Aniruddha Datta, Ranadip Pal, Ashish Choudhary, and Edward R. Dougherty

Control Approaches for Probabilistic Gene Regulatory Networks

What approaches have been developed for addressing the issue of intervention?

In recent years, there has been a considerable amount of interest in the area of genomic signal processing, which is the engineering discipline that studies the processing of genomic signals. Since regulatory decisions within the cell utilize numerous inputs, analytical tools are necessary to model the multivariate influences on decision-making produced by complex genetic networks. Signal processing approaches such as detection, prediction and classification have been used in the recent past to construct genetic regulatory networks capable of modeling genetic behavior. To accommodate the large amount of uncertainty associated with this kind of modeling, many of the networks proposed are probabilistic. One of the objectives of network modeling is to use the network to design different intervention approaches for affecting the time evolution of the gene activity profile of the network. More specifically, one is interested in intervening to help the network avoid undesirable states such as those associated with a disease.

INTRODUCTION

From a translational perspective, the ultimate objective of genetic regulatory network modeling is to use the network to design different approaches for affecting network dynamics in such a way as to avoid undesirable phenotypes, for instance, cancer. In this article we present a tutorial survey of some of the recent results on intervention in the context of probabilistic gene regulatory networks, which, owing to their original binary formulation and their usual application using binary and ternary gene-expression quantization, are generically called *probabilistic Boolean networks* (*PBNs*) [1]. These are essentially probabilistic generalizations of the standard *Boolean networks* (*BNs*) introduced by Kauffman [2]–[4] that allow the incorporation of uncertainty into the intergene relationships. Given a PBN, the transition from one state to the next takes place in accordance with certain transition probabilities and their dynamics, and hence intervention can be studied in the context of homogeneous Markov chains [5] with finite state spaces.

A major goal of functional genomics is to screen for genes that determine specific cellular phenotypes (disease) and model their activity in such a way that normal and abnormal behavior can be differentiated. The pragmatic manifestation of this goal is the development of therapies based on the disruption or mitigation of aberrant gene function contributing to the pathology of

1053-5888/07/$25.00©2007IEEE

a disease. Mitigation would be accomplished by the use of drugs to act on the gene products. Engineering therapeutic tools involves synthesizing nonlinear dynamical networks, analyzing these networks to characterize gene regulation, and developing intervention strategies to modify dynamical behavior. For instance, changes in network connectivity or functional relationships among the genes in a network, via mutations or rearrangements, can lead to steady-state behavior associated with tumorigenesis, and this is likely to lead to a cancerous phenotype unless corrective therapeutic intervention is applied.

To date, intervention studies have used three different approaches: i) resetting the state of the PBN, as necessary, to a more desirable initial state and letting the network evolve from there [6], ii) changing the steady-state (long-run) behavior of the network by minimally altering its rule-based structure [7], and iii) manipulating external (control) variables that alter the transition probabilities of the network and can, therefore, be used to desirably affect its dynamic evolution [8]. The control-theoretic approach has subsequently been extended. First, the optimal intervention algorithm has been modified to accommodate the case where the entire state vector, or *gene activity profile* (*GAP*) as it is known, is not available for measurement [9]. Second, whereas the original control-theoretic approach has been developed in the framework of *instantaneously random* PBNs, the intervention results have been extended to *context-sensitive* PBNs (terminology to be defined shortly) [10]. Third, in [11], control algorithms have been developed for a family of genetic regulatory networks as opposed to a single network. Finally, in [12], the earlier finite horizon results have been extended to the infinite horizon case in an effort to alter the steady-state behavior of the genetic regulatory network.

We will provide a tutorial exposition of the recent results in [11] and [12]. A similar exposition of the earlier intervention strategies has been provided in [13].

REVIEW OF PROBABILISTIC BNS

Our review focuses on aspects critical to intervention and we refer to the literature for more detailed accounts [1], [14], [15]. A BN consists of a set of nodes (genes) in which each gene can take on one of two binary values, 0 or 1 [2], [4]. Given n genes, the activity level [16] of gene i at time step k is denoted by $x_i(k)$, where $x_i(k) = 0$ indicates that gene i is not expressed and $x_i(k) = 1$ indicates that it is expressed. The overall expression levels of all the genes in the network at time step k is given by the state (row) vector $x(k) = [x_1(k), x_2(k), \ldots, x_n(k)]$, also called the GAP of the network at time k. Gene i evolves from time k to $k+1$ according to the Boolean function $f_i(x_1(k), x_2(k), \ldots, x_n(k))$. Usually the value of f_i does not depend on the entire set $\{x_1, x_2, \ldots, x_n\}$ of n gene values but only on a finite subset $\mathcal{P}_i$ of it. This set $\mathcal{P}_i$ is called the *predictor set* for the ith gene. Specifying the truth table for the functions $f_1, f_2, \ldots, f_n$ along with the associated predictor sets $\mathcal{P}_1$, $\mathcal{P}_2, \ldots, \mathcal{P}_n$ supplies all the information necessary to determine the time evolution of the states of the BN.

The binary n-digit state vector $x(k)$ can be mapped to positive integers $z(k)$ so that as $x(k)$ ranges from $00\cdots0$ to $11\cdots1$,$z(k)$ goes from 1 to 2^n. Here we employ the decimal representation $z(k)$ and the set $S = \{1, 2, \ldots, 2^n\}$ constitutes the state space for the BN. Furthermore, each $z(k)$ can be uniquely represented by a basis vector $w(k) \in \{0, 1\}^{2^n}$, where $w(k) = e_{z(k)}$, e.g., if $z(k) = 1$, then $w(k) = [1, 0, 0, \ldots..]$. Then, as discussed in [1], the evolution of the vector $w(k)$ proceeds according to the difference equation

$$w(k+1) = w(k)A \tag{1}$$

where A is a $2^n \times 2^n$ matrix having only one nonzero entry (equal to one) in each row.

A PBN consists of a finite collection of BNs over a fixed set of genes, where each BN is defined by a fixed network function. At any given moment of discrete time there is a probability p of randomly switching the state of the PBN, so that each constituent BN is a BN with random perturbation. Moreover, at each moment of time there is a probability q of switching to a different constituent BN, where, given a switch, each BN composing the network has a probability of being selected. If $q = 1$, the PBN is said to be *instantaneously random*, the idea being to model uncertainty in model selection; if $q < 1$, it is said to be *context-sensitive*, the idea being to model the situation where the model is affected by latent variables outside the model.

In the case of PBNs, we have a stochastic counterpart of (1) given by

$$w(k+1) = w(k)A \tag{2}$$

where $w(k)$ denotes the probability distribution vector at time k, i.e., $w_i(k) = Pr\{z(k) = i\}$ and A denotes the probability transition matrix.

Several approaches have been used for the derivation of the probability transition matrix (A) for PBNs. One of the approaches for discovering multivariate relationships between genes is using the CoD (coefficient of determination) technique [17]–[19]. The CoD procedure has been used in [8] and [9] to generate instantaneously random PBNs from actual gene expression data. In [10], a Bayesian connectivity-based approach [20] has been used to generate possible BNs for the construction of a context-sensitive PBN. The transition probabilities of the context-sensitive PBN were derived based on the structure and Bayesian scores of the individual BNs. The detailed derivation of the transition probabilities is available in [10]. Another approach to construct BNs from gene expression data is based on steady-state considerations and prescribed attractor structure [21]. This method has been used in [11] and [12] for the construction of BNs and subsequently PBNs.

As with the majority of the literature, we have focused on binary quantization; nevertheless, one should recognize that most of the theory and application carry over to any finite quantization in a fairly obvious fashion—that is, to *probabilistic gene regulatory networks* (PGRN). The terminology

probabilistic BN is usually applied to PGRNs in general, under the supposition that it is the logical character of the networks that is at issue.

EXTERNAL INTERVENTION BASED ON OPTIMAL CONTROL THEORY

PBNs can be used for studying the dynamic behavior of gene regulatory networks. Once a probability distribution vector has been specified for the initial state, the probability distribution vector evolves according to (2). From this perspective PBNs are *descriptive* in nature. There is no mechanism for controlling the evolution of the probability distribution vector. For treatment or intervention purposes, we are interested in working with PBNs in a *prescriptive* fashion, where the transition probabilities of the associated Markov chain depend on certain auxiliary variables, whose values can be chosen to make the probability distribution vector evolve in some desirable manner.

The use of such auxiliary variables makes sense from a biological perspective. For instance, in the case of diseases like cancer, auxiliary treatment inputs such as radiation and chemotherapy may be employed to move the state probability distribution vector away from one associated with uncontrolled cell proliferation or markedly reduced apoptosis. The auxiliary variables could also include genes that serve as external master-regulators for all the genes in the network. To be consistent with the binary nature of the expression status of individual genes in a PBN, we will assume that the auxiliary variables (*control inputs*) can take on only the binary values 0 or 1. The values of the individual control inputs can be changed from one time step to another in an effort to make the network behave in a desirable fashion.

Suppose a PBN has n genes and m control inputs, u_1, $u_2, \ldots, u_m$, each of which can take on only the binary values 0 or 1. One can represent the control input status using a decimal integer $v(k)$ ranging from 1 to 2^m, so that $U = \{1, \ldots, 2^m\}$ is the set of possible control actions. As shown in [8], the one-step evolution of the probability distribution vector in the case of a PBN containing 2^n states with control inputs takes place according to the equation

$$w(k+1) = w(k)A(v(k)) \tag{3}$$

where $w(k)$ is the 2^n dimensional state probability distribution vector and $A(v(k))$ is the $2^n \times 2^n$ control-dependent transition probability matrix. The system in (3) can be equivalently represented as a stationary discrete-time dynamic system

$$z(k+1) = f(z(k), v(k), d(k)), \ k = 0, 1, \ldots., \tag{4}$$

where for all k, the state $z(k)$ is an element of S, the control input $v(k)$ is an element of U, the disturbance $d(k)$ is an element of a space D and $f : S \times U \times D \mapsto S$. The disturbance $d(k)$ is manifested in terms of change of network based on the network transition probability q or change of state due to perturbation probability p. $d(k)$ is independent of prior disturbances $d(0), d(1) \ldots . d(k-1)$.We will interchangeably use either representation (3) or (4) depending on their suitability for a particular context or a particular derivation.

Since the transition probability matrix is a function of the control input $v(k)$, the evolution of the probability distribution vector of the PBN with control now depends not only on the initial distribution vector but also on the values of the control input at different time steps. Intuitively, it appears possible to make the states of the network evolve in a desirable fashion by appropriately choosing the control input at each time step. We next proceed to formalize these ideas.

Suppose that the number of steps over which the control input is to be applied has been a priori determined to be M and we are interested in controlling the behavior of the PBN over the interval $k = 0, 1, 2, \ldots, M-1$. Suppose at time step k, the state of the PBN is given by $z(k)$ and the corresponding control input is $v(k)$. (In the rest of this article, we will be referring to $z(k)$ as the state of the probabilistic BN since $z(k)$ is equivalent to the actual state $x(k)$.) Then we can define a cost $C_k(z(k), v(k))$ as being the cost of applying the control input $v(k)$ when the state is $z(k)$. The other component of the finite-horizon cost is the cost associated with the terminal state $z(M)$. Owing to the probabilistic nature of the evolution, the terminal state $z(M)$ is a random variable that can possibly take on any of the values $1, 2, \ldots, 2^n$. We assign higher terminal costs to the undesirable states. For instance, a state associated with rapid cell proliferation leading to cancer should be associated with a high terminal penalty while a state associated with normal behavior should be assigned a low terminal penalty. For the purposes of this section, we will assume that the assignment of terminal penalties has been carried out and we have at our disposal a terminal penalty $C_M(z(M))$ that is a function of the terminal state.

With these definitions, the finite-horizon cost to be minimized is given by

$$E\left[\sum_{k=0}^{M-1} C_k(z(k), v(k)) + C_M(z(M))|z(0)\right]. \tag{5}$$

To proceed further, let us assume that at time k the control input $v(k)$ is a function of the current state $z(k)$, namely,

$$v(k) = \mu_k(z(k)) \tag{6}$$

where $\mu_k : \{1, 2, \ldots., 2^n\} \to \{1, 2, \ldots, 2^m\}$. The optimal control problem can now be stated. Given an initial state $z(0)$, find a control law $\pi = \{\mu_0, \mu_1, \ldots.., \mu_{M-1}\}$ that minimizes the cost functional

$$J_0(z(0)) = E\left[\sum_{k=0}^{M-1} C_k(z(k), \mu_k(z(k))) + C_M(z(M))\right] \tag{7}$$

subject to the constraint

$$Pr\{z(k+1) = j | z(k) = i, v(k) = v\} = a_{ij}(v) \tag{8}$$

where $a_{ij}(v)$ is the ith row, jth column entry of the matrix $A(v)$.

SOLUTION USING DYNAMIC PROGRAMMING

Optimal control problems of the type described by (7) and (8) can be solved using the technique of dynamic programming. This technique, pioneered by Bellman in the 1960s, is based on the so-called principle of optimality. When an optimal strategy exists, the principle of optimality asserts: if one searches for an optimal strategy over a subset of the original number of steps, then this new optimal strategy will be given by the overall optimal strategy, restricted to the steps being considered. It can be used to arrive at the dynamic programming [22], [23], [8] solution to (7) and (8):

$$J_M(z_M) = C_M(z_M) \tag{9}$$

$$J_k(z_k) = \min_{v_k \in U} \left\{ C_k(z_k, v_k) + \sum_{j=1}^{2^n} a_{z_k,j}(v_k) J_{k+1}(j) \right\} \tag{10}$$

for $k = 0, 1, \ldots, M-1$. (In the rest of this article, we will be denoting $w(k)$, $z(k)$, $v(k)$, $d(k)$ by w_k, z_k, v_k, d_k respectively, mainly for the purpose of simplifying the notation.)

Illustrative examples to show optimal control design for instantaneously random PBNs are available in [8]. In the following section, we provide an example of applying dynamic programming over finite time steps for a context-sensitive PBN derived from actual gene expression data.

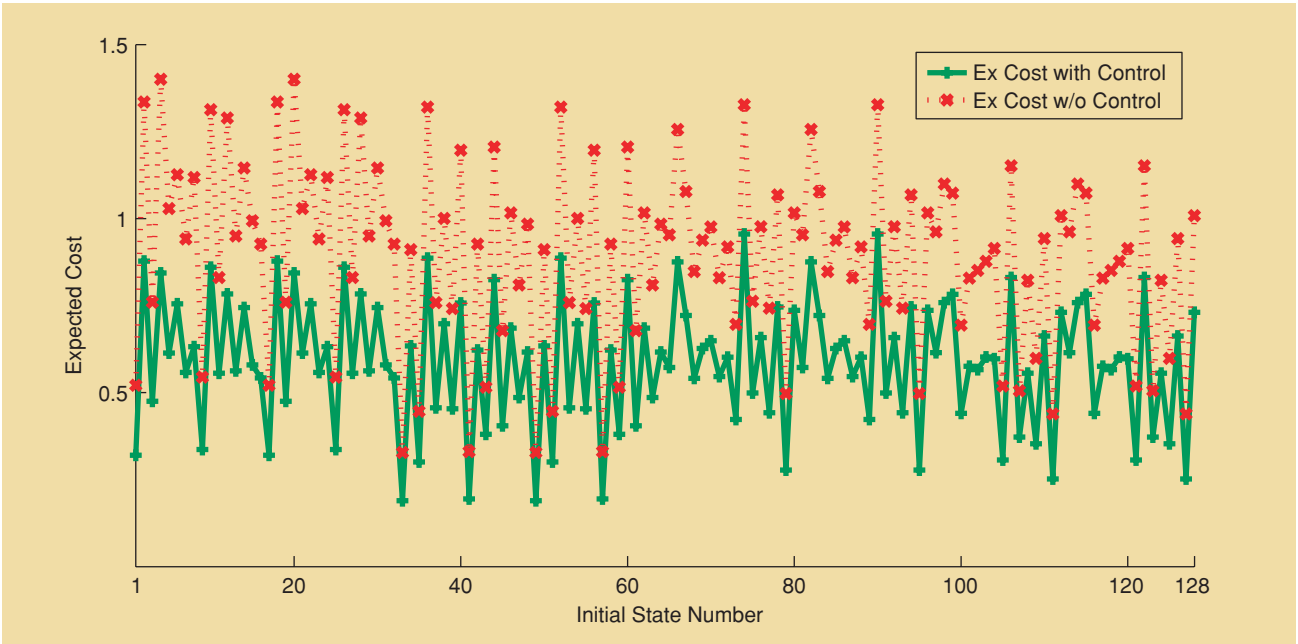
MELANOMA EXAMPLE

The network chosen as an example of how control might be applied is one developed from data collected in a study of metastatic melanoma [24]. In this expression profiling study, the abundance of messenger RNA for the gene WNT5A was found to be a highly discriminating difference between cells with properties typically associated with high metastatic competence versus those with low metastatic competence. These findings were validated and expanded in a second study [25]. Because it is biologically known that WNT5A ceasing to be down regulated is strongly predictive of the onset of metastasis, the control objective for this network is to externally down-regulate the WNT5A gene.

We consider a seven-gene network with genes WNT5A, pirin, S100P, RET1, MART1, HADHB, and STC2. We have used the Bayesian connectivity-based approach of [20] to construct four highly probable BNs that are used as the constituent BNs in the PBN, with their selection probabilities based on their Bayesian scores. The control strategy of this section has been applied to the designed PBN with pirin chosen as the control gene and $p = q = 0.01$. Figure 1 shows the expected cost for a finite horizon problem of length 5 originating from each of the 128 states (Note that the choice of $M = 5$ is arbitrary.) The cost of control is assumed to be 0.5 and the states are assigned a terminal penalty of 5 if WNT5A is 1 and 0 if WNT5A is 0. Here the choice of the numbers 0 and 5 is arbitrary but it does reflect our attempt to capture the intuitive notion that states where WNT5A equals one are less desirable than those where WNT5A equals 0. The control objective is to down-regulate the WNT5A gene. From Figure 1, it is clear that the expected cost with control is much lower than that without control, which agrees with our objective.

EXTERNAL INTERVENTION FOR A FAMILY OF BNS

The results of this section are motivated by the fact that most gene expression data used for PBN design are likely to come from the phenotype observed at steady-state. For instance, the gene expression data for cancer genomics studies are usually obtained from tumor biopsies. Given a data set consisting of



[FIG1] Expected cost for a finite horizon problem of length five originating from the different initial states [10].

gene-expression measurements, PBN design constitutes an ill-posed inverse problem that is treated by using a design algorithm to generate a solution. Inference can be formalized by postulating criteria that constitute a solution space for the inverse problem. The criteria come in two forms: (1) the *constraint criteria* are composed of restrictions on the form of the network, and (2) the *operational criteria* are composed of relations that must be satisfied between the model and the data. The solution space consists of all PBNs that satisfy the two sets of criteria. Recognizing that PBNs are composed of BNs, and since it is difficult to infer the probabilistic structure among the constituent BNs from the steady-state data typically used for design, a more general view may be taken in which the inverse problem is restricted to determining a solution space of BNs and then finding networks in that space [21]. Without a probabilistic structure between the BNs, we have a family of BNs satisfying both the constraint and operational criteria. If desired, one can then go further and construct a PBN by using networks from the family, or one can simply treat the family as a collection of solutions to the Boolean-network inverse problem.

A MAJOR GOAL OF FUNCTIONAL GENOMICS IS TO SCREEN FOR GENES THAT DETERMINE SPECIFIC CELLULAR PHENOTYPES (DISEASE) AND MODEL THEIR ACTIVITY IN SUCH A WAY THAT NORMAL AND ABNORMAL BEHAVIOR CAN BE DIFFERENTIATED.

In [11], we derived a control algorithm that can be applied to the second situation, i.e., to a family of BNs. This is accomplished by minimizing a composite cost function that is a weighted average cost over the entire family. Ideally, the weighting for each member of the family at any time point would be proportional to the instantaneous probability of a particular network being the governing network. Although these instantaneous probabilities are not known, we adaptively estimate them from the available data and the estimate is used to implement the control algorithm. We next provide a tutorial presentation of the details.

If a family of BNs is designed whose attractors match the data, assuming the family is not too small we have the expectation that the underlying biological phenomena are closely modeled by at least some of the BNs in the family. In the absence of perfect knowledge as to which BNs are capable of better representing the underlying phenomena, we develop a control policy that optimizes a composite cost function over the entire family of BNs.

Toward this end, let $\mathcal{N}$ be a set of L BNs $N_1, N_2, \ldots, N_L$ possessing identical sets of singleton attractors, all sharing the same state space S and the same control space U. Associated with each network is an initial probability of it representing the underlying phenomenon. Since this information is not available, we will adaptively estimate these probabilities as more transitions are observed. For each network N_l, $l = 1, 2, \ldots, L$ define:

- $a_{ij}^l(v)$ to be the ith row, jth column entry of the matrix $A^l(v)$ of the network N_l;
- $C_k^l(i, v)$ to be the cost of applying the control v at the kth time step in state i in network N_l;
- $C_M^l(i)$ to be the terminal cost associated with state i in network N_l.

We define the *belief vector* $\pi_k = [\pi_k^1, \pi_k^2, \ldots, \pi_k^L]$, where π_k^l is the probability of network N_l being the underlying network at the kth time step. π_k is the probability distribution vector for the family of networks at the kth time step. Since π_k is unknown, we will make an initial guess for it and update it as more information becomes available. The use of this vector is inspired by the *information vector* in [26].

Suppose i is the current state at step k, π is the current estimate of the belief vector, and upon application of control v we observe state j at the next time step. Then the new belief vector is $\pi' = T(\pi, i|j, v)$, where the transformation T can be obtained by use of Bayes' theorem and the theorem of total probability,

$$\pi' = \left[\cdots, \frac{a_{ij}^l(v) \cdot \pi_k^l}{\sum_{s \in \mathcal{N}} a_{ij}^s(v) \cdot \pi_k^s}, \cdots \right] \tag{11}$$

We will now make use of this belief vector to set up the optimal control solution over a family of BNs. Suppose we are given an initial belief vector π_0 and an initial state z_0. The initial belief vector is based on our prior knowledge of the system. It could be a function of likelihood or Bayesian scores of networks, or it could be uniform to reflect no prior knowledge. Our objective is to find controls $v_0, v_1, \ldots, v_k, \ldots, v_{M-1}$ to minimize the expectation of the cost-to-go function over all networks in $\mathcal{N}$. The cost to go at the kth time step ($0 \le k < M$) is a function of the current state z_k and the updated belief vector π_k. Motivated by (10) for the single PBN case, we define the average optimal cost-to-go function by

$$J_k(\pi_k, i) = \min_{v \in U} \left[\sum_{l \in \mathcal{N}} \pi_k^l \left\{ C_k^l(i, v) + \sum_{j \in S} a_{ij}^l(v) J_{k+1}(T(\pi_k, i|j, v), j) \right\} \right] \tag{12}$$

The inner summation is the expectation over all $j \in S$ of the cost to go at the $(k+1)$th step in the lth network on observing j. We then add to it the cost of control at the kth step in the lth network and average over all the networks in the family. Finally we take the minimum over all control actions in U to obtain the optimal policy and the cost to go at the kth step.

Similarly, in view of (9), the terminal cost for a state i is trivially defined to be the average terminal cost over the entire family:

$$J_M(\pi_M, i) = \sum_{l \in \mathcal{N}} \pi_M^l . C_M^l(i). \tag{13}$$

In the melanoma example of the previous section, terminal penalties were assigned to states based on the expression level of a certain key gene, namely WNT5A; however, as discussed in [27], it may be more reasonable to assign terminal penalties based on the long-term prospective behavior of the system in the absence of control. Using the procedure in [27], for singleton attractors the penalty C_M^l is set according to the status of the *penalty gene*(s). A penalty gene is a gene for which certain expression statuses are known to be undesirable, e.g., WNT5A for the melanoma example. The attractors are shared by each network in the family and will have the same penalty across the different networks. Penalties for nonattractor states will differ across networks, depending on the particular attractor in whose basin that nonattractor state may happen to lie in.

THE ULTIMATE OBJECTIVE OF GENETIC REGULATORY NETWORK MODELING IS TO USE THE NETWORK TO DESIGN DIFFERENT APPROACHES FOR AFFECTING NETWORK DYNAMICS IN SUCH A WAY AS TO AVOID UNDESIRABLE PHENOTYPES.

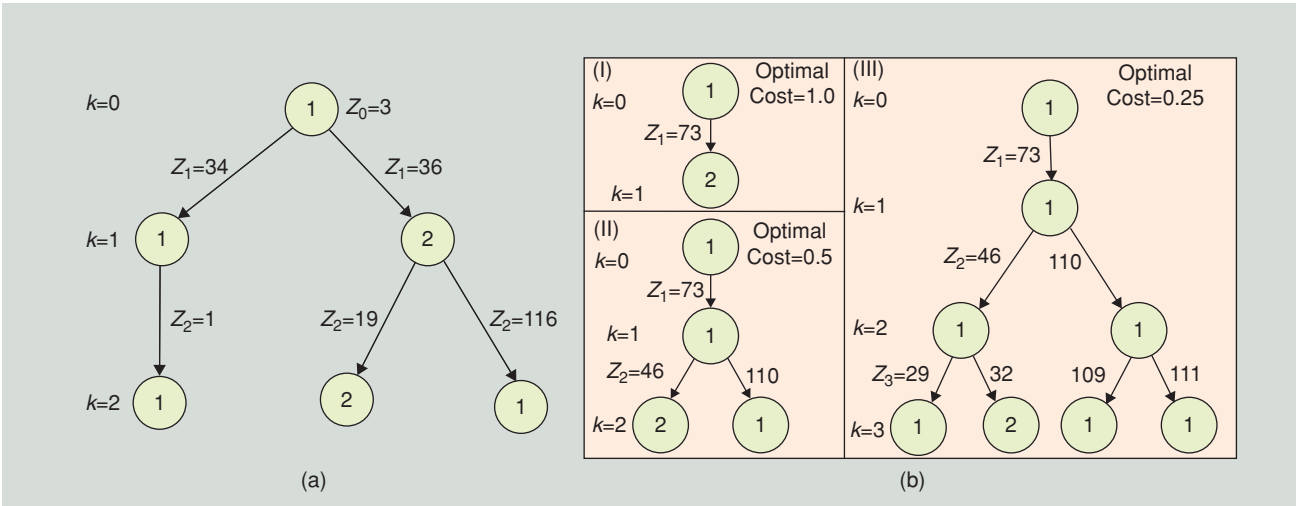
MELANOMA EXAMPLE

Here we apply the methodology of this section to the same melanoma data considered earlier. As before, a family of networks with seven genes: PIRIN, S100P, RET1, MART1, HADHB, STC2 and WNT5A is constructed. Since all the available 31 data points correspond to steady-state behavior, they should be considered as attractors in the networks. However, out of the 31 samples only 18 were distinct. To reduce the number of attractors, we form seven clusters from the data points and treat the cluster centers as attractors. These attractors are classified into two categories, GOOD and BAD, depending on the status of the WNT5A gene [27]

Using the procedure of [21], we obtain four distinct BNs (N_1, N_2, N_3, N_4) with the same set of seven attractors. These networks are available in [27]. We assigned a penalty of five to all states in the basin of the undesirable attractors (WNT5A $= 1$) and 0 to all the other states. We used PIRIN as the control gene with cost of control equal to one. A forcible alteration in the expression level of PIRIN is associated with $v = 2$ while $v = 1$ represents no control.

To present the results, we make use of policy trees where the number inside each circle represents the optimal control action and the arc following each circle corresponds to the next observed state which leads to the next optimal control action. A policy tree for $M = 3$ with initial belief vector $\pi_0 = [1/4, 1/4, 1/4, 1/4]$ and initial state $z_0 = 3$ is shown in Figure 2(a).

For purposes of comparison, three different policies are considered side by side: Pol^{TR} being the optimal policy of this section; Pol^1, Pol^2, Pol^3, Pol^4 being the optimal policies tuned to the individual BNs N_1, N_2, N_3 and N_4 respectively; and Pol^{SW} being the policy obtained for a PBN in which each BN is assigned equal selection probability. The expected cost is 0.75 when we control using Pol^{TR}, 1.5 when using Pol^{SW}, and 1.75, 2.5, 1.5 and 1.75 when using Pol^1, Pol^2, Pol^3 and Pol^4, respectively. The expected uncontrolled cost is 2.5. For all horizons M and all initial states $z_0 = i \in S$ the method of this section is superior to the other methods considered. Out of the 128 states in the network, 89 states needed to be controlled in at least one of the four networks. In particular for $M = 5$, starting from such states, Pol^{TR} was more effective than Pol^{SW} in reducing the cost by 0.1152 on average. In terms of absolute probabilities Pol^{TR} was able to take the system to a desirable attractor starting from all initial states and all networks with a probability 1.0, except for states 4, 36, 68, and 100 in network N_2, which are uncontrollable from PIRIN. For Pol^{SW}, states 4,



[FIG2] (a) Policy tree for $M = 3$, initial state $z_0 = 3$ and initial belief vector $\pi_0 = [1/4, 1/4, 1/4, 1/4]$, [11]. (b) Policy Trees and optimal costs, for initial state $z_0 = 93$, $\pi_0 = [1/4, 1/4, 1/4, 1/4]$, $M = 2$ (I), $M = 3$ (II) and $M = 4$ (III) [11].

8, 24, 36, 68, and 100 are not taken to a desirable attractor in N_2. In the event of N_2 being the underlying network, starting from states 4, 36, 68, and 100, Pol^{TR} recognizes this and gives up promptly, while Pol^{SW} keeps on applying control, incurring extra costs, without any extra benefit. Policy trees for initial state $z_0 = 93$, $\pi_0 = [1/4, 1/4, 1/4, 1/4]$, and $M = 2, 3, 4$ are shown in Figure 2(b). The expected cost with $M = 2$ is 1.0 that can be further reduced to 0.25 if $M \geq 4$. This is reasonable because the algorithm has more time steps to identify and control the system.

EXTERNAL INTERVENTION IN THE INFINITE HORIZON CASE

The external control approaches presented so far have all focused on manipulating external (control) variables that affect the transition probabilities of a PBN to desirably affect its dynamic evolution over a finite time horizon. These short-term policies are not always effective in changing the steady-state behavior of the PBN, even though they can change the dynamical performance of the network for a small number of stages. Motivated by this, in [12] we considered intervention via external control variables in PBNs over an infinite length of time. We derived a control policy that does not change from one time step to the next because implementations of such stationary policies are often simple and stationary policies can be used to shift the steady-state distribution from undesirable states to desirable ones. We next present a tutorial discussion of the results obtained.

We first note that the problem formulation and results summarized earlier for the finite horizon case serve to motivate the infinite horizon developments here. Consider the finite horizon cost function being minimized in (7) and suppose that the control horizon characterized by M is made larger and larger and in the limit we would like for it to tend to infinity. In trying to do so, we immediately encounter a number of potential obstacles that did not arise in the finite horizon case.

First, in the infinite horizon problem, the control horizon is infinite and therefore there is no terminal state or its associated terminal penalty. Consequently, for the infinite horizon case, the cost per stage should incorporate the penalty of the state along with the cost of control. $C_k(i, v)$ of the finite horizon problem should now be replaced by $\tilde{C}(i, v, j)$ so that the per stage cost takes into account the origin, the destination and the control.

Second, in the finite horizon problem, the summation in (7) is a finite one and so the quantity being minimized is finite. If we let the control horizon go to infinity, there is a possibility that the summation of the one stage costs may go to infinity (for all controls) leading to an ill-posed optimization problem. To make the optimization problem well posed, the cost considered in (7) has to be modified before letting the length M of the control horizon tend to infinity. There are a couple of approaches for doing this. We will use the approach referred to in the literature as the problem of total cost with discounted and bounded cost per stage.

We assume that the cost per stage $\tilde{C}(i, v, j)$ is bounded $\forall\, i, j \in S$ and $v \in U$ and a discounting factor $\alpha \in (0, 1)$ is introduced in the cost to make sure that the limit of the finite sums converges as the horizon length goes to infinity. More specifically, our objective is to find a policy $\pi = \{\mu_0, \mu_1 \ldots\ldots\}$, where $\mu_k : S \to U, k = 0, 1\ldots.$, that minimizes the cost function

$$J_\pi(z_0) = \lim_{M\to\infty} E\left\{\sum_{k=0}^{M-1} \alpha^k \tilde{C}(z_k, \mu_k(z_k), d_k)\right\}, \qquad (14)$$

where the cost per stage $\tilde{C} : S \times U \times D \to \Re$ is given. (Note that a Markov chain can be modeled by $z_{k+1} = d_k$ [23]. Hence the destination state is the same as the disturbance d_k.) In the general formulation, the inclusion of α in the cost captures the fact that costs incurred at a later time are less significant. In the case of cancer treatment, $\alpha < 1$ signifies that the condition of the patient in the initial stages of treatment is more important than the condition at a later stage, or in other words, the reward for improving the condition of the patient in the present is more significant than the reward obtained from similar improvement at a later stage. This approach is reasonable if we keep in mind the expected life span of the patient. Another approach for ensuring that the infinite horizon cost is well defined is referred to as the average cost per stage formulation and has been considered in [12].

We next present the solution to the discounted and bounded cost per stage problem.

Let us denote by Π the set of all admissible policies π, i.e.,, the set of all sequences of functions $\pi = \mu_0, \mu_1, \ldots.$ with $\mu_k(z) : S \to U, k = 0, 1, \ldots\ldots$ The optimal cost function J^* is defined by

$$J^*(z) = \min_{\pi\in\Pi} J_\pi(z), z \in S. \qquad (15)$$

A stationary policy is an admissible policy of the form $\pi = \mu, \mu, \ldots.$, and its corresponding cost function is denoted by J_μ. We say that the stationary policy $\pi = \mu, \mu\ldots.$ is optimal if $J_\mu(z) = J^*(z)$ for all states z.

OPTIMAL CONTROL SOLUTION

In this section, we solve the problem of minimizing the cost in (14) under the assumption that the cost per stage $\tilde{C}(i, v, d)$ is bounded, i.e., $\exists\, B > 0$ such that $\tilde{C}$ satisfies $|\tilde{C}(z, v, d)| \leq B$, for all $(z, v, d) \in S \times U \times D$. In the case of context-sensitive PBNs, $\tilde{C}(i, v, d)$ is bounded since the control and disturbance spaces are finite.

Observe that if we set $C_M(z_M) = 0\ \forall\, z_M \in S$ and $C_k(z_k, v_k) = \alpha^k C(z_k, v_k)$ (where $C(z_k, v_k)$ is the expectation of $\tilde{C}(z_k, v_k, d_k)$ over d_k) in the finite horizon problem of (7) and let $M \to \infty$, then we obtain the infinite horizon cost function considered in (14). Thus it seems reasonable that the finite horizon solution described by (9) and (10) could provide a basis for arriving at the solution of the optimization problem in (15) where J_π is given by (14). A formal derivation of this connection is given in [23]. Here we simply state the result and present an intuitive justification for it.

Towards this end, note that (10) in the dynamic programming algorithm basically describes how the optimal cost J_{k+1}

propagates backwards in time to the optimal cost J_k in the finite horizon problem of (7). For the cost function considered in (14), it is clear that the cost J_{k+1} must be discounted by the factor α while being propagated to the previous stage. Consequently, for the optimal control problem of this section, (10) will have to be replaced by

$$J_k(i) = \min_{v \in U} \left[C(i, v) + \alpha \sum_{j=1}^{2^n} a_{ij}(v) J_{k+1}(j) \right]. \quad (16)$$

The above equation motivates the introduction of the following mapping. For any cost function $J : S \to \Re$, define the mapping $TJ : S \to \Re$ by

$$(TJ)(i) = \min_{v \in U} \left[C(i, v) + \alpha \sum_{j=1}^{2^n} a_{ij}(v) J(j) \right], \ i \in S. \quad (17)$$

Note that TJ is the optimal cost function for the one-stage (finite horizon) problem that has stage cost C and terminal cost αJ.

It can be shown [23] that the optimal cost function J^* is the unique fixed point of the map T and the iteration $J_{k+1} = TJ_k$ converges to J^* as $t \to \infty$. This provides us with a computational algorithm for determining the optimal cost function by running the recursion

$$J_{k+1} = TJ_k, \ k = 0, 1, 2, \ldots \quad (18)$$

for any bounded initial cost function $J_0 : S \to \Re$. The iteration described in (18) above is referred to as the value iteration procedure since at every stage we are iterating on the values of the cost function and the optimal policy simply falls out as a by product when the iteration converges to the optimal value of the cost function. The details of the optimality and convergence of the value iteration procedure, along with that of another computational algorithm referred to as policy iteration, are available in [23].

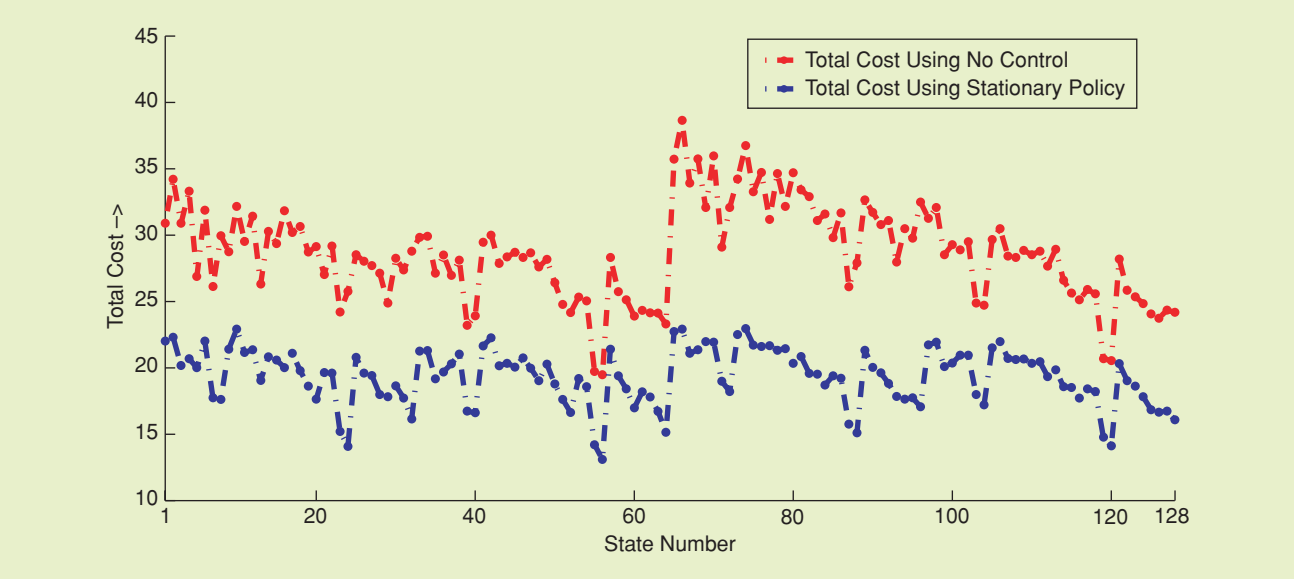
THE DYNAMIC PROGRAMMING TECHNIQUE, PIONEERED BY BELLMAN IN THE 1960S, IS BASED ON THE SO-CALLED PRINCIPLE OF OPTIMALITY.

MELANOMA EXAMPLE

In this section, we apply the infinite horizon control policy to a context-sensitive PBN derived from the same gene expression data as before. The network contains the seven genes WNT5A, pirin, S100P, RET1, MART1, HADHB and STC2. In this case, to obtain the PBN, we have used the algorithms described in [21] to construct four highly probable BNs to use as the constituent BNs in the PBN. The states are ordered as WNT5A, pirin, S100P, RET1, MART1, HADHB and STC2, with WNT5A as the most significant bit (MSB) and STC2 as the least significant bit (LSB).

The control strategy of this section is applied to the designed PBN with pirin chosen as the control gene ($v = 2$ signifying the state of pirin is reversed and $v = 1$ signifying no intervention) and $p = q = 0.01$ (see the p and q definitions). Since our objective is to down regulate the WNT5A gene, a higher penalty of five is assigned for destination states having WNT5a up-regulated. Also for a given WNT5A status for the destination state, a higher penalty is assigned when the control is active (control cost assumed to be one) versus when it is not. Further details on the penalty assignment are available in [12].

Figure 3 shows the total cost for the discounted cost function with bounded cost per stage originating from each of the



[FIG3] **Total cost originating from different initial states [12].**

128 states, after the iterations have converged, with the discount factor α chosen to be 0.9. As before, the control objective is to down regulate the WNT5A gene. From Figure 3, it is clear that the total cost with an optimal stationary policy is much lower than that without control, which agrees with our objective.

Figure 4(a) shows the steady-state distribution of the PBN using the obtained stationary policy, and (b) shows the original PBN steady-state distribution for comparison purposes. We should note that the states from 1–64 have WNT5A 0 and hence are desirable states, as compared to states 65–128 that have WNT5A 1 and hence are undesirable. The steady-state distribution Figure 4(a) and (b) shows that the stationary policy has enabled us to shift the probability mass from the bad states to states with lower metastatic competence. For example, state 66 (WNT5A is 1) has a high probability mass (0.15) in the original steady state but stationary control has reduced its steady-state mass to 0.04. Similarly, the probability mass of state 64 (desirable state) is high when using the stationary policy.

CONCLUDING REMARKS

We have discussed several approaches that have been recently developed for addressing the issue of intervention in probabilistic gene regulatory networks. The results reported indicate that significant progress has been made in this area; however, numerous open issues remain, and these will have to be successfully tackled before the methods suggested in this article find application in actual clinical practice. We next discuss some of the issues that we are aware of at the current time.

METHODICAL ASSIGNMENT OF TERMINAL PENALTIES

The formulation of the optimal control problem assumes that there is a terminal penalty associated with each state of the PBN; however, assignment of these terminal penalties for cancer therapy is by no means a straightforward task. The reason is that while the intervention will be carried out only over a finite horizon, one would like to continue to enjoy the benefits in the steady state. For such purposes, the kind of terminal penalty used for the melanoma cell line study is inadequate since it fails to capture the steady-state behavior once the intervention has ceased. To remedy the situation, we propose to assign terminal penalties based on equivalence classes. The results of preliminary simulation studies in this regard [27] appear to be encouraging.

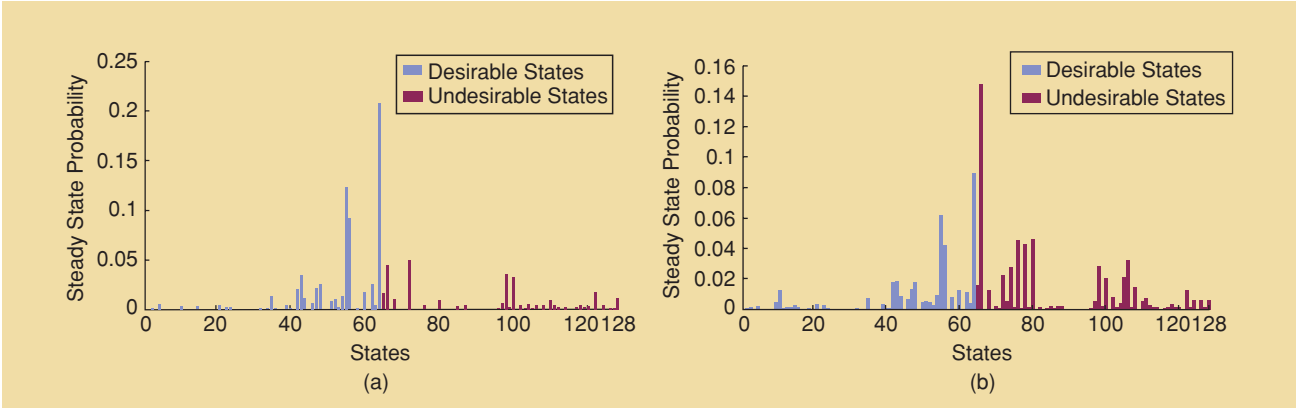
CHOICE OF CONTROL INPUT

In the case of the melanoma cell line studies presented here, one of the genes in the PBN, namely pirin, has been used as a control input. The question is how to decide which gene to use. Of course, one consideration is to use genes for which inhibitors or enhancers are readily available. However, even if such a gene is chosen, how can we be certain that it is capable of controlling some other gene(s)? Although the answer is not clear at this stage, we do believe that the traditional control theoretic concept of *controllability* [28] may yield some useful insights. Another possibility is to use the concept of gene *influence* introduced in [1], an approach that we have preliminarily explored in [10].

ROBUSTNESS OF THE CONTROL STRATEGIES

The control algorithms presented in this article have all been analyzed assuming that the PBN model perfectly captures the actual behavior of the gene regulatory network. Since errors between the PBN model and the actual gene regulatory network are inevitable, the designed control algorithms will have to be robust to modeling errors if there is to be any hope of success upon actual implementation. Such robustness considerations have dominated the control literature for more than two decades now and we believe that some of the results obtained could be exploited in the context of application to genetic regulatory networks.

The optimal control results presented in this article assume known transition probabilities and pertain to a problem of known length for the finite-horizon case. Their extension to the situation where the transition probabilities and the horizon length are unknown is a topic for further investigation. Finally, the results presented here correspond to the following stages in standard control design: modeling, controller design,



[FIG4] **(a) Steady state using discounted cost stationary policy [12] and (b) original steady state [12].**

and verification of the performance of the designed controller via computer simulations. The designed controllers will have to be successfully implemented in practical studies, at least with cancer cell lines, to validate the use of engineering approaches in translational medicine. A considerable amount of effort needs to be focused on this endeavor.

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation under Grants ECS-0355227 and CCF-0514644, the National Cancer Institute under Grant CA 90301, the National Human Genome Research Institute, the University of Texas M. D. Anderson Cancer Center, and the Translational Genomics Research Institute.

AUTHORS

Aniruddha Datta (datta@ece.tamu.edu) is a professor in the Department of Electrical and Computer Engineering at Texas A&M University, College Station. He holds a B.Tech. degree in electrical engineering from the Indian Institute of Technology, Kharagpur, an M.S.E.E. degree from Southern Illinois University, Carbondale, and an M.S. (applied mathematics) and Ph.D. degrees from the University of Southern California. He is the author of three books in the controls area and has authored more than 80 journal and conference papers. His areas of interest include adaptive control, robust control, PID control, and genomic signal processing.

Ranadip Pal (ranadip@ece.tamu.edu) received the B.Tech. degree in electronics and electrical communication engineering from the Indian Institute of Technology, Kharagpur, in 2002 and the M.S. degree in electrical engineering from Texas A&M University, College Station, in 2004. He is currently working toward the Ph.D. degree in electrical and computer engineering at Texas A&M University. His research areas include computational biology, genomic signal processing, and control of genetic regulatory networks.

Ashish Choudhary (cdry@ece.tamu.edu) received the B.Tech. degree in electrical engineering from Indian Institute of Technology Bombay, Mumbai, in 2001 and the M.S. and Ph.D. degrees in electrical engineering from Texas A&M University in 2003 and 2006, respectively. His research areas are genomic signal processing, morphological image processing, statistical learning, and control.

Edward R. Dougherty (edward@ece.tamu.edu) is a professor in the Department of Electrical and Computer Engineering and director of the Genomic Signal Processing Laboratory at Texas A&M University in College Station. He is also director of the Computational Biology Division of the Translational Genomics Research Institute in Phoenix. He holds a Ph.D. in mathematics from Rutgers University and an M.S. in computer science from Stevens Institute of Technology. He is author of 12 books and more than 190 journal articles and the editor of five books. He is an SPIE fellow and a recipient of the SPIE President's Award.

REFERENCES

[1] I. Shmulevich, E.R. Dougherty, S. Kim, and W. Zhang, "Probabilistic Boolean networks: A rule-based uncertainty model for gene regulatory networks," *Bioinformatics*, vol. 18, no. 2, pp. 261–274, 2002.

[2] S. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," *Theor. Biol.*, vol. 22, no. 3, pp. 437–467, 1969.

[3] S. Kauffman and S. Levin, "Towards a general theory of adaptive walks on rugged landscapes," *Theor. Biol.,* vol. 128, no. 1, pp. 11–45, 1987.

[4] S. Kauffman, *The Origins of Order: Self-Organization and Selection in Evolution*. New York: Oxford Univ. Press, 1993.

[5] P.G. Hoel, S.C. Port, and C.J. Stone, *Introduction to Stochastic Processes*, Long Grove, Illinois: Waveland Press, 1986.

[6] I. Shmulevich, E.R. Dougherty, and W. Zhang, "Gene perturbation and intervention in probabilistic Boolean networks," *Bioinformatics*, vol. 18, no. 10, pp. 1319–1331, 2002.

[7] I. Shmulevich, E.R. Dougherty, and W. Zhang, "Control of stationary behavior in probabilistic Boolean networks by means of structural intervention," *Biol. Syst.*, vol. 10, no. 2, pp. 431–446, 2002.

[8] A. Datta, A. Choudhary, M.L. Bittner, and E.R. Dougherty, "External control in markovian genetic regulatory networks," *Mach. Learn.*, vol. 52, no. 1-2, pp. 169–191, 2003.

[9] A. Datta, A. Choudhary, M.L. Bittner, and E.R. Dougherty, "External control in markovian genetic regulatory networks: The imperfect information case," *Bioinformatics*, vol. 20, no. 6, pp. 924–930, 2004.

[10] R. Pal, A. Datta, M.L. Bittner, and E.R. Dougherty, "Intervention in context-sensitive probabilistic Boolean networks," *Bioinformatics*, vol. 21, no. 7, pp. 1211–1218, 2005.

[11] A. Choudhary, A. Datta, M.L. Bittner, and E.R. Dougherty, "Intervention in a family of Boolean networks," *Bioinformatics*, vol. 22, no. 2, pp. 226–232, 2006.

[12] R. Pal, A. Datta, and E.R. Dougherty, "Optimal infinite horizon control for probabilistic Boolean networks," *IEEE Trans. Signal Processing*, vol. 54, no. 6, pp. 2375-2387, 2006.

[13] A. Datta, R. Pal, and E.R. Dougherty, "Intervention in probabilistic gene regulatory networks," *Current Bioinformatics*, vol. 1, no. 2, pp. 167–184, 2006.

[14] I. Shmulevich and E. Dougherty, *Modeling Genetic Regulatory Networks with Probabilistic Boolean Networks*. New York: Hindawi, 2005.

[15] I. Shmulevich, E.R. Dougherty, and W. Zhang, "From Boolean to probabilistic Boolean networks as models of genetic regulatory networks," *Proc. IEEE*, vol. 90, no. 11, pp. 1778–1792, 2002.

[16] B. Alberts, A. Johnson, J. Lewis, M. Raff, D. Bray, K. Hopkins, K. Roberts, P. Walter, *Essential Cell Biology*, 2nd ed. New York: Garland, 2003.

[17] E. Dougherty, S. Kim, and Y. Chen, "Coefficient of determination in nonlinear signal processing," *Signal Process.*, vol. 80, no. 2, pp. 2219–2235, 2000.

[18] S. Kim, E. Dougherty, Y. Chen, K. Sivakumar, P. Meltzer, J. Trent, M. Bittner, "Multivariate measurement of gene-expression relationships," *Genomics*, vol. 67, pp. 201–209, 2000.

[19] S. Kim, E. Dougherty, M. Bittner, Y. Chen, K. Sivakumar, P. Meltzer, J. Trent, "A general framework for the analysis of multivariate gene interaction via expression arrays," *Biomed. Optics*, vol. 4, no. 4, pp. 411–424, 2000.

[20] X. Zhou, X. Wang, R. Pal, I. Ivanov, M.L. Bittner, and E.R. Dougherty, "A bayesian connectivity-based approach to constructing probabilistic gene regulatory networks," *Bioinformatics*, vol. 20, no. 17, pp. 2918–2927, 2004.

[21] R. Pal, I. Ivanov, A. Datta, M.L. Bittner, and E.R. Dougherty, "Generating boolean networks with a prescribed attractor structure," *Bioinformatics*, vol. 21, no. 21, pp. 4021–4025, 2005.

[22] R. Bellman, *Dynamic Programming*. Princeton, NJ: Princeton Univ. Press, 1957.

[23] D.P. Bertsekas, *Dynamic Programming and Optimal Control,* 2nd ed. Belmont, MA: Athena Scientific, 2001.

[24] M. Bittner, P. Meltzer, Y. Chen, Y. Jiang, E. Seftor, M. Hendrix, M. Radmacher, R. Simon, Z. Yakhini, A. Ben-Dor, E. Dougherty, E. Wang, F. Marincola, C. Gooden, J. Lueders, A. Glatfelter, P. Pollock, E. Gillanders, D. Leja, K. Dietrich, C.M. Berens, D. Alberts, V. Sondak, N. Hayward, and J. Trent, "Molecular classification of cutaneous malignant melanoma by gene expression profiling," *Nature*, vol. 406, no. 6795, pp. 536–540, 2000.

[25] A.T. Weeraratna, Y. Jiang, G. Hostetter, K. Rosenblatt, P. Duray, M. Bittner, J.M. Trent, "Wnt5a signalling directly affects cell motility and invasion of metastatic melanoma," *Cancer Cell*, vol. 1, no. 3, pp. 279–288, 2002.

[26] R.D. Smallwood and E.J. Sondik, "Optimal control of partially observable markov processes over a finite horizon," *Oper. Res.*, vol. 21, no. 2, pp. 1071–1088, 1973.

[27] A. Choudhary, A. Datta, M. Bittner, and E. Dougherty, "Assignment of terminal penalties in controlling genetic regulatory networks," in *Proc. Amer. Control Conf.*, 2005, no. 2, pp. 417–422.

[28] R. Kalman, "Canonical structure of linear dynamical systems," *Nat. Acad. Sci.*, pp. 596–600, 1962.

SP