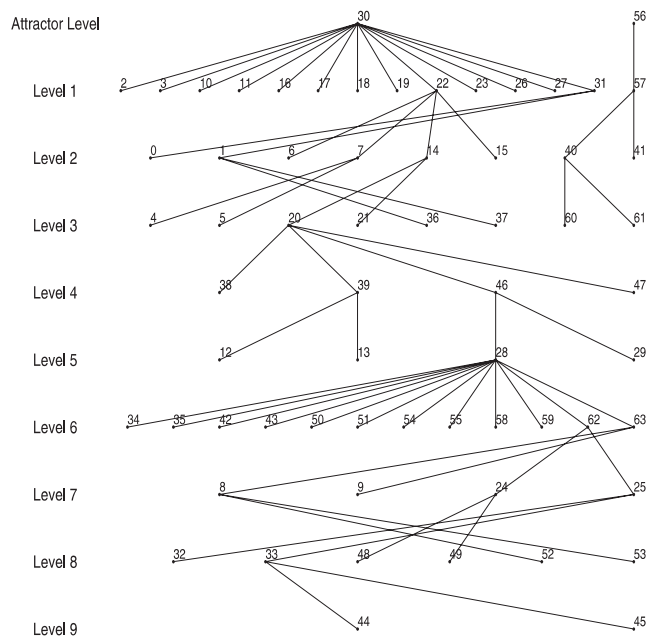
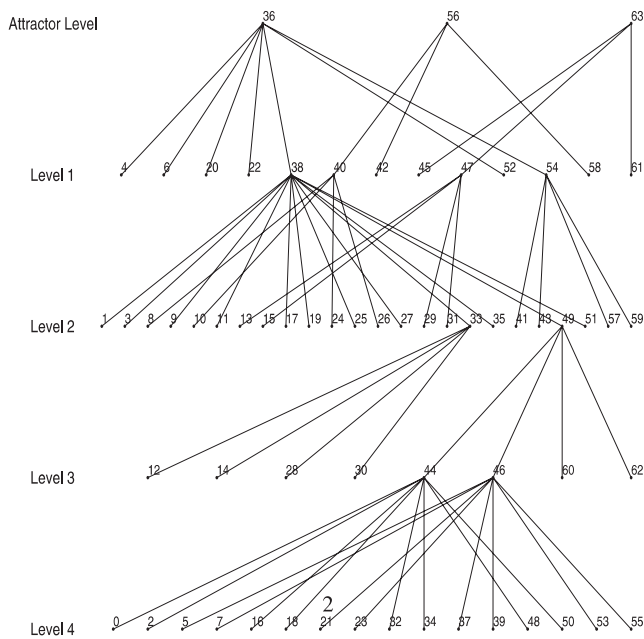


Fig. 3. Network  $B_{21}$

and cyclic, in the 10 networks taken to form the designed PBN. The heavy attractor states 20 and 32 appear as singleton attractors in all 10 Boolean networks. Those with negligible mass do not appear, nor did they appear in the sample data. There are no spurious singleton attractors and very few spurious cyclic attractors in the designed PBN. Upon running the PBN with the 10 constituent networks having equal probability to estimate its steady-state distribution, the attractors 20 and 32 contain the majority of the steady-state mass, with  $P(20) = 0.59$  and  $P(32) = 0.19$ . The four states composing the spurious cyclic attractors possess a total steady-state mass of 0.15. This concentration of steady-state mass within the sample-data states compares very favorably with a strictly conditional-probability-based design (Kim *et al.*, 2002).

For PBN  $N_2$ , the attractor-basin structures for  $B_{21}$  and  $B_{22}$  are shown in Figures 3 and 4, respectively.  $B_{21}$  has two singleton attractors, states 30 and 56, whereas  $B_{22}$  has three singleton attractors, states 36, 56 and 63. The steady-state distribution for  $N_2$  has been obtained via simulation. Attractors 30 and 36 carry most of the steady-state mass, with  $P(30) = 0.51$  and  $P(36) = 0.48$ . The other attractors, 56 and 63, carry negligible mass. A key point is the small basins of attractors 56 and 63. Supplementary Table 2 shows the attractors, both singleton and cyclic, in the 10 networks taken to form the designed PBN. The most important point is that states 30 and 36 are singleton attractors in all 10 Boolean networks. State 63, which occurred once in the sample taken from the steady-state distribution, appears in 6 out of the 10, and state 56, which did not occur in the sample, does not appear. The anomalous appearance of state 49 as a singleton attractor



**Fig. 4.** Network  $B_{22}$

6 times can perhaps be explained by the fact that half of the transients in basin level 3 and all of the transients in basin level 4 for the constituent network  $B_{22}$  pass through state 49 to reach attractor 36. There are more spurious cycles in the case of  $N_2$  than in  $N_1$ . This is reflected in the fact that the total mass of the original attractors is reduced in the steady-state distribution of the designed network:  $P(30) = 0.23$ ,  $P(36) = 0.17$ ,  $P(63) = 0.06$  and  $P(56) = 0.02$ . Even with this diminution, the strong singleton attractors far outweigh any other states, and the results continue to compare favorably to strictly conditional-probability-based design.

Before proceeding, we point out several design limitations. First, there is the issue of designing from steady-state data that has been discussed in the Introduction section. Second, we have very small sample sizes relative to the complexity of the inference when using microarray data. Part of the motivation for a Bayesian connectivity approach is to better fit the connectivity to the data given the paucity of data. Finally, the assumption that the biological system being modeled does not possess cyclic attractors suggests that we adjust the design procedure so as to reduce their number. Although we are pursuing this possibility, it is hampered by two large impediments.

The obvious impediment is computational. The Bayesian algorithm is already computationally intensive. Directly checking for cycles and readjusting at each step would make the algorithm unfeasible, even more so were we to consider steady-state mass at each step. Another difficulty is the large effects that very small network changes can have on the steady-state distribution, thereby making it tricky to make network alterations to achieve desirable steady-state behavior (Ivanov and Dougherty, 2004).

## 6 MELANOMA APPLICATION

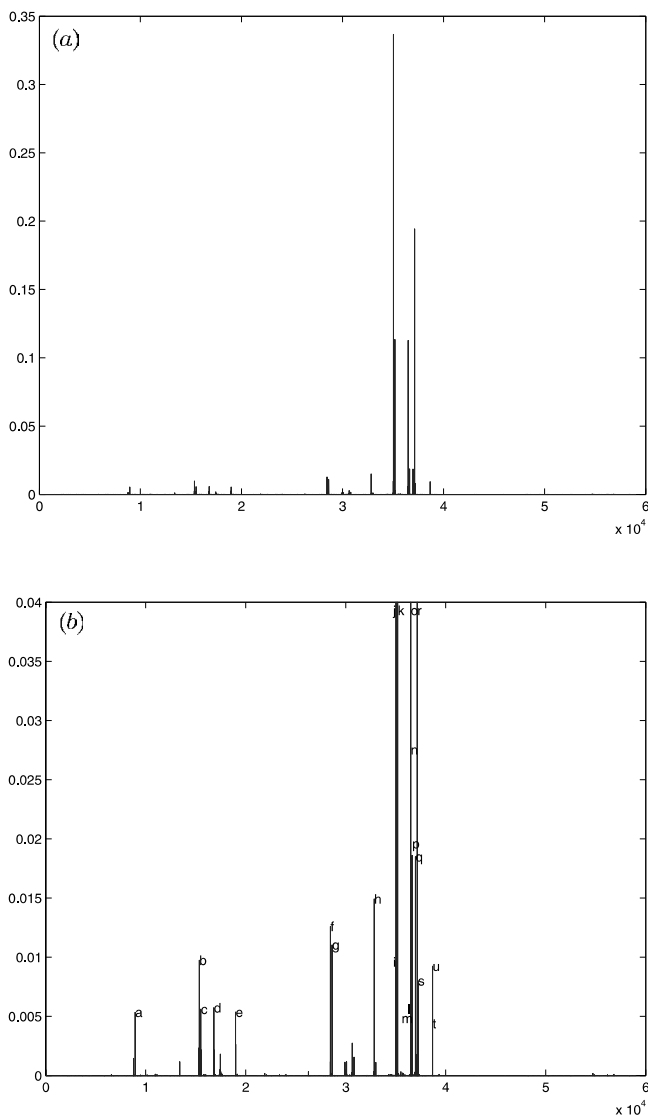
The gene-expression profiles used in this study result from the study of 31 malignant melanoma samples (Bittner *et al.*, 2000). For the study, total messenger RNA was isolated directly from melanoma biopsies, and fluorescent cDNA from the message was prepared and hybridized to a microarray containing probes for 8150 cDNAs (representing 6971 unique genes). The 10 genes used here for the model were chosen from a set of 587 genes from the melanoma dataset that have been subjected to an analysis of their ability to cross predict each other's state in a multivariate setting (Kim *et al.*, 2002): pirin, WNT5A, S100P, RET-1, MMP-3, PHO-C, MART-1, HADHB, synuclein and STC2.

The columns of Supplementary Table 3 refer to data states that appear as singleton attractors for the 25 top-scoring networks, with the last two columns giving the numbers of data-state attractors and the total number of singleton attractors for the network (Supplementary Table 4 for a listing of the data states). Four states are of particular interest: the data states  $A = 35\,031$  and  $B = 37\,139$ , and the non-data states  $C = 35\,193$  and  $D = 36\,491$ . There are five networks for which states 3 and 17 502 are singleton attractors, and these do not appear as singleton attractors for any other networks. Let  $\mathcal{Y}$  denote the class consisting of these five networks. No network in  $\mathcal{Y}$  has state 32 041,  $A$  or  $B$  as a singleton attractor. But these are all singleton attractors for the other 20 networks, except for network 23 on the list, for which  $A$  is not a singleton attractor. Let  $\mathcal{X}$  denote the class consisting of these twenty networks. Note that  $\mathcal{X} \cap \mathcal{Y} = \emptyset$  and  $\mathcal{X} \cup \mathcal{Y} = \mathcal{N}$  (the full PGRN). Finally, let  $\mathcal{Z}$  denote the class of networks having the two-state attractor cycle  $C \leftrightarrow D$ .  $\mathcal{Z}$  is the subset of  $\mathcal{X}$  for which the data state 8428 is a singleton attractor.

### 6.1 Steady-state analysis

We have constructed a ternary PGRN using 10 networks in  $\mathcal{X}$ , keeping the number to 10 to facilitate the steady-state distribution being estimated via simulation using all observed states as initial states. We have found the steady-state distributions for the different initializations to be essentially the same. The steady-state distribution is shown in Figure 5a. Owing to the concentration of mass in four states, the rest of the distribution appears negligible. Hence, in Figure 5b we show the steady-state distribution with the masses of the dominating states truncated to 0.04 to show that there are other states with appreciable mass. The states with highest probabilities are labeled from (a) to (t), referring to states 8940, 15 348, 15 510, 16 806, 18 985, 28 470, 28 632, 32 841, 35 022, 35 031, 35 193, 36 480, 36 482, 36 489, 36 491, 36 651, 36 977, 37 139, 37 218, 38 676, 38 677, respectively.

Supplementary Table 4 lists the 21 states observed in the data, their state numbers as stored in our database, and their ranks according to steady-state mass (with the number in parentheses next to the rank giving the number of times the state is observed in the data, if more than once).



**Fig. 5.** Estimated distribution after long run: (a) steady-state distribution; (b) truncation at 0.04.

Considering the total number,  $3^{10} = 59\,049$ , of states, for the most part the data states are fairly high ranked in the steady-state distribution, which is expected. The situation is more striking if we note that data states *A* and *B* carry over half (54%) of the steady-state mass. Almost another quarter of the steady-state mass is carried by the non-data states *C* and *D*, each carrying 11.3% of the mass. They are very close to states *A* and *B*: state *C* differs from state *A* only in PHO-C, with PHO-C having value 1 in *C* and value 0 in *A*; and state *D* differs from state *B* only in MMP-3 and PHO-C, with MMP-3 and PHO-C having value 0 in *D* and value 1 in state *B*. Altogether, the four states *A*, *B*, *C*, and *D* carry almost 75% of the steady-state mass and they agree for six of the component genes, differing only among the genes RET-1, MMP-3, PHO-C and STC2.

## 6.2 Connectivity analysis

A key interest in constructing PGRNs is to examine the connectivity of the constituent networks, since the entire Bayesian design is predicated on optimizing connectivity. Moreover, since it is the attractors of the constituent networks that comprise the PGRN attractors, we would like to investigate the relationship between connectivity and attractors.

A connectivity matrix is associated with each of the 25 networks (the full set of connectivity matrices being provided on the companion website). In the matrices, rows refer to targets and columns refer to predictors. For individual networks, most of the time targets have three predictors, but not always. Nine matrices (Supplementary Tables 3–11) provide summary views of the connectivity across the 25 constituent networks.

Matrix  $M_1$  gives the overall summary. Note that WNT5A is a predictor for five genes in all 25 networks, and a predictor for a sixth gene in almost all of the networks. Although less so, pirin and S100P are also frequent predictors. Note also that pirin and MMP-3 are always predicted by the same three genes. Except for a couple of instances, this is also true for synuclein and STC2. Since the matrix gives collective connectivity across the full PGRN, these connectivity relations hold regardless of whether a network is a member of  $\mathcal{X}$  or  $\mathcal{Y}$ .

Matrices  $M_2$ ,  $M_3$  and  $M_4$ , summarize the connectivity information for classes  $\mathcal{Y}$ ,  $\mathcal{X}$  and  $\mathcal{Z}$ , respectively. Note the perfect consistency of predictors for pirin, MP-3, synuclein and STC2. Also note that STC2 is a predictor of WNT5A in only two of the networks.

Since we want to compare connectivity relations between  $\mathcal{X}$  and  $\mathcal{Y}$ , we consider the matrix  $M_5 = |M_3 - 4M_2|$ , where all operations are componentwise. The disparity in counts between  $\mathcal{X}$  and  $\mathcal{Y}$  has been balanced by multiplying  $M_2$  by 4. The absolute value is taken because we wish to consider the absolute discrepancy. Note that there are very few large discrepancies. The matter is further clarified by forming the matrix  $M_6$  by zeroing out all entries in  $M_5$  less than 5. This is done because a random variation of 1 in matrix  $M_2$  can create a difference of 4 in  $M_5$ . The fact that most of the entries in matrix  $M_6$  are 0 indicates a very strong congruency of connectivity between the two classes of networks having very different singleton attractors. There are really only three differences: WNT5A is predicted frequently by STC2 in  $\mathcal{X}$  but not in  $\mathcal{Y}$ , and it is predicted somewhat frequently (twice) by MMP-3 in  $\mathcal{Y}$  but never in  $\mathcal{X}$ . Analogous comments apply to PHO-C and MART-1 as targets. The key point here is that optimizing connectivity has led to fairly consistent connectivity; however, it has produced two classes of networks with significantly different attractors. This shows that focusing on connectivity does not necessarily preclude discovering a rich attractor structure. We conjecture that this phenomenon is in agreement with real networks because, even in widely varying contexts, one would not expect a significant change in connectivity.

Turning to the class  $\mathcal{Z}$  we let matrix  $M_7 = M_3 - M_4$ . Recalling that  $\mathcal{Z} \subset \mathcal{X}$ , we see that  $M_7$  gives the connectivity summary for the class difference  $\mathcal{X} - \mathcal{Z}$ . To compare  $\mathcal{Z}$  and  $\mathcal{X} - \mathcal{Z}$ , we define the matrix  $M_8 = |M_4 - (3/2)M_7|$ , where the factor  $3/2$  is necessary to correct the difference in class sizes. Matrix  $M_9$  is constructed by turning to 0 all entries in  $M_8$  less than 3. It is interpreted similarly to  $M_6$ . Note that it is practically all 0s. Even the PHO-C change is slight (3.5). The only significant change is for RET-1, with pirin and S100P being frequent predictors for class  $\mathcal{Z}$  but not for class  $\mathcal{X} - \mathcal{Z}$ , and PHO-C being a frequent predictor for  $\mathcal{X} - \mathcal{Z}$  but not for  $\mathcal{Z}$ . This means that occurrence of the attractor cycle  $C \leftrightarrow D$ , which possesses significant steady-state mass, happens with very little change in connectivity.

## 7 CONCLUSION

We have proposed a fully Bayesian approach to constructing PGRNs emphasizing network topology. The method computes the possible parent sets of each gene, the corresponding predictors, and the associated probabilities based on a nonlinear perceptron model, using a reversible jump MCMC technique, and a MCMC method is employed to search the network configurations. In proceeding with a topology-based approach, our hypothesis was that construction of transcriptional regulatory networks using a method that optimizes connectivity would capture features expected from biology. This has clearly been the result with regard to the considered melanoma study. When operated, the PGRN produces a steady-state distribution containing attractors that are either identical or very similar to the original observations. Moreover, many of the attractors are singletons, mimicking the biological propensity to stably occupy a given state. Most interestingly, the connectivity rules for the most optimal generated networks constituting the PGRN are remarkably similar, as would be expected for a network operating on a distributed basis, with strong interactions between the components.

## ACKNOWLEDGEMENTS

The authors would like to thank the anonymous reviewers for their careful reading of manuscript and their constructive comments. This research was supported by the National Human

Genome Research Institute and the Translational Genomics Research Institute. I.I. was supported by the National Cancer Institute under grant CA90301. X.W. was supported in part by the US National Science Foundation under grant DMS-0244583.

## REFERENCES

- Andrieu, C., Freitas, J. and Doucet, A. (2001) Robust full Bayesian learning for neural networks. *Neural Comput.*, **13**, 2359–2407.
- Bittner, M., Meltzer, P., Khan, M., Chen, Y., Jiang, Y., Seftor, E., Hendrix, M., Radmacher, M., Simon, R., Yakhini, Z. *et al.* (2000) Molecular classification of cutaneous malignant melanoma by gene expression profiling. *Nature*, **406**, 536–540.
- Friedman, N., Linial, M., Nachman, I. and Pe'er, D. (2000) Using Bayesian network to analyze expression data. *Comput. Biol.*, **7**, 601–620.
- Giudici, D. and Castelo, R. (2003) Improving Markov chain Monte Carlo model search for data mining. *Mach. Learning*, **50**, 127–158.
- Hartemink, A.J., Gifford, D.K., Jaakkola, T.S. and Young, R.A. (2001) Using graphical models and genomic expression data to statistically validate models of genetic regulatory networks. *Pacific Symp. Biocomput.*, **6**, 23–32.
- Ivanov, I. and Dougherty, E.R. (2004) Reduction mappings between probabilistic Boolean networks. *Appl. Signal Process.*, **4**, 125–131.
- Kauffman, S.A. (1993) *The Origins of Order: Self-organization and Selection in Evolution*. Oxford University Press, NY.
- Kim, S., Li, H., Chen, Y., Cao, N., Dougherty, E.R., Bittner, M.L. and Suh, E.B. (2002) Can Markov chain Models mimic biological regulation? *J. Biol. Sys.*, **10**, 337–357.
- Robert, C.P. and Casella, G. (1999) *Monte Carlo Statistical Methods*. Springer-Verlag, NY.
- Shmulevich, I., Dougherty, E.R., Kim, S. and Zhang, W. (2002a) Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks. *Bioinformatics*, **18**, 261–274.
- Shmulevich, I., Dougherty, E.R. and Zhang, W. (2002b) Gene perturbation and intervention in probabilistic Boolean networks. *Bioinformatics*, **18**, 1319–1331.
- Shmulevich, I., Dougherty, E.R. and Zhang, W. (2002c) From Boolean to probabilistic Boolean networks as models of genetic regulatory networks. *Proc. IEEE*, **90**, 1778–1792.
- Zhou, X., Wang, X. and Dougherty, E.R. (2003) Construction of genomic networks using mutual-information clustering and reversible-jump Markov-chain-Monte-Carlo predictor design. *Signal Processing*, **84**, 745–761.