**Probabilistic representation of gene regulatory networks**

Linyong Mao and Haluk Resat*

Computational Biosciences Group, Pacific Northwest National Laboratory, PO Box 999, Mail Stop K1-92, Richland, WA 99352, USA

Received on October 28, 2003; revised on March 5, 2004; accepted on March 20, 2004

Advance Access publication April 8, 2004

**ABSTRACT**

**Motivation:** Recent experiments have established unambiguously that biological systems can have significant cell-to-cell variations in gene expression levels even in isogenic populations. Computational approaches to studying gene expression in cellular systems should capture such biological variations for a more realistic representation.

**Results:** In this paper, we present a new fully probabilistic approach to the modeling of gene regulatory networks that allows for fluctuations in the gene expression levels. The new algorithm uses a very simple representation for the genes, and accounts for the repression or induction of the genes and for the biological variations among isogenic populations simultaneously. Because of its simplicity, introduced algorithm is a very promising approach to model large-scale gene regulatory networks. We have tested the new algorithm on the synthetic gene network library bioengineered recently. The good agreement between the computed and the experimental results for this library of networks, and additional tests, demonstrate that the new algorithm is robust and very successful in explaining the experimental data.

**Availability:** The simulation software is available upon request.

**Contact:** haluk.resat@pnl.gov

**Supplementary information:** Supplementary material will be made available on the OUP server.

**INTRODUCTION**

Networks of interactions between cellular constituents control the biochemical processes of an organism and the resulting regulatory information is used by the cells to control the expression patterns of their genes. Therefore, knowing how gene expression patterns are regulated is very important in deciphering the ways organisms function and respond to environmental signals. With the completion of an increasing number of genomes, and with the rapid advancement in proteomics, microarray and other high-throughput techniques, it is now becoming possible to construct reasonably complete network diagrams that show the regulatory relationships among the genes of an organism. Such network graphs, which tabulate and sometimes quantify the regulatory interactions between the genes of organisms, are known as the gene regulatory networks (Bolouri and Davidson, 2002; De Jong, 2002). Although gene regulatory networks have been a subject of investigation for decades, recent recognition that biological networks are analogous to the circuits and systems encountered in engineering applications has led to the use of systems engineering approaches to describe these networks (Arkin et al., 1998; Hasty et al., 2001a, 2002b; McAdams and Shapiro, 1995). It has also been realized that the large quantitites of data resulting from high-throughput measurements, and their complex nature, make mathematical models and computational simulations essential for integrating and analyzing the experimental data, for deriving the underlying biological information, and in helping to design new experiments in a rational manner (Arkin, 2001; Bolouri and Davidson, 2002; De Jong, 2002; Endy and Brent, 2001; Hasty, 2002; Hasty et al., 2002b, 2001b; Rao and Arkin, 2001).

The recent fluorescence imaging experiments where carefully designed alleles of green florescent protein (GFP) were incorporated into the chromosome of *Escherichia coli* have established that the range of expression of a single gene in the same intracellular environment can be large (Elowitz et al., 2002). These experiments, and various modeling studies, demonstrated quite clearly that gene expression in living cells is highly stochastic in nature, in particular when the amount of transcription is low (Becskei and Serrano, 2000; Berg et al., 2000; Blake et al., 2003; Endy and Brent, 2001; Isaacs et al., 2003; Kepler and Elston, 2001; Ko, 1992; McAdams and Arkin, 1997, 1999; Ozbudak et al., 2002; Paulsson and Ehrenberg, 2001; Swain et al., 2002; Thattai and van Oudenaarden, 2001). In the past, a variety of mathematical models have been developed to simulate the qualitative and quantitative behavior of genetic regulatory networks (De Jong, 2002). However, most of these models are based on deterministic methods, such as logical operations and/or a set of (non-)linear differential equations, and so, do not account for the stochastic nature of gene expression. Various aspects

*To whom correspondence should be addressed.*
of how variations in kinetic rates, stochasticity or noise might affect the gene expression levels have been investigated by including terms with random factors to the deterministic models. Although they can be instrumental in investigating the role and the importance of random fluctuations, such models are hybrid at best and are not truly stochastic in character. In addition, detailed kinetic models are often ‘parameter rich’, i.e. even a small network comprising several genes requires the determination of a large set of kinetic rate constants. Unfortunately, most of the rates needed in such models are not easily measurable under in vivo conditions. Thus, the usefulness of fully kinetic or hybrid models is severely limited even for relatively modest sized gene regulatory networks. Noting such shortcomings, in this paper, we present a new approach in which changes in the expression levels of the genes in a genetic regulatory network are described by a purely probabilistic model. In our model, expression levels of genes are represented as multi-state functions. The gene expression levels go up or down with certain probabilities, thus allowing for the fluctuations in the gene expression levels. In our approach, properties of genes are defined with a very small number of lumped empirical parameters, making it suitable to study large-sized networks. It should be noted that the usage of our model is limited to (pseudo-) steady-state studies. Exploring the dynamical features of gene regulatory networks such as transient activation cannot be temporally resolved with the current form of our model.

We have tested our new approach by simulating the synthetic genetic regulatory networks recently engineered by Guet et al. (2002). The results of our simulations for the set of 22 synthetic networks constructed by forming various combinations of four well-characterized genes are in good agreement with the experimental results.

METHODS

In its simplest description, a genetic regulatory network is a graph consisting of a set of nodes and edges. A node in the network graph represents a gene and its promoter region (Fig. 1). The expression level of a gene (node) is controlled by the interaction of the regulatory transcription factors with its promoter, and edges (links) of the network graph represent such interactions between any pair of genes. For example, a link from genes \( x \) to \( y \) can show that, depending on the type of the link, the product of gene \( x \) inhibits or activates gene \( y \). In this case, the link shows that the protein product of gene \( x \) is a transcription factor that binds to the operator sites in the promoter region of gene \( y \) and regulates its expression levels. In the earlier gene regulatory network studies, gene expression levels were assumed to be either on or off to simplify the computations (Glass and Kauffman, 1973). This approach results in Boolean models, which cannot accommodate the partial expressions of genes that are often present in biological systems. Therefore, recent studies have concentrated on multi-state models in which expression levels of genes can have intermediate values between the full and zero expression limits (de Jong et al., 2003; Heidtke and Schulze-Kremer, 1998).

The methods used in previous gene regulatory network modeling studies can be grouped into three categories according to their formalism: (i) approaches based on the description of the regulatory effects using sets of logical rules (e.g. Heidtke and Schulze-Kremer, 1998; Thieffry and Thomas, 1998); (ii) approaches based on a set of deterministic (including piecewise linear) differential equations characterizing the regulation patterns of the network (e.g. Becskei and Serrano, 2000; Chen et al., 1999; de Jong et al., 2003; Goryanin et al., 1999; Hasty et al., 2000, 2002a; Savageau, 1998; Vohradsky, 2001; Vu and Vohradsky, 2002; Wessels et al., 2001); and (iii) fully kinetic models that also include mechanistic details, such as the protein production rates, the affinity between a regulator and its binding site, cell cycle time to calculate the protein dilution and degradation rates, and the DNA to RNA to protein cycle (Arkin et al., 1998; Kastner et al., 2002; McAdams and Arkin, 1997; McAdams and Shapiro, 1995; Rosenfeld et al., 2002; Shea and Ackers, 1985; Swain et al., 2002; Wong et al., 1997). Combination of these approaches, as well as inclusion of certain aspects of the stochasticity, have also been utilized (Davidson et al., 2002; McAdams and Shapiro, 1995). We refer the readers to the recent reviews by Bolouri and Davidson (2002) and by De Jong (2002), and the references therein, for further details of the model formalisms and for an overview of the field. Although the principles underlying the above categories are quite different, they have been mostly used in a deterministic manner in the past, i.e. in a way that the equations characterizing the model fully determine the dynamics of the system. It should be noted that, the system of equations characterizing the kinetic models can be integrated using a stochastic algorithm. This would allow for including some of the factors that give rise to the fluctuations...
in the gene expression levels. However, stochastic integration can only include the variations due to possible changes in the sequence of reactions or the variations due to the discreteness of the system. Alternatively, addition of stochastic random terms to the system equations has been employed to investigate the sources and consequences of randomness in biological systems (Blake et al., 2003; Isaacs et al., 2003; Kepler and Elston, 2001; Ozbudak et al., 2002; Paulsson and Ehrenberg, 2001; Thattai and van Oudenaarden, 2001).

It has been observed that regulatory systems can be ‘leaky’, i.e. it may not be possible to fully repress or induce gene expressions, and that uncontrollable biological effects can often change the strengths of regulatory interaction even in isogenic populations (Elowitz et al., 2002; Swain et al., 2002). Fully kinetic models based on deterministic models cannot account for such leak effects, even if they are integrated by using stochastic methods. Similarly, stochasticity added to the models using random noise terms only allows for the characterization of a particular random effect. In addition, fully kinetic models require the derivation and use of a large set of kinetic parameters, which are available for only a very limited number of small gene regulatory networks. Therefore, employment of fully kinetic models to study relatively large-sized networks is currently unfeasible. Based on a understanding of the capabilities of existing models, we developed a new approach to model gene regulatory networks with the long-term aim of being able to study large size gene regulatory networks. In our algorithm, changes in the expression levels of the genes in a network are represented using a probabilistic model. The gene expression levels go up or down with certain probabilities. Transition probabilities, that govern how the system may evolve, depend on the overall state of the network. Computed transition probabilities are used in a stochastic procedure to determine the equilibrium distribution of the expression levels. The new algorithm was implemented using a Monte Carlo approach where a multitude of Markov chains are created using the computed transition probabilities. Expression levels of the genes in a regulatory network are represented by using multi-state functions. Gene expression levels can vary between the allowed minimum and maximal values, which can be set differently for different genes. Expression levels of the genes can increase or decrease by one, or stay unchanged between successive steps of the Markov chain according to the employed transition probabilities. Transition probabilities were computed using the following rules: for each link in the network representing a regulatory effect of the product of gene \( i \) on gene \( k \), we define a weight \( W_{ik} \) that indicates the strength of the regulation of gene \( k \) by gene \( i \) (Mjolsness et al., 1991; Vohradskey, 2001; Weaver et al., 1999; Wessels et al., 2001). Total regulating strength for gene \( k \), \( S_k(t) \), at time (or Markov chain step) \( t \) is calculated as \( S_k(t) = \sum_i W_{ik} * G_i(t) \), where \( G_i(t) \) is the expression level of gene \( i \) at time \( t \). The \( W_{ik} \) parameters can be negative or positive indicating inhibition or activation, respectively, and the following values were used: \( W(\lambda - cI - P_{\lambda i}^+) = -1.0 \); \( W(\lambda - cI - P_{\lambda i}^-) = +1.0 \); \( W(\operatorname{LacI} - P_{\lambda i}^+) = -4.0 \); \( W(\operatorname{LacI} - P_{\lambda i}^-) = -1.0 \); and \( W(\operatorname{TetR} - P_{\lambda i}^+) = -4.0 \). It is assumed that if the expression of a gene is regulated by more than one regulator, the regulatory effects are additive. We note that this assumption can be altered without changing the basic philosophy of our algorithm. Transition probabilities, which determine how the expression value of gene \( k \) may change at the next step of the Markov chain, are calculated differently based on the sign of \( S_k(t) \). If \( S_k(t) \) is negative, transition probabilities are given as

\[
P_{k,j}(\uparrow) = P_0 \\
P_{k,j}(\downarrow) = P_0 * \left[ 1 + \frac{|S_k(t)|^{n_k}}{|S_k(t)|^{n_k} + C_{nk}^k} \right].
\]

(1)

In the above equation, \( P_{k,j}(\uparrow) \) and \( P_{k,j}(\downarrow) \) are the probabilities that the expression level of gene \( k \) will be lower or higher by one unit in the next step \((t + 1)\) of the Markov chain respectively, and \( P_{k,j}(\downarrow) \) is the probability that it stays unchanged. \( P_0 \) is the basal transition probability, which combines with various factors that may give rise to cell-to-cell variations among the members of a cell culture. Examples of such factors would be cell constituency differences, regulation leaks or fuzziness, differences in the transcript copy numbers and differences due to the point mutations. \( P_0 \) was set equal to 0.2 in our simulations. Test cases showed that, although it has a slight effect on the variance of the expression level distributions, changing the \( P_0 \) value does not affect the average mean expression levels significantly. The symbol \([\ldots]\) denotes the absolute value of the enclosed quantity, i.e. \( S(t) \) is always included as a positive number in Equation (1).

Parameters \( C_k \) and \( n_k \) are constants of the sigmoid function associated with gene \( k \), and the following values were used in our simulations: \( (C_k, n_k) = (150, 3.0) \) for \( \lambda - cI \), \( (100, 2.0) \) for \( \lambda \), \( (200, 1.5) \) for \( \lambda \), and \( (100, 2.0) \) for \( \lambda \). In the above equation, \( C_k \) parameter defines the expression value at which the regulatory interaction switches from weak to strong regulation and the \( n_k \) parameter represents the possible co-operativity between the operators. A sigmoid function was chosen because it is a simple function which, for large Hill coefficients \( n_k \), can reflect non-linear responses. We also tried a linear function to calculate the transition probabilities but the sigmoid function gave better overall agreement with the experimental results. As will be discussed in the Sensitivity analysis section, the main purpose of this study was to show that our simple algorithm can be applied successfully to study gene regulatory networks to detect the trends in regulatory patterns. Therefore, in this proof-of-principle study, we did not resort to using optimization methods to derive the parameters used in the model. Gene parameters \( C_k \) and \( n_k \), and the
weight factors $W_i$ representing the strength of the regulatory interactions were derived by trial and error. Since $S_i(t)$ is negative in Equation (1), it indicates that gene $k$ is repressed, and therefore, a decrease in its expression level would be more likely. This is reflected in Equations (1) that $P_{k,i}(\downarrow)$ is larger than $P_{k,i}(\uparrow)$. When $S_i(t)$ is positive, indicating an induction, transition probabilities were calculated as

$$P_{k,i}(\uparrow) = P_0 \times \left[ 1 + \frac{|S_{ki}|n}{|S_{ki}|n + C_k} \right]$$

$$P_{k,i}(\downarrow) = P_0$$

$$P_{k,i}(\downarrow) = 1 - P_{k,i}(\uparrow) - P_{k,i}(\downarrow).$$

When $S_i(t)$ is zero, up and down probabilities are both equal to $P_0$ and either Equation (1) or (2) can be used.

For each gene $k$, the transition probability table is prepared and a random number is picked from a uniform distribution in the unit interval at each simulation step. By comparing the random number with the transition probabilities, the expression level of gene $k$ in the next step is decided to be higher, lower or stay unchanged. For example, if the picked random number is between zero and $P_{k,i}(\downarrow)$, the expression level of gene $k$ goes down by one unit; if the random number is between $P_{k,i}(\downarrow)$ and $P_{k,i}(\downarrow) + P_{k,i}(\uparrow)$, the expression goes up by one; and the expression stays unchanged otherwise. At each Monte Carlo step, expression levels of every gene in the network were allowed to change simultaneously. After updating the network’s configuration according to the accepted transitions, the probability tables are recomputed to be used in the next simulation step. It should be noted that our method is analogous to the approach pursued by Kepler and Elston (2001) to derive the master equation for the Markov processes. The difference between our method and this earlier approach is that we do not use a reaction kinetic model to convert probabilistic equations into a set of differential equations for the probabilities. This allows us to keep the generality of the probabilistic model. It also makes it possible to avoid the introduction of rates into the algorithm, and thus, allows the use of a minimal number of required parameters. Avoiding the use of rates explicitly also allows us to eliminate possible multiple timescale problems that often arise in kinetic models (Resat et al., 2001).

In the numerical implementation of the algorithm, expression level of a gene is not allowed to go outside the extreme limits defined by the number of its multi-states. For example, if the transition selection for a gene is to decrease further its expression level, it is assumed that its expression level stays at the minimum and does not decrease when the expression of the gene is at its minimum possible level. A similar restriction applies to the situation when a gene reaches its maximal value. Expression levels of the genes were limited to be between 0 and 200 and were allowed to change simultaneously at each time step. Note that when a gene is not regulated (i.e. when $S_k$ is zero), the outcome of the simulations for the mean expression level of the gene will be half the maximum allowed level. Thus, the expression level of an unregulated gene will be 100 in our simulations. With this choice, we treat the computed average expressions as the percentage expression values of the genes. In other words, our model is constructed in such a way that unregulated genes are expressed fully. This is a particular choice and it can be modified as necessary without altering the basic concepts of our approach. All simulation runs were started from a random system configuration and, after a brief equilibration period, trajectories have been continued to run for 60 million Monte Carlo steps during which information about the gene expression levels and their fluctuations is collected for later analysis. By using concepts from statistical mechanics (Leach, 1996), expression values computed in our simulations can be considered to be either the mean expression value of a certain gene across many cells in an isogenic population or the time averaged mean expression value of a certain gene in a single cell. In these cases, the observed fluctuations about the mean correspond to expression value variations among isogenic cell populations or to the dynamical fluctuations in a particular cell.

In deterministic models based on rate equations, steady state is defined as the state where the net flow of material vanishes. In probabilistic models like ours, an analogous definition would be the state in which the transition probabilities for the expression level of a gene to go up or down are equal. This definition, of course, would be valid only when the system has no absorbing Markov states. However, a closer look at Equations (1) and (2) show that the up and down transition probabilities become equal only when $S(t)$ is zero. Although this seems to be a reasonable conclusion, it is misleading. As shown in the Appendix, because of the bounds on the gene expression values, it is possible to use Equations (1) and (2) and still achieve a steady state.

It should be noted that our algorithm is a new approach to simulations of gene regulatory networks. We have not yet optimized the formulas and transition rules used to compute the transition probabilities. Equations (1) and (2) can be adapted or changed as necessary without altering the underlying philosophy of our algorithm.

The simulation program was designed to be object-oriented and implemented in C++. Each gene in a gene regulatory network is represented by an object of the Gene class. The transition probability is captured as a member function of the Gene class. The program requires a user to prepare text files describing the network configuration and the gene parameters. The program outputs genes expression values as a function of simulation run length. The program has been tested under IRIX 6.5 and Linux 2.4.17. For a SiliconGraphics O2 machine with a 250 MHz CPU and 256 MB memory R10K processor, it takes about 10 min to run a 60 million step simulation for a four-gene network.
RESULTS

Simulation of genetic regulatory networks

In a recently published study, by forming various regulatory combinations of the \( \text{lacI}, \lambda-cI, \text{tetR} \) and \( \text{gfp} \) genes, Guet et al. (2002) designed and synthesized a library of 22 synthetic networks. By employing genetic engineering techniques, Guet et al. were able to arrange the four genes in the particular order: –promoter–\( \text{lacI} \)–promoter–\( \lambda \)–\( \lambda \)-\( cI \)–promoter–\( \text{tetR} \)–promoter–\( \text{gfp} \)– (Fig. 1). The first three genes code for the well-studied transcriptional factors, and the last gene is the reporter gene to monitor the regulatory activity in the cells. Each promoter in the network library was chosen from a set of five promoters which were chosen in such a way that they are regulated by the transcription factors coded by the used genes. This allowed Guet et al. (2002) to form small isolated networks whose logical behavior can be controlled with chemical inducers. The five promoters were \( P\lambda^- \) and \( P\lambda^+ \) (repressed and activated, respectively, by \( \lambda-cI \)), \( P_{\text{L}1} \) and \( P_{\text{L}2} \) (repressible by LacI) and \( P_{\text{T}} \) (repressible by TetR). The promoter for the \( \text{gfp} \) gene was chosen to be \( P\lambda^- \) in all cases, and the promoters for the other three genes were varied. The circuits forming the library of genetic regulatory networks are listed in Table 1.

For example, in network 7d/7e (Fig. 1b), through its interaction with the \( P_{\text{T}} \) promoter, the product of \( \text{tetR} \) gene suppresses its own expression as well as the \( \text{lacI} \) expression. Similarly, LacI inhibits \( \lambda-cI \) expression through the \( P_{\text{L}2} \) promoter, and \( \lambda-cI \) represses \( \text{gfp} \) expression through interacting with the \( P\lambda^- \) promoter.

Using our new probabilistic approach (detailed in the Methods section), six simulations were conducted for each of the 22 networks. Each simulation lasted 60 million Monte Carlo steps and the reported results are the average values of the six simulation runs. Equilibration analysis showed that the mean values converge well within 15 million steps and that statistics computed using \( 6 \times 60 \) million step runs should be quite precise (Supplementary Figure 1). Table 2 tabulates the computed mean expression levels of the genes in all networks and compares them with the experimental results. The results tabulated in Table 2 and the correlation plot in Figure 2 show that, with the exception of a few networks, there is a very good agreement between simulation and experimental results.

We have calculated the linear correlation between the simulation and the experimental results for the \( \text{GFP} \) expression and found a correlation coefficient of \( R^2 = 0.91 \) and a slope of 0.82 (Fig. 2). Five of the networks (2b/2c, 5a/5b, 9b/9c, 5c and 6a) were excluded from the correlation calculations. Excluded networks either had mutations observed in the experiments or had contradictory results. Although networks 5a and 5b were designed to have the same network connectivity, it was observed that there were mutations in the upstream regions of the genes (Guet et al., 2002). Since they would alter the transcription factor binding affinity, mutations in the promoter region can affect the gene expressions severely (see note 11 in Guet et al., 2002). The \( \text{GFP} \) expression levels were measured to be 20 and 80 for networks 5a and 5b, respectively. Clearly, it is not appropriate to use the average of these two values to compare with our simulation results. An obvious choice would have been to pick up the experimental result which gave a better agreement but, we erred on the conservative side and excluded the results for the networks 5a/5b from the correlation analysis. Networks 2b/2c and 9b/9c were also excluded.

<table>
<thead>
<tr>
<th>Table 1. Set of studied gene regulatory networks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network</td>
</tr>
<tr>
<td>---------</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2a</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5a/5b</td>
</tr>
<tr>
<td>6a</td>
</tr>
</tbody>
</table>

In the diagrams, the gene names are shortened to L = \( \text{lacI} \), \( \lambda = \lambda-cI \), T = \( \text{tetR} \), and G = \( \text{gfp} \). Activation is denoted by pointed arrows (↑) and repression is denoted by blunt arrows (⊥). A description of the connectivity diagram of network 7d/7e can be found in the text.
The contradiction exists for network 6a. Compared with networks for the two cases contradict each other (Table 2). A similar relation results confirm this expectation but experimental results should be roughly the same in networks 5c and 10b. Our simulation should not affect the expression levels of the other genes from the rest of the network (Table 1), the TetR expression in network 6a is activated by λ-cr. Therefore, one would expect that the TetR level would be higher in network 6a than in networks 7a/7b leading to an increased suppression of λ-cr, which in turn would decrease the repression of gfp by λ-cr. Thus, the GFP level in network 6a is expected to be larger than the GFP level in networks 7a/7b. To avoid including contradictory results, we have opted to exclude these cases from the correlation analysis as well.

We have also investigated whether our simple algorithm can capture the subtle effects of the regulatory networks under study. As the first example, we compared the results of networks 8 and 10a. In network 8, gfp and lacI have the same promoters and both are suppressed by λ-cr. In the formulas used to determine the transition probabilities (see Methods section), gfp and lacI have the same interaction parameters and, because of their symmetric configuration, the mean expression levels for GFP and LacI are expected to be the same in network 8. The statistical distribution of GFP and LacI expression levels shown in Figure 3 confirm that this is indeed the case. In contrast, the toggle switch (Hasty et al., 2002b) in network 10a couples the expression level changes of the genes λ-cr and lacI. In terms of its function, mutual inhibition of two genes is equivalent to a positive feedback loop (Hasty et al., 2002b). As LacI expression level goes up, increased repression of λ-cr causes λ-cr expression to decrease, which in turn lowers the repression of lacI further. The positive feedback in the network 10a introduces an asymmetry between the regulation of the lacI and gfp genes by λ-cr, which leads to differences in their expression levels. Our algorithm nicely captures this subtle effect, and our results show that there is a difference between the expression levels of these two genes (81 for LacI versus 70 for GFP; Table 2). Since the positive feedback loop decreases stability, expression of LacI in network 10a can be expected to have a wider spread than that of GFP, and our results confirm this expectation as well (Fig. 3). The difference between the regulation of the lacI and gfp genes by λ-cr in network 10a is most evident when one compares the computed mean expression levels of LacI and GFP as a function of λ-cr expression (Fig. 3d). Results for the LacI and λ-cr expressions clearly show the correlation expected in a co-repressive toggle switch network, with its two stable states, where LacI and λ-cr expression levels are either high and low or low and high, respectively (Hasty et al., 2001b). In contrast, as expected, GFP expression decreases monotonically when the λ-cr level increases.

As the second example, we have compared the behavior of networks 2a and 10a. In network 2a, lacI is auto-regulated limiting its expression level to a narrow range below 20 and the positive feedback in network 10a allows for the expression of lacI at almost all levels (Supplementary Figure 2). This difference in the expressed LacI levels gets reflected in the expression profiles of λ-cr and GFP, where strong auto-repression of lacI in network 2a allows for the expression of λ-cr at high levels, which suppresses GFP expression (Table 2).

### Table 2. Mean gene expression values of the genes.

<table>
<thead>
<tr>
<th>Network ID</th>
<th>GFP (expt)</th>
<th>GFP (sim)</th>
<th>LacI (sim)</th>
<th>TetR (sim)</th>
<th>λ-cr (sim)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
<td>6</td>
<td>8</td>
<td>199</td>
</tr>
<tr>
<td>2a</td>
<td>2</td>
<td>13</td>
<td>6</td>
<td>8</td>
<td>57</td>
</tr>
<tr>
<td>2b/2c</td>
<td>60/29</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>99</td>
</tr>
<tr>
<td>3</td>
<td>13</td>
<td>17</td>
<td>16</td>
<td>40</td>
<td>30</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1</td>
<td>6</td>
<td>21</td>
<td>199</td>
</tr>
<tr>
<td>5a/5b</td>
<td>20/80</td>
<td>70</td>
<td>79</td>
<td>8</td>
<td>22</td>
</tr>
<tr>
<td>5c</td>
<td>99</td>
<td>12</td>
<td>15</td>
<td>8</td>
<td>90</td>
</tr>
<tr>
<td>6a</td>
<td>1</td>
<td>92</td>
<td>6</td>
<td>106</td>
<td>2</td>
</tr>
<tr>
<td>6b</td>
<td>90</td>
<td>80</td>
<td>128</td>
<td>8</td>
<td>5</td>
</tr>
<tr>
<td>7a/7b</td>
<td>28/36</td>
<td>39</td>
<td>6</td>
<td>21</td>
<td>15</td>
</tr>
<tr>
<td>7c</td>
<td>65</td>
<td>58</td>
<td>16</td>
<td>40</td>
<td>9</td>
</tr>
<tr>
<td>7d/7e</td>
<td>1/1</td>
<td>7</td>
<td>11</td>
<td>8</td>
<td>93</td>
</tr>
<tr>
<td>8</td>
<td>29</td>
<td>19</td>
<td>19</td>
<td>8</td>
<td>37</td>
</tr>
<tr>
<td>9a</td>
<td>0</td>
<td>8</td>
<td>2</td>
<td>97</td>
<td>87</td>
</tr>
<tr>
<td>9b/9c</td>
<td>12/98</td>
<td>85</td>
<td>2</td>
<td>97</td>
<td>3</td>
</tr>
<tr>
<td>10a</td>
<td>94</td>
<td>70</td>
<td>81</td>
<td>11</td>
<td>21</td>
</tr>
<tr>
<td>10b</td>
<td>0</td>
<td>13</td>
<td>16</td>
<td>72</td>
<td>91</td>
</tr>
<tr>
<td>11a/11b</td>
<td>73/95</td>
<td>94</td>
<td>104</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>12a/12b</td>
<td>23/28</td>
<td>24</td>
<td>21</td>
<td>10</td>
<td>23</td>
</tr>
<tr>
<td>12c</td>
<td>16</td>
<td>39</td>
<td>31</td>
<td>22</td>
<td>14</td>
</tr>
<tr>
<td>13a</td>
<td>0</td>
<td>6</td>
<td>1</td>
<td>192</td>
<td>100</td>
</tr>
<tr>
<td>13b/13c</td>
<td>1/0</td>
<td>7</td>
<td>193</td>
<td>2</td>
<td>96</td>
</tr>
</tbody>
</table>

*Connection diagrams of the networks are tabulated in Table 1.*

**Fig. 2.** Comparison of the simulation results with the experimental results for the mean GFP expression for 17 networks. The solid line (y = 0.82x + 7.31) is the result of linear regression analysis, with correlation coefficient $R^2 = 0.91$.
Fig. 3. Comparison of the distribution of the gene expression levels in networks 8 and 10a. (a) $\lambda$-cl, (b) LacI and (c) GFP. Mean expression values of the genes are tabulated in Table 2. Results are for network 8 (white) and network 10a (black). In (a–c), the $x$-axis has a logarithmic scale and the data point with the lowest $x$-value represents the count frequency for the intensities $\leq 1$. (d) Shows the expression values of LacI and GFP versus the expression value of $\lambda$-cl in network 10a. The expression value of LacI (or GFP) at $\lambda - \text{cl} = X$ was computed by taking the average of the LacI (or GFP) expressions when the expression level of $\lambda$-cl was equal to X. Results are for LacI (gray triangles) and GFP (black rectangles).

In the third case, networks 6b and 10b were compared. These two networks differ in that the loops between $lacI$ and $\lambda$-cl genes have different feedback characteristics (Table 1). In network 6b, the induction of $lacI$ by $\lambda$-cl drives the network to its extreme condition with very low $\lambda$-cl and high LacI levels, and leads to a high GFP expression (Supplementary Figure 3). In contrast, as discussed earlier, the positive feedback loop in network 10b drives the network towards one of its bi-stable points, where $\lambda$-cl is expressed at high levels causing strong suppression of gfp (Table 2).

As the final example, we compare networks 10a and 10b. These two networks have identical network topology (Table 1). The only difference between these two networks is that the genes regulated by LacI have different promoters. The difference however is profound enough that the GFP is either expressed at a very high (network 10a) or a very low (network 10b) level (Table 2). In these two networks, expression of the gfp gene is controlled indirectly by LacI through its regulation of $\lambda$-cl. With the difference in a single model parameter for the interaction of LacI with $P_{L1}^T$ and $P_{L2}^T$ (see Methods section), our method can account for such indirect regulations and give results that agree well with the experimental observations.

Fig. 4. Distribution of the GFP expression for network 7a as computed in the simulations. Count frequency distributions computed in the simulations correspond to the cell counts versus the intensity measurements in the flow cytometry measurements for the population of cells. In this respect, computed distribution can be compared with the FACS distribution reported by Guet et al. (2002, Figure 2). The $x$-axis has a logarithmic scale and the data point with the lowest $x$-value represents the count frequency for the intensities $\leq 1$.
shows that our algorithm accurately represents the inherent biological fluctuations.

**Sensitivity analysis**

Since the main purpose of this study was to show that our new algorithm can be successfully applied to study gene regulatory networks, we did not resort to using optimization methods to derive the parameters used in the model. Although multi-variate approaches can in principle be used to derive an optimal parameter set, the model parameters used in this study were determined by trial and error until reasonable agreement with experimental results were obtained. For this reason, we performed a sensitivity analysis by increasing or decreasing one of the 13 model parameters by 10% from its original value while keeping the remaining 12 parameters unchanged. This analysis, Figure 5, showed that the simulation results have only a very weak dependence on the parameter set and that our simple model for the regulatory networks is quite robust. An extensive tabulation of the sensitivity analysis results can be found in the Supplementary material.

Changes in the gene expression values when one of the model parameters is varied actually can give ideas about the stiffness of the expression ranges in a given network. For example, the GFP expression is high in networks 6b, 10a and 11a/11b (Table 2). However, varying the gene parameters for λ-cI by 10% changes the GFP expression in network 10a the most (Fig. 5). In networks 6b and 11a/11b (Table 1), the mutual feedback loop between λ-cI and LacI stabilizes the λ-cI expression at a very low level and this gets reflected as insensitivity to the gene parameters. In contrast, in network 10a, mutual inhibition of two genes in a toggle switch arrangement results in a positive feedback and destabilizes the system, which leads to a higher sensitivity to the model parameters.

**Response to the addition of a chemical inducer**

The binding affinity of transcription factors to promoter regions can be altered with the addition of suitable chemical inducer molecules (Aubrecht et al., 1996). For example, the inducer anhydrotetracycline (aTc) changes the DNA binding state of TetR (to the P^T promoter in this case). aTc forms a complex with TetR with a high affinity, and significantly reduces the binding of TetR protein to its operator site (Scholz et al., 2000). Thus, addition of aTc leads to a higher expression of genes repressed by TetR. To validate our model further, we have simulated the effect of the addition of chemical inducer aTc on the expression level of the reporter gfp gene. This parallels the approach followed by Guet et al. (2002) in their recent study, where significant changes in the GFP expression levels were observed in some of the networks when aTc was added to the system.

The parameter defining the strength of the interaction of TetR with the promoter P^T was set to −4.0 (Methods section).
The effect of the addition of the inducer aTc was represented by reducing the strength of the TetR–P\textsuperscript{T} interaction to $-0.04$, and one 60 million step simulation was run for each of the 17 networks. A closer look at the network topologies (Table 1) shows that in networks 1, 2a, 3, 4, 6b, 10a, 10b and 11a/11b, tetR has no effect on the expression level of gfp gene since it is either isolated from the subnet containing gfp or is an end point of networks not regulating the other genes. Thus, adjustment of the strength of the TetR–P\textsuperscript{T} interaction would not alter the GFP expression in these networks.

Figure 6 reports how mean GFP expression levels change when aTc is added to the system and compares the simulation and experimental results. As shown in the figure, the simulation results are in very good agreement with experimental observations except in one case, network 9a. In this case, experimental results showed that adding aTc did not affect the GFP expression. Network 9a’s connectivity diagram (Table 1) indicates that high affinity binding of aTc to TetR would remove the repression of the lacI. Increased levels of LacI should increase the repression of λ-cI and lead to an increase in the GFP expression level. Although there is disagreement with experiment, our simulation results are in line with this expectation. It should be noted that the effect of aTc was incorporated into our model by decreasing the strength of interaction of TetR with the promoter P\textsuperscript{T} with a factor of 100, which was chosen arbitrarily and it can be adjusted further to improve the agreement with the experimental results. However, it is remarkable that a single parameter adjustment in a simple model can explain the changes induced by the addition of aTc in a large library of networks. Albeit indirect, this finding points to the robustness of our model.

DISCUSSION

We presented a new approach to model gene regulatory networks. The new algorithm is based on a fully probabilistic representation which allows for the employment of stochastic simulation methods such as Monte Carlo techniques. An important aspect of the new algorithm is that it allows for the simultaneous representation of repression/activation and leakiness of the gene expression. Recent experiments have established unambiguously that biological systems can have significant cell-to-cell variations due to intrinsic or external noise (Elowitz et al., 2002). Since fluctuations are a direct natural consequence of the new probabilistic model, our approach allows for a more realistic representation of cellular systems in gene regulatory network simulations and makes it possible to study the effects of biological fluctuations and of leaks and fuzziness in gene expression. Another important aspect of the new method is that it uses a minimal number of parameters (a total of 13 parameters for a large library of four-gene networks) to represent the dynamics of the biological system. It will soon be possible to derive large-sized gene regulatory networks from high-throughput data. As they require a large number of parameters, the usefulness of fully kinetic or hybrid models is severely limited even for relatively modest sized gene regulatory networks. Therefore, development of simplistic approaches such as ours will be necessary to quantitatively investigate large-sized regulatory networks. Based on the good agreement with real biological data for the test system and the demonstrated robustness of the model, we can confidently state that our new method is a very promising initial attempt to fill this gap.

Guet et al. (2002) have recently synthesized a set of four-gene networks by forming combinatorial connections among the genes. Using their networks as a test case, we have shown that our simple probabilistic algorithm can explain the experimental data. The linear correlation coefficient between the computed and the experimental results was larger than 0.9 showing good agreement. Guet et al. have also investigated how the synthetic networks can be used as logical gates when their behavior is controlled by chemical inducers. We mimicked this set of experiments by modifying the strength of the interaction of the involved transcription factor with the promoter sequence that it recognizes. With a simple alteration of a single model parameter, we are able to account for the effect of the added inducer aTc and our results agree well with the experimental results.
We have also investigated the robustness of the new model by analyzing the sensitivity of the results to the model parameters. The sensitivity analysis showed that our model is robust enough that the model parameters do not have to be optimized fully to obtain good agreement between the computed and experimental results. As we have not pursued complete optimization of the model parameters, the good agreement between our computed and experimental results could be improved further by fine tuning the model parameters.

In addition to showing that the new probabilistic algorithm is very successful, our study also showed that it is possible to find simple models in which a minimal number of parameters are enough to characterize the gene regulatory networks. This simplicity supports the concept that the new systems biology approach to modeling biological systems as engineering networks is feasible. There are recent efforts to separate the observed biological fluctuations into intrinsic and extrinsic components (Elowitz et al., 2002; Swain et al., 2002). Because of the lack of experimental data to justify a more complicated model, we have opted to lump these contributions into one term. However, as biological data becomes available to support the separation of the factors, our method can be easily generalized by adjusting the transition probability expressions to include specific contributions to the stochasticity explicitly.

ACKNOWLEDGEMENTS

We would like to thank Calin C. Guet and Stanislas Leibler for making their data available to us and David Dixon for his helpful comments and careful reading of the manuscript. This work was supported by the Advanced Modeling and Simulation of Biological Systems Program of the Office of Advanced Scientific Computing Research, Office of Science, the U.S. Department of Energy, and by the Laboratory Directed Research and Development (LDRD) funds of Pacific Northwest National Laboratory, operated for the DOE by Battelle.

REFERENCES


APPENDIX

Determination of the steady state

When there are no absorbing Markov states, the steady state of our probabilistic model can be defined as the state in which the transition probabilities for the expression level of a gene to go up or down are equal. An initial look at Equations (1) and (2) could lead to the misleading conclusion that the up and down transition probabilities become equal only when \( S(t) = 0 \). We show in this Appendix that it is possible to use Equations (1) and (2) and still achieve a steady state.

Consider a network consisting of only two genes, where the first gene \( \lambda - cI \) suppresses the second gene \( gfp \). In this arrangement, with the transition rules given in Equation (1), the probability that GFP expression will decrease would always be greater than the probability of increasing. Then, how can \( gfp \) have a steady, finite mean expression value? The answer to this question lies in the fact that the expression levels of the genes are bound by extreme values. In our simulation, when the expression level is at its minimum, the expression is not allowed to decrease any further. Consider for simplicity that the expression level of \( \lambda - cI \) in the two-gene network is kept constant at \( G(\lambda - cI) \). A finite \( G(\lambda - cI) \) would give rise to a difference between the up and down transition probabilities for the \( gfp \) gene. Define the probability difference \( \Delta P \equiv P(\downarrow) - P(\uparrow) > 0 \) and assume that the fraction of times when the expression level of GFP is at its minimum is \( f_{\min} \). Similarly, \( f_{\max} \) is the fraction of the times when the expression level of GFP is at its maximum. Since GFP expression is not allowed to go outside its finite range, the balance between the up and down probabilities requires that

\[
\begin{align*}
\frac{1}{f_{\min}} & = (1 - f_{\min} - f_{\max}) \cdot \Delta P + f_{\max} \cdot P(\downarrow) \\
\frac{1}{f_{\max}} & = f_{\min} \cdot \Delta P + P(\uparrow)
\end{align*}
\]

The sum of the above equations is \( \frac{1}{f_{\min}} + \frac{1}{f_{\max}} = 1 \). This simple analysis assumes that \( f_{\min} + f_{\max} = 1 \), which is not during the simulation trajectory, but one can generalize the above to include time dependence and the basic concept remains valid.

In order to address this question further, and prove the validity of our explanation, we have performed additional simulations. To increase statistical accuracy, we reverted to multi-copy stochastic simulations. In these runs, the simulated system included 10 000 copies of the \( gfp \) gene, which was equivalent to running 10 000 single copy trajectories in parallel. In the simulated network, \( \lambda - cI \) inhibits \( gfp \) with an interaction weight of \(-1.0\). The expression level of \( \lambda - cI \) was held constant during the simulation run, \( G(\lambda - cI) = 30 \). GFP expression was limited to the \([0,200]\) range and its parameters were \( C_{\text{gfp}} = 100 \) and \( n_{\text{gfp}} = 2.0 \). According to the transition rules [Equation (1)], for each copy, the transition probabilities are \( P(\uparrow) = 0.2, P(\downarrow) = 0.2165 \) and \( P(\uparrow) = 0.5835 \) at every time step. With these transition probabilities, and considering that \( f_{\max} \sim 0 \), one expects to
obtain $f_{\text{min}} = 7.62\%$ as the fraction of time the GFP expression will be at its minimum. In the simulations, a different random number was generated at each time step to determine how the expression level of a certain copy of the gfp gene would change. The run length of the simulations was 300,000 steps, and statistics were collected with a frequency of every other 30 time steps. The simulations showed that when GFP reached a steady, finite mean expression value, on average 760 out of 10,000 copies of the genes had a zero (i.e. minimum) value, which corresponds to a $f_{\text{min}}$ of 7.6% and agrees well with the expectation based on the discussion given above.

This test case study also shows that one can easily generalize the new algorithm to include multiple copies of the genes, which may or may not be allowed to affect the expression levels of (i.e. interact with) each other. Although it was not pursued in this study, such models can be very useful in incorporating various biochemical effects, such as the co-operativity and concentration-dependent regulation, etc., that are often encountered in cellular systems.