Evolving Noisy Oscillatory Dynamics in Genetic Regulatory Networks

André Leier[1], P. Dwight Kuo[2], Wolfgang Banzhaf[3], and Kevin Burrage[1]

[1] Advanced Computational Modelling Centre,
University of Queensland, Brisbane, QLD 4072, Australia
{leier, kb}@acmc.uq.edu.au
[2] Department of Bioengineering, University of California,
San Diego, La Jolla, CA 92093-0412, USA
pdkuo@ucsd.edu
[3] Dept. of Computer Science, Memorial University of Newfoundland,
St. John's, NL A1B 3X5, Canada
banzhaf@cs.mun.ca

Abstract. We introduce a genetic programming (GP) approach for evolving genetic networks that demonstrate desired dynamics when simulated as a discrete stochastic process. Our representation of genetic networks is based on a biochemical reaction model including key elements such as transcription, translation and post-translational modifications. The stochastic, reaction-based GP system is similar but not identical with algorithmic chemistries. We evolved genetic networks with noisy oscillatory dynamics. The results show the practicality of evolving particular dynamics in gene regulatory networks when modelled with intrinsic noise.

1 Introduction and Background

In recent years, there has been significant interest in synthetic biology and the engineering of genetic circuits [1,2,3,4,5,6,7]. To this end, efforts have been made to construct small constituent subnetworks or "modules" for general use in larger genetic circuits [1,6]. Typically, synthetic genetic circuits [1] are either designed by hand or by using the directed evolution paradigm *in vivo* [7]. This process is both time–consuming and expensive. Alternatively, evolutionary approaches *in silico* have shown that regulatory networks can be evolved to display certain dynamical characteristics (e .g. as bistable switches or oscillators) [5,8,9]. Essentially, these approaches differ in the specific formalism describing genetic networks (e. g. piece–wise linear differential equations augmented by Boolean functions [5], differential equations corresponding to deterministic rate equations [8], artificial regulatory network model with dynamics derived from differential equations [9]). In this contribution, we introduce a genetic programming (GP) approach for evolving biochemical reaction networks based on simple enzyme kinetics which demonstrate sustained (noisy) oscillations when *simulated* as *discrete stochastic models*.

Stochasticity (or noise) is a fundamental phenomenon in many biological systems such as gene regulatory systems [10,11,12,13]. Although noise can adversely

P. Collet et al. (Eds.): EuroGP 2006, LNCS 3905, pp. 290–299, 2006.

affect cell function, it is also considered a source of robustness and stability, signal amplification, and selection of signalling pathways. Stochasticity originates from the fact that the relative statistical uncertainty for the system state is inversely proportional to the square root of the system size, i. e. the number of elements or molecules. As a result, with smaller numbers of interacting molecules, fluctuations become increasingly noticeable. Due to the uncertainty of knowing when a reaction occurs and which reaction it might be, this form of stochasticity is also called *intrinsic* stochasticity, as opposed to *extrinsic* stochasticity which results from environmental effects. Here, only intrinsic stochasticity is considered.

In order to model intrinsic noise, we use the stochastic simulation algorithm (SSA) of Gillespie [14]. The SSA represents a nonlinear discrete Markov process, $X(t)$, whose elements represent the number of molecules of molecular species in a well-mixed system at time t (see Sec. 3). Since the dynamical behaviour of chemical systems can be very different in the ODE regime (where we deal with very large numbers of molecules neglecting the stochastic nature of their interactions) from the SSA regime, it is important to see how certain dynamical behaviour can evolve when there are only small numbers of certain key molecules. By taking this stochastic nature into account, this can be considered to be a more realistic scenario. Examples of different dynamical behaviours between deterministic continuous and stochastic discrete versions of a model can be found in [6,15]. Specifically, the deterministic model of blood testosterone levels in [15] shows a globally stable fixed point while its discrete stochastic counterpart shows sustained oscillations. The repressilator designed in [6] behaves in an oscillatory fashion in both regimes. However, stochastic simulations of this system exhibit large variabilities in oscillations. These and other studies confirm the relevance of considering stochasticity in modelling and analysis of biochemical systems.

In this contribution genetic networks are modelled as reaction systems. The underlying model is explained in the following section.

2 The Reaction Model

In this reaction model inspired by the work of François and Hakim [8], a genetic network is defined by a set of species (genes, mRNAs, proteins and complexes such as gene-protein bindings or protein complexes) and *elementary*, *irreversible* chemical reactions (first–order reactions, second–order reactions and homodimer formations) governing their interactions. That is, each reaction has associated reaction substrates, products and the specific rate constant. One or more elementary reactions are combined in *master* reactions which constitute the building blocks of the genetic network and correspond to biologically meaningful processes (cf. Table 1). The following seven biological (master) reactions are modelled:

1. Transcription and translation: a new gene, its mRNA, and the corresponding protein are added to the genetic network model. Elementary reactions for the basal transcription, translation, mRNA degradation, and protein degradation are generated. Unlike the reaction model in [8], transcription and translation are modelled as separate reactions.

Table 1. Set of master reactions that are the building blocks of the genetic networks. Lowercase letters followed by two underscores, such as $a\_\_$, represent genes with unbound regulatory sites. The corresponding mRNA is indicated such as in a_{mRNA}. The associated capitalized letters (A,B, etc.) represent the proteins produced by the associated genes. Protein complexes are represented using colons (i.e. a protein complex composed of proteins A and B is represented by $A{:}B$). When a promoter P is bound to an unbound gene $a\_\_$ the binding is denoted $aP\_$. The case of a repressor R bound to $aP\_$ is denoted as aPR. Each reaction is specified by a reaction rate constant that is ignored in this representation.

No.	Master Reaction	Single Reactions
1	Transcription and translation	$a\_\_ \rightarrow a\_\_ + a_{\mathrm{mRNA}}$ $a_{\mathrm{mRNA}} \rightarrow a_{\mathrm{mRNA}} + A$ $a_{\mathrm{mRNA}} \rightarrow \emptyset$ $A \rightarrow \emptyset$
2	Regulation	$a\_\_ + P \rightarrow aP\_$ $aP\_ \rightarrow a\_\_ + P$ $aP\_ \rightarrow aP\_ + a_{\mathrm{mRNA}}$ $aP\_ + R \rightarrow aPR$ $aPR \rightarrow aP\_ + R$
3	Protein modification	$A \rightarrow A^*$ $A^* \rightarrow \emptyset$
4	Dimerization	$A + B \rightarrow$ A:B $A{:}B \rightarrow A + B$
5	Partial degradation	$A{:}B \rightarrow A$
6	Catalytic degradation	$A + B \rightarrow A$
7	Partial cat. degradation	$AB + C \rightarrow A$

2. Regulation: transcriptional regulation is based on Goutsias' simplified model of transcriptional regulation of the bacteriophage λ repressor protein [16]. Each gene has two regulatory binding sites, R_1 and R_2. Binding of a transcription factor at R_1 activates transcription for every non-zero reaction rate, whereas binding at R_2 excludes any transcriptional activity and hence, represses transcription. In addition, binding of a transcription factor at R_2 requires R_1 to be occupied by another factor.
3. Protein modification: a single protein or protein complex reacts leading to an altered version of the original species (e.g. phosphorylation).
4. Dimerization: two proteins / protein complexes form a compound product.
5. Partial degradation: a protein complex degrades such that a constituent protein is the degradation product.
6. Catalytic degradation: one protein / protein complex catalyses degradation of another protein / protein complex.
7. Partial catalytic degradation: in the case of protein complexes, this reaction is a catalytic degradation where one of the proteins (or sub–complexes) being part of the complex is also the reaction product.

3 Stochastic Simulation

To analyse and evaluate the dynamics of regulatory networks given in the reaction model described above, we use Gillespie's stochastic simulation algorithm [14]. This is a method for *exact* simulation of biochemical systems that are assumed to be homogeneous and well-mixed within a constant volume.

In the following we briefly describe the functioning of the SSA according to [17]: Let the biochemical system consist of $N \geq 1$ molecular species $\{S_1, \ldots, S_N\}$ that chemically interact through $M \geq 1$ reaction channels $\{R_1, \ldots, R_M\}$. The system state at time t is described by a vector $X(t) \equiv (X_1(t), \ldots, X_N(t))^T$ where $X_i(t)$ is the number of molecules of species i at time t. Let $X(t_0) = X_0$ be the initial state. For each $j = 1, \ldots, M$ we can define the *propensity function* a_j for reaction R_j such that $a_j(X)dt$ is the probability that given $X(t) = X$, one reaction R_j will occur somewhere in the system in the next infinitesimal time interval $[t, t + dt)$. The *state-change* or *stoichiometric vector* ν_j specifies the update of the system state when reaction R_j occurred. This is defined by ν_{ji} for $i = 1, \ldots, M$, which is the change in the number of S_i molecules produced by one R_j reaction. Our SSA implementation simulates the time evolution of a system according to the *direct method*: two independent samples r_1 and r_2 of the uniform random variable $\mathbf{U}(0, 1)$ are drawn consecutively. The length of the time interval $[t, t + \tau)$ is given by

$$\tau = \frac{1}{a_0(X(t))} \ln(\frac{1}{r_1}) \,,$$

where

$$a_0(X(t)) = \sum_{j=1}^{M} a_j(X(t))$$

is the sum of all propensities. The specific reaction R_j occurring in $[t, t + \tau)$ is determined by the index j satisfying

$$\sum_{j'=1}^{j-1} a_{j'}(X(t)) < r_2 a_0(X(t)) \leq \sum_{j'=1}^{j} a_{j'}(X(t)) \,.$$

Table 2 specifies the propensity functions and non-zero entries of the state-change vectors for the three elementary reaction types: the first and second order reaction and homodimer formation (cf. Sec. 2). As the SSA becomes computationally intensive for systems with a large number of reaction channels and/or fast reactions due to large reaction rates and/or large numbers of molecules, we limit our model to small numbers of species with small population size. In fact, we keep the number of genes/mRNA/protein creations fixed and limit the number of reactions creating new species. In addition, in order to avoid "unending" calculations, the algorithm stops simulation if the reciprocal value of the summed propensities (a_0) falls below a predefined threshold (e. g. 10^{-8}). Methods to accelerate the SSA while maintaining a reasonable accuracy such as the τ-leap method, the midpoint-τ-leap method [17] or binomial leap methods [18]

Table 2. For the three types of elementary reactions we determine the propensity functions and non-zero entries of the state-change vectors for the present state $X(t) = X$. c_j is the reaction rate constant of the respective reaction.

Reaction	Propensity Function	Stoichiometric Coefficients
First order reaction $S_k \xrightarrow{c_j} S_l$	$a_j = c_j * X_k$	$\nu_{jk} = -1, \nu_{jl} = 1$
Second order reaction $S_k + S_l \xrightarrow{c_j} S_m$ with $S_k \neq S_l$	$a_j = c_j * X_k * X_l$	$\nu_{jk} = \nu_{jl} = -1, \nu_{jm} = 1$
Homodimer formation $S_k + S_k \xrightarrow{c_j} S_l$	$a_j = c_j * X_k * (X_k - 1)/2$	$\nu_{jk} = -2, \nu_{jl} = 1$

are not used as they allow all the reaction channels to fire within each time step with a certain frequency.

4 The GP System

Here we use a GP–based algorithm to evolve genetic networks that obtain sustained oscillations in an arbitrarily chosen protein or mRNA. Typical GP algorithms use tree–based encodings [19,20]. This allows an individual solution to be parsed into an equation where order of operations is important. However, this encoding is inappropriate for this application since the order in which reactions are triggered is chosen randomly. Instead, we choose a set–based encoding scheme where each individual is represented by a set of biochemical reactions. This *reaction-* or *set-based* GP approach is very similar (but not equivalent) to a GP-approach based on algorithmic chemistries [21] which, unlike our approach acts on instruction multisets and aims to create functioning algorithms.

Each individual initially starts with two gene (+ mRNA + protein) creation reactions (reaction 1) and three other master reactions. This is not essential for evolution but complies with our intention of studying small regulatory systems consisting of two genes. The individual master reactions (2 to 7) listed in Section 2 are added to an individual through subsequent mutation steps. Reactions of type 1 cannot be added to genetic circuits during evolution. When a reaction is added to the network this may introduce a new product. Therefore, its list of species, i. e. proteins and bindings, is updated. Reaction rates are uniformly drawn between 0 and 1 and reactants are randomly chosen from the list of suitable reactants while avoiding the generation of duplicate reactions. Other mutation operations involve deletions of reactions and modification of reaction rates. The deletion of a reaction eventually includes the deletion of the product introduced by this reaction and of all other reactions using the product as a substrate. Reaction rates are modified by multiplication with a random number from $U[0, 2]$. At the beginning of each evolutionary run, the initial

concentrations of proteins and protein complexes are randomly chosen from $\{1, 2, \ldots, 10\}$ and remain fixed for the entire evolution.

Recognizing sustained oscillations from noisy signals is the crucial point in our evolution. The individual's fitness is calculated by simulating the corresponding reaction system over a predefined simulation time using the SSA. In a second step, the resulting trajectory of length N for a specified species is assessed according to its oscillatory behaviour. This is accomplished by applying the Fast Fourier Transform. A rather simple and coarse indicator for oscillatory behaviour is the ratio of the summed magnitude over the first $N/2+1$ Fourier values, M_{sum}, to the maximum magnitude within a predefined frequency range (e. g. $1/N \ldots 1/4$), M_{max}. In summary, the fitness value is calculated as $1 - M_{max}/M_{sum} + 1/M_{max}$.

Stochasticity in the outcome of the fitness evaluation is a problem that must be dealt with: a trajectory (resulting from an SSA run) may show a certain behaviour but another simulation may be different due to the different stochastic path. To get a reliable result we perform several simulations. Calculating an "average" trajectory from the resulting single trajectories and performing fitness evaluation on this mean behaviour would be misleading since the average trajectory might not match any single trajectory. Therefore, we calculate the mean fitness over all SSA runs.

The selection method driving evolutionary dynamics is a simple $(\mu + \mu)$ strategy: each individual generates one offspring by performing two mutations on its own copy; the best μ out of 2μ individuals build the new generation. The generational GP algorithm is implemented as a synchronous parallel GP using MPI (message passing interface). Evolution is terminated if the number of generations without fitness improvement exceeds a certain threshold. Table 3 lists the most important parameters and their values used in evolutionary runs.

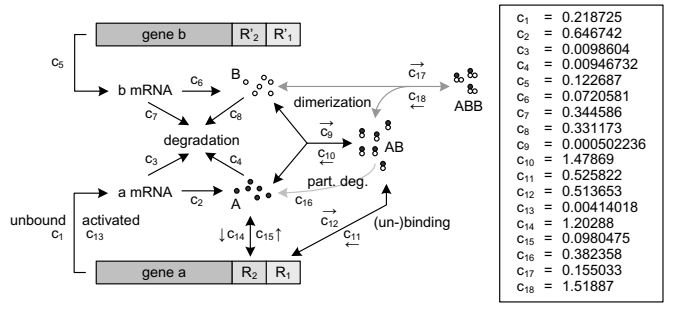
Table 3. These are some exemplary parameter settings of our GP system. With these settings we evolved the genetic network shown in Fig. 1.

Parameter	**Value**
no. of SSA runs (for each individual)	20
length of (SSA) simulation (in time units)	2048
GP termination threshold (in generations)	100
population size	100
max. no. of master reactions:	
gene/mRNA/protein creation	2
regulation	2
protein modification	2
dimerization	3
partial degradation	2
catalytic degradation	2
partial catalytic degradation	2
mutation probabilities:	
add reaction	0.1
delete reaction	0.1
modify rate constant	0.5

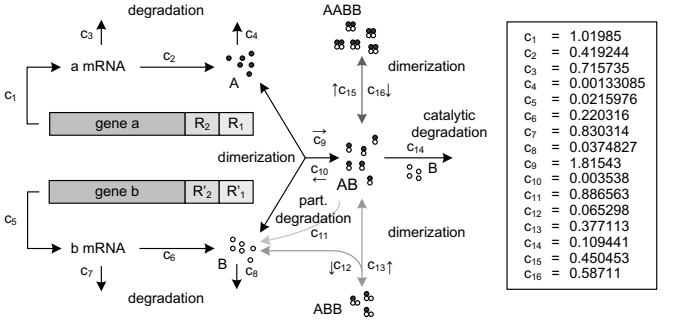
5 Preliminary Results

Here we present two evolved genetic networks showing noisy oscillatory dynamics. They are good representatives of other evolved networks featuring noisy oscillatory behaviour resulting from a total of 50 GP runs. So far, we have not focused on the performance of the evolution itself. However, the evolved solutions were usually generated in the first 150 generations. Figure 1(a) shows a regulatory genetic network that utilizes the regulation (master) reaction (cf. Table 1). This exerts negative feedback on the transcription and translation of gene a whenever protein A binds to the regulatory site R_2, thus repressing the transcription activated by the binding of dimer AB at R_1. The Figures 2(a) and 2(b) show the resulting dynamics for a single simulation run.

A second genetic regulatory network is depicted in Figure 1(b). This network generates a less regular form of oscillation in the concentration of protein A



(a)



(b)

Fig. 1. Schematic representation of two evolved genetic networks exhibiting noisy oscillatory dynamics. (a) A core element in this genetic network that shows a regular, sustained oscillation in protein A concentration (cf. Fig. 2(a)) is the negative autoregulation of gene a. (b) This network comes without the regulation reaction but still shows some form of oscillatory dynamics in the concentrations of protein A (cf. Fig. 2(c)). Apparently, post-translational modifications are sufficient for generating pulsed signals.

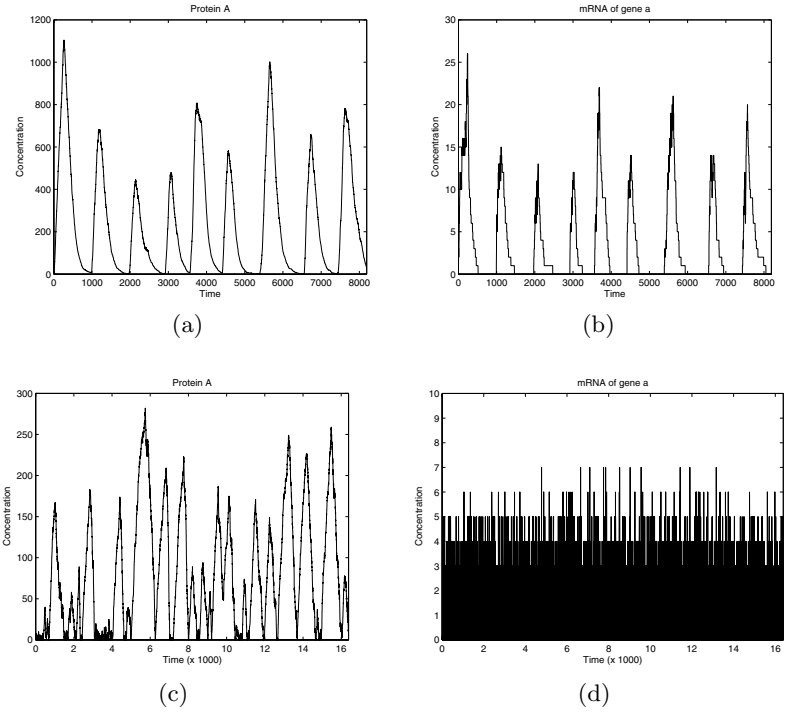


Fig. 2. Simulation results showing the concentration dynamics of protein A and a mRNA of the genetic regulatory networks in Figure 1(a) (a,b) and 1(b) (c,d)

(cf. Figure 2(c)). It does so without utilizing any direct regulation reactions (cf. Table 1) but by solely using post-translational modifications. A similar network based only on post-translational reactions that exhibits a sustained oscillation in the ODE model was also reported in [8]. In this model, however, one protein complex is constantly produced without being consumed. The Figures 2(c) and 2(d) depict the concentration dynamics of protein A and a mRNA. The dynamics of protein A are controlled by only a few other molecules that occasionally initiate the production of protein A. This leads to a short burst in the molecular concentration that appears quite regular. Note that all evolved networks were simulated several times to verify sustained oscillatory behaviour. Moreover, the corresponding ODE models of the networks in Figure 1 do not show oscillatory behaviour which underpins the necessity of stochastic simulation.

6 Discussions and Suggestions for Future Work

In this contribution, we present a GP approach for evolving genetic regulatory networks. Unlike others evolutionary approaches [5,8] we model those networks as sets of elementary reactions based on simple enzyme kinetics and simulate the network using Gillespie's exact SSA. We showed that evolution of noisy

oscillatory dynamics in genetic regulatory networks is practical also in the discrete, stochastic regime. The networks found in [8] and the ones presented here show that post-translational modifications can be crucial to network function. As such, network function in this model cannot be understood by focusing only on transcriptional interactions. This is an important consideration for researchers in the bioinformatics community since such post-translational interactions are often omitted from such models.

In our simulations, the number of genes was fixed but can be changed to evolve specific dynamical behaviour in larger networks. For evolving desired dynamics in the concentrations of several species, the fitness function must be redesigned. Using our fitness function for detecting oscillatory behaviour in one protein showed success. Evolutions with more sophisticated fitness functions are worthy of future consideration. At this point, parameter settings are heuristics. Changing such settings might accelerate evolution. In order to obtain a better understanding of the solution space and our representation, explorations on the fitness landscape should be performed.

We also plan to evolve genetic regulatory networks with other types of dynamics. Since bistable behaviour can be observed in many biological systems, evolution of genetic toggle switches under intrinsic noise would be of particular interest [22]. An additional step would be to consider time delays. By using delay-SSA (DSSA), a modified SSA algorithm incorporating delay effects [23,24], we can model natural behaviour of processes such as transcription and translation in a more detailed manner since they do not occur instantaneously [25].

This contribution shows how methods from evolutionary computation can be used to achieve improved models of genetic regulatory networks, a better understanding of regulation in cells, the finding of functional design principles and the search for novel genetic networks.

Acknowledgements

KB would like to thank the Australian Research Council for funding via his Federation Fellowship. DK was supported by the National Science and Engineering Research Council of Canada PGS-D3. WB has been supported by NSERC under grant RGPIN 283304-04.

References

1. Kobayashi, H., Kaern, M., Chung, K., Gardner, T.S., Cantor, C.R., Collins, J.J.: Programmable cells: Interfacing natural and engineered gene networks. Proc. Natl. Acad. Sci. **101(22)** (2004) 8414–8419
2. Isaacs, F.J., Hasty, J., Cantor, C.R., Collins, J.J.: Prediction and measurement of an autoregulatory genetic module. Proc. Natl. Acad. Sci. **100**(13) (2003) 7714–7719
3. Gardner, T.S., Cantor, C.R., Collins, J.J.: Construction of genetic toggle switch in *escherichia coli*. Nature **403**(6767) (2000) 339–342
4. Guet, C.C., Elowitz, M.B., Hsing, W., Leibler, S.: Combinatorial synthesis of genetic networks. Science **296** (2002) 1466–1470

5. Mason, J., Linsay, P.S., Collins, J.J., Glass, L.: Evolving complex dynamics in electronic models of genetic networks. Chaos **14(3)** (2004) 707–715
6. Elowitz, M.B., Leibler, S.: A synthetic oscillatory network of transcriptional regulators. Nature **403**(6767) (2000) 335–338
7. Yokobayashi, Y., Weiss, R., Arnold, F.H.: Directed evolution of a genetic circuit. Proc. Natl. Acad. Sci. **99**(26) (2002) 16587–16591
8. François, P., Hakim, V.: Design of genetic networks with specified functions by evolution *in silico*. Proc. Natl. Acad. Sci. **101**(2) (2004) 580–585
9. Kuo, P.D., Banzhaf, W., Leier, A.: Network topology and the evolution of dynamics in an artificial genetic regulatory network model created by whole genome duplication and divergence. BioSystems (2006) In press.
10. McAdams, H.H., Arkin, A.: Stochastic mechanisms in gene expression. Proc. Natl. Acad. Sci. **94**(3) (1997) 814–819
11. Arkin, A., Ross, J., McAdams, H.H.: Stochastic kinetic analysis of developmental pathway bifurcation in phage λ–infected Escherichia coli cells. Genetics **149**(4) (1998) 1633–1648
12. Elowitz, M.B., Levine, A.J., Siggia, E.D., Swain, P.S.: Stochastic gene expression in a single cell. Science **297**(5584) (2002) 1183–1186
13. Hasty, J., Collins, J.J.: Translating the noise. Nature Genetics **31**(1) (2002) 13–14
14. Gillespie, D.T.: Exact stochastic simulation of coupled chemical reactions. J. Phys. Chem. **81**(25) (1977) 2340–2361
15. Heuett, W.J., Qian, H.: A stochastic model of oscillatory blood testoterone levels. To appear in Bulletin for Mathematical Biology (2005)
16. Goutsias, J.: Quasiequilibrium approximation of fast reaction kinetics in stochastic biochemical systems. J. Chem. Phys. **122**(184102) (2005) 1–15
17. Gillespie, D.T.: Approximate accelerated stochastic simulation of chemically reacting systems. J. Chem. Phys. **115**(4) (2001) 1716–1733
18. Tian, T., Burrage, K.: Binomial leap methods for simulating stochastic chemical kinetics. J. Chem. Phys. **121**(21) (2004) 10356–10364
19. Koza, J.R.: Genetic Programming. MIT press (1992)
20. Banzhaf, W., Francone, F.D., Keller, R.E., Nordin, P.: Genetic programming: an introduction. Morgan Kaufmann (1998)
21. Lasarczyk, C.W.G., Banzhaf, W.: An algorithmic chemistry for genetic programming. In Keijzer, M., Tettamanzi, A., Collet, P., Tomassini, M., van Hemert, J., eds.: Proceedings of the 8th European Conference on Genetic Programming. Volume 3447 of LNCS., Springer (2005) 1–12
22. Tian, T., Burrage, K.: Stochastic models for regulatory networks of the genetic toggle switch. Proc. Natl. Acad. Sci. (2005) Submitted.
23. Barrio, M., Burrage, K.: Oscillatory regulation of hes1: Discrete stochastic delay modelling and simulation. PLoS Computational Biology (2005) Submitted.
24. Bratsun, D., Volfson, D., Tsimring, L.S., Hasty, J.: Delay-induced stochastic oscillations in gene regulation. Proc. Natl. Acad. Sci. **102**(41) (2005) 14593–14598
25. Monk, N.A.M.: Oscillatory expression of Hes1, p53, and NF-kappaB driven by transcriptional time delays. Curr. Biol. **13**(16) (2003) 1409–1413