

Systems biology

SGN Sim, a Stochastic Genetic Networks Simulator

Andre S. Ribeiro^{1,2,*} and Jason Lloyd-Price¹

¹Institute for Biocomplexity and Informatics, University of Calgary, Canada and ²Centre for Computational Physics, University of Coimbra, P-3004-516 Coimbra, Portugal

Received on November 5, 2006; revised on January 4, 2007; accepted on January 10, 2007

Advance Access publication January 31, 2007

Associate Editor: Chris Stoeckert

ABSTRACT

Summary: We present *SGNSim*, ‘Stochastic Gene Networks Simulator’, a tool to model gene regulatory networks (GRN) where transcription and translation are modeled as multiple time delayed events and its dynamics is driven by a stochastic simulation algorithm (SSA) able to deal with multiple time delayed events. The delays can be drawn from several distributions and the reaction rates from complex functions or from physical parameters. *SGNSim* can generate ensembles of GRNs, within a set of user-defined parameters, such as topology. It can also be used to model specific GRNs and systems of chemical reactions. Perturbations, e.g. gene deletion, over-expression, copy and mutation, can be modeled as well. As examples, we present a model of a toggle switch without cooperative binding subject to perturbations, a system of reactions within a compartmentalized environment where membrane crossing is controlled by a negative feedback mechanism and a simulation based on the yeast transcriptional network.

Availability: *SGNSim* program, instructions and examples available at <http://www.cs.tut.fi/~sanchesr/SGN/SGNSim.html>.

Contact: andre.sanchesribeiro@tut.fi

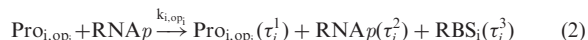
1 INTRODUCTION

One of the goals of systems biology is to understand how genes and proteins interact by developing computer models of biological phenomena, data or patterns, such as complex biochemical reactions or gene networks. Recent studies on gene expression (Raser and O’shea, 2003) and small synthetic genetic networks (Elowitz and Leibler, 2005; Gardner *et al.*, 2000) provided experimental data on the phenotypic variability and stochastic nature of gene expression. Corresponding theoretical investigations incorporating the stochastic nature of chemical reactions have given reasonable explanations for these observations. While non-delay reactions correctly model equilibrium states of simple synthetic gene systems, the dynamics of real gene regulatory networks (GRNs) are so complex [e.g. due to many feedback loops (Monk, 2003)] that time delays (Gibson, *et al.*, 2003), namely for transcription and translation (Gaffney and Monk, 2006), should be included (Bratsun *et al.*, 2005). In many cases the behavior of GRNs is characterized by transients, rather than just steady states. For example protein production delays were shown to significantly alter transients between attractors, even in simple GRNs (Ribeiro *et al.*, 2006).

Here, we use a generalized delay algorithm that handles multiple distinct delayed output events for each input event (Roussel and Zhu, 2006). We model GRNs using a set of reactions (Ribeiro *et al.*, 2006), including time-delayed transcription and translation. Genes interact via activation and repression reactions and the dynamics is driven by the Stochastic simulation algorithm (SSA) (Gillespie, 1976). We present *SGNSim* which models a wide range of systems of chemically interacting elements. The reaction rates may be constant or a complex function of any element’s concentration. Multiple time delays, generated from distributions, are also supported. It can incorporate diffusion and membrane transport as time delayed events, perturbations (e.g. gene deletion, knockout, over-expression, copy, mutation) and importantly it can generate random GRNs with some average features and apply the ensemble approach (Kauffman, 2004) to study large networks.

2 METHODS

SGNSim consists of a GRN or chemical system generator and a dynamics simulator. *SGNSim* is command line driven, receiving as input a .g file where reactions and elements initial quantities are specified. The output is a tab-delimited text file with a time series of all elements’ quantities (see Availability section). GRNs are generated from the following reactions: for gene $i = 1, \dots, N$, basal transcription of promoter Pro_i by RNA Polymerase (RNAP) (1), activated transcription where operators sites j of gene i , $Pro_{i,j}$, are occupied by one or more activators (2), translation of RNA by ribosomes (Rib) into proteins (p_i) (3), repression/unrepression of a gene at operator site j (4) and (5), decay of ribosome binding sites (RBS) and proteins (p_i ’s) (6), and proteins polymerization (here, limited to dimers for simplicity) (7), that can act as indirect repression or activation. Unless time delays, τ ’s, are explicitly represented in the products, all events, reactants depletion and products appearance, occur instantaneously at t :



*To whom correspondence should be addressed.

In Equations 1–3, τ 's superscripts distinguish the delays between products. Subscripts distinguish the delays of products of reactions associated to different genes. For example if reaction of Equation 1 occurs for gene i , at time t , Pro_i and one $RNAp$ are removed from the system and placed in a waiting list of events. At $t + \tau_1^1$, Pro_i and RBS and at $t + \tau_2^2$ $RNAp$ are released (unchanged), becoming available for reactions. We do not explicitly represent complete RNA molecules since RBS's appearance determines when translation can start. Transcription and translation could also be modeled in a single step (Roussel and Zhu, 2006). For simplicity, proteins and RBS degradation are assumed to occur at constant rate and modeled as unimolecular reactions. Since genes have multiple operator sites, a promoter 'state' is defined by the combined states of its operator sites, i.e. if they have their respective transcription factor (TFs) bound to it or not. We adopted the following notation: an array is used as the index of a promoter, e.g. $Pro_{i,op(i)}$ where $op(i) = (0, p_w, 0)$. In this case, only p_w is bound to operator site $j = 2$ of gene i , while the other operator sites are free.

SGNSim creates a GRN by generating a graph with the desired topology, imposing in-degree and out-degree distributions. Inputs are monomers or combined into multimers and set as direct or indirect. After, each direct input is assigned to an operator site (different TFs can be allowed, or not, to compete for the same operator site), while indirect inputs are given a target. Lastly, a function is assigned to each gene, defining the gene's response to a combination of TFs (promoter state). The functions can be unate, or a transcription rate can be randomly assigned to each combination of promoter states, since the combined effect of TFs on transcription rates is not necessarily unate.

The information defining the set of reactions that the resulting GRN consists of is stored in an adjacency matrix. A site a_{ij} of the adjacency matrix contains the set of values (x_1, x_2, x_3, x_4) . x_1 defines the type of interaction from j to i : 0 (no interaction), 1 (monomer) or 2 (dimer). In general, this value can grow to include higher-order multimers. x_2 defines if it is a direct interaction (+1), i.e. if p_j binds to Pro_i , or indirect (−1), as in (5). x_3 is not null in the case of a dimer interaction and equal to the number identifying the protein that binds to p_j and forms the heterodimer. If higher-order multimers are allowed, this field becomes a list of all participating monomers. Finally, x_4 identifies which operator site j of Pro_i the complex binds to, and is 0 in the case of an indirect interaction.

Given the formalism described earlier, *SGNSim* can generate ensembles of random networks within user-defined parameters, allowing exploration of a large phase space. The user can, for example, select the topology from several provided topologies, specify the distributions from which the initial concentrations of chemical species and the reaction rates are generated, set the ratio of activation/inhibition interactions and the fraction of housekeeping genes (those with basal level of expression), etc.

Because genes have different lengths, resulting in RNA and proteins with different sizes, their transcription and translation delays differ. Also, $RNAp$'s do not transcribe at a constant speed due to molecular noise, so we allow delays to vary from the mean delay at runtime. The resulting network can be simulated for a specified amount of time and sampled at set regular intervals.

SGNSim can also simulate specific systems of reactions such as chemical pathways or small GRNs. In addition, these systems can be modeled as if in compartmentalized spaces. Here, two processes are important: membrane crossing and diffusion. The crossing of a molecule A from compartment 1, of volume V_1 , to compartment 2, can be modeled by a single step reaction ($A_1 \rightarrow A_2$) with a propensity given by (8):

$$C^{\text{cross(3dim)}}(A, 1 \rightarrow 2) = \frac{2 \cdot A_m \cdot v_{A/m}}{\pi^2 \cdot V_1} \cdot P_{\text{cross}}(m_{1 \rightarrow 2}) \quad (8)$$

Here, A_m is the membrane surface area, $v_{A/m}$ is the molecule's average velocity perpendicular to the membrane and $P_{\text{cross}}(m_{1 \rightarrow 2})$ is the

probability of molecule A crossing when colliding with the membrane. When relevant to the system's dynamics, diffusion can be modeled by a time delayed version of the membrane crossing reaction, where the time delay should follow a normal distribution. In some cases, as the one just described, it might be useful to set the physical parameters from which the propensity can be computed. For that purpose, *SGNSim* also provides an interface that calculates rate constants from physical parameters if these are provided.

Whether using the ensemble approach or a user defined system of reactions, *SGN Sim* can test the system in many ways. For instance, in GRNs, it can do 'mock' perturbation experiments like gene deletion, over-expression, copy and mutation. For example to model a gene over-expression t seconds after the experiment started, a specific inducer is introduced at that time in the simulation.

Importantly, reaction rates can be some complex functions, such as hill functions, that depend on concentrations at each moment. This allows, for example, modeling active transport via a membrane.

For models of gene expression, *SGNSim* can include alternative splicing where a pre-mRNA transcribed from a gene can lead to different mature mRNA molecules and, thus, different proteins. This can be achieved by introducing an extra reaction: $PreRBS_i \xrightarrow{k_{\text{splice}}} RBS_{i,j}(\tau_i^{\text{splice}})$.

SGNSim dynamics are simulated using the 'delayed SSA' (Roussel and Zhu, 2006), that, unlike the non-delayed SSA, uses a waiting list to store the delayed output events, and proceeds as follows:

- (1) Set $t \leftarrow 0$, $t_{\text{stop}} \leftarrow$ stop time, read initial number of molecules and reactions, create empty waiting list L .
- (2) Do an SSA step for input events to get next reacting event R_1 and corresponding occurrence time t_1 .
- (3) If $t_1 + t < t_{\text{min}}$ (the least time in L), set $t \leftarrow t + t_1$. Update number of molecules by performing R_1 , adding delayed products into L as necessary.
- (4) If $t_1 + t \geq t_{\text{min}}$, set $t \leftarrow t_{\text{min}}$. Update number of molecules by releasing the first element in L .
- (5) If $t < t_{\text{stop}}$, go to step 2.

3 DISCUSSION

SGNSim can model a wide range of chemical systems since it allows compartmentalization, complex rate constants, multiple time delays and perturbations, among other features. Importantly, using an improved version of the GRN model proposed in (Ribeiro *et al.*, 2006), it can generate ensembles of large GRNs, within a large range of user-defined parameters. As examples of applications of *SGNSim* we modeled a toggle switch without cooperative binding subject to perturbations after a transient, a compartmentalized environment with active membrane transport and a model of the yeast transcriptional network (see Availability section).

ACKNOWLEDGEMENTS

The authors thank iCORE, a funding agency of Alberta and the IBI Informatics of the U. of Calgary.

Conflict of Interest. none declared.

REFERENCES

- Bratsun,D. *et al.* (2005) Delay-induced stochastic oscillations in gene regulation. *PNAS*, **102**, 14593.
- Elowitz,D.C. and Leibler,M.B. (2005) A synthetic oscillatory network of transcriptional regulators. *Nature*, **403**, 335–338.
- Gaffney,E.A. and Monk,N. (2006) Gene expression time delays and turing pattern formation systems. *Bull. Math. Biol.*, **68**, 99–130.
- Gardner,T.S. *et al.* (2000) Construction of a genetic toggle switch in *Escherichia coli*. *Nature*, **403**, 339–342.
- Gibson,M.A. and Bruck,J. (2000) Efficient exact stochastic simulation of chemical systems with many species and many channels. *J. Phys. Chem. A*, **104**, 1876–1889.
- Gillespie,D.T. (1976) A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *J. Comput. Phys.*, **22**, 403–434.
- Kauffman,S.A. (2004) The ensemble approach to understand genetic regulatory networks. *Physica A*, **340**, 733–740.
- Matsumoto,M. and Nishimura,T. (1998) Mersenne Twister. *SACM Trans. on Modeling and Comp. Simulation*, **8**, 3–30.
- Monk,N.A.M. (2003) Oscillatory Expression of Hes1, p53, and NF κ B driven by transcriptional time delays. *Current Biology*, **13**, 1409–1413.
- Raser,J.M. and O’Shea,E.K. (2005) Noise in gene expression: origins, consequences, and control. *Science*, **309**, 2010–2013.
- Ribeiro,A.S. *et al.* (2006) General, modeling strategy for gene regulatory networks with stochastic dynamics. *J. Comp. Bio.*, **13**, 1630–1639.
- Roussel,M.R. and Zhu,R. (2006) Validation of an algorithm for delay stochastic simulation of transcription and translation in prokaryotic gene expression, (submitted).