Available online at www.sciencedirect.com
ScienceDirect
Journal of Theoretical Biology 246 (2007) 725–745
Journal of Theoretical Biology
www.elsevier.com/locate/yjtbi

Studying genetic regulatory networks at the molecular level: Delayed reaction stochastic models

Rui Zhu[a,b,*], Andre S. Ribeiro[b,c,e], Dennis Salahub[a,b], Stuart A. Kauffman[b,c,d]

[a]Department of Chemistry, University of Calgary, Canada T2N 1N4
[b]Institute for Biocomplexity and Informatics, University of Calgary, Canada T2N 1N4
[c]Department of Physics and Astronomy, University of Calgary, Canada T2N 1N4
[d]Department of Biological Sciences, University of Calgary, Canada T2N 1N4
[e]Centre for Computational Physics, University of Coimbra, P-3004-516 Coimbra, Portugal

Received 24 August 2006; received in revised form 22 January 2007; accepted 25 January 2007
Available online 6 February 2007

Abstract

Current advances in molecular biology enable us to access the rapidly increasing body of genetic information. It is still challenging to model gene systems at the molecular level. Here, we propose two types of reaction kinetic models for constructing genetic networks. Time delays involved in transcription and translation are explicitly considered to explore the effects of delays, which may be significant in genetic networks featured with feedback loops. One type of model is based on delayed effective reactions, each reaction modeling a biochemical process like transcription without involving intermediate reactions. The other is based on delayed virtual reactions, each reaction being converted from a mathematical function to model a biochemical function like gene inhibition. The latter stochastic models are derived from the corresponding mean-field models. The former ones are composed of single gene expression modules. We thus design a model of gene expression. This model is verified by our simulations using a delayed stochastic simulation algorithm, which accurately reproduces the stochastic kinetics in a recent experimental study. Various simplified versions of the model are given and evaluated. We then use the two methods to study the genetic toggle switch and the repressilator. We define the "on" and "off" states of genes and extract a binary code from the stochastic time series. The binary code can be described by the corresponding Boolean network models in certain conditions. We discuss these conditions, suggesting a method to connect Boolean models, mean-field models, and stochastic chemical models.

Keywords: Systems biology; Genetic regulatory networks; Boolean networks; Stochastic models; Time delay

1. Introduction

Advances in experimental genetics and molecular biology have brought about a flood of genomic data. This creates a new challenge to understanding how genes and proteins work collectively, i.e. the analysis of these data. This challenge leads to a significant increase of computer applications for modeling and data interpretation methods. The aim is to develop computer simulations that mimic biological phenomena, data or patterns, such as complex biochemical reactions and gene networks in cellular media (Turner et al., 2004).

One interesting biological phenomenon observed in the experiments is that a genetically identical (isogenic) population of cells exposed to the same environmental conditions can have phenotypically distinct individuals (Raser and O'Shea, 2005). The question that arises is: How does the phenotypic variability arise from identical genetic networks? A simple answer is that cells are intrinsically stochastic systems. This is because biochemical processes involved in gene expression and regulations are the source of variability due to intrinsically stochastic essence of chemical reactions. Besides intrinsic molecular fluctuations,

*Corresponding author. Department of Chemistry, University of Calgary, 2500 University Drive NW, Calgary, AB, Canada T2N 1N4. Tel.: +1 403 220 2961; fax: +1 403 210 8655.
E-mail address: rzhu@ucalgary.ca (R. Zhu).

0022-5193/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jtbi.2007.01.021

extrinsically related stochastic factors, such as environmental differences and state differences between cells, contribute to the cell–cell variability as well. Also, genetic variation (including genetic mutations and DNA duplication errors) is another important factor. Naturally extrinsic factors and genetic variation all finally function as sources of variability through stochastic reaction events. Accordingly, constructing a modeling framework for gene networks based on stochastic reaction kinetics is critical to explore stochastic life phenomena.

Recent studies on gene expression analysis (Ozbudak et al., 2002; Blake et al., 2003; Raser and O'Shea, 2004; Kærn et al., 2005; Cai et al., 2006; Yu et al., 2006) and simple genetic network synthesis (Elowitz and Leibler, 2000; Gardner et al., 2000; Atkinson et al., 2003; Kramer et al., 2004; Hooshangi et al., 2005; Pedraza and van Oudenaarden, 2005; Rosenfeld et al., 2005; Guido et al., 2006) provided considerable experimental information about phenotypic variability and stochasticity of gene expression. The corresponding theoretical investigations incorporating the stochastic nature of chemical reactions indeed gave reasonable explanations for the experimental observations. One of the most used theoretical methods is the stochastic simulation of the chemical master equation which simply requires a chemical reaction mechanism (model) and a stochastic simulation algorithm (SSA) (Gillespie, 1976, 1977, 1992). This simulation mimics the evolution of a reaction system by carrying out single steps which specify the next effective random collision and when this event will happen. This method assumes that in a cell all the interactions involved in a gene network are pure chemical reaction processes, which are in a homogeneous state. While physical processes like diffusion and transportation are also important in gene network kinetics, and cells are by no means homogenous compartments, using this method has proved to be a useful tool to investigate stochasticity in gene networks (McAdams and Arkin, 1997, 1998; Arkin et al., 1998), especially in prokaryotes.

Real gene network systems are composed of a large number of reactions and reacting species. There are too many items to include them all in models, due to our current limited knowledge and understanding. Even if we had a complete model consisting of all the reactions and species, it is also rather difficult to find an effective method to analyze such a complex model. Thus, we are currently using simplified models. Two widely used simplification ideas are: designing effective reactions and making virtual reactions. The effective reactions are designed to include the inputs and outputs of a biochemical process without explicitly considering a series of intermediate reactions. For example, we can use one effective first-order reaction, mRNA $\rightarrow$ Protein, to model the whole translation process without considering the multi-step elongation process. In contrast to effective reactions, virtual reactions do not explicitly correspond to real biochemical processes. They are built from the mathematical functions that model biochemical functions such as gene inhibition. For example, a virtual zeroth-order reaction like $\xrightarrow{\kappa_1} P_1$ with $\kappa_1 = a/(1+P_2^b)$ can be used to model the expression of gene 1 which is inhibited by the expressed products of gene 2. Here, P_1 and P_2 are the expressed products of genes 1 and 2, respectively, and a and b are constants. The purposes of using effective reactions and virtual reactions are the same, i.e. to use simple single-step reactions to model complex multiple-step biochemical processes. While current models in the literature based on these ideas are very simple in comparison with the complexity of real gene systems, it seems that these simplified models work well to capture key features of the studied gene systems.

However, most of the methods using effective reactions and virtual reactions neglect temporal delays that inevitably exist due to the time lags between the inputs and outputs of the modeled events. Especially, for the two core processes in gene expression, transcription and translation, it may take considerable time for a RNA polymerase/ribosome to generate an RNA/polypeptide depending on the length of a gene, which varies significantly from gene to gene. It should be noted that these delays seem to be unimportant to the stochastic kinetics of the gene systems in previous studies where only the non-delay models were used. We note that previously studied systems are simple synthetic genetic systems, such as the toggle switch (Gardner et al., 2000), and that most of the interesting states are steady states. It is understandable that, for a delayed reaction and a non-delay reaction with the same input rate, the rates of their outputs would become the same when their steady states are reached after some transients. Using non-delay reactions is thus enough to study the equilibrium states of such simple synthetic gene systems. In these cases, the transients are simple. But, in real genetic regulatory networks, the regulation is so complex (associated with many feedback loops) that gene activation and gene inhibition could occur very frequently. In such cases, time delays, especially for transcription and translation (Lewis, 2003; Monk, 2003; Gaffney and Monk, 2006), should be included in the model to capture the features of transients (Bratsun et al., 2005; Veflingstad et al., 2005). These transients can be very long so that the behavior of gene systems could depend critically on transients rather than steady states. For example, the protein production delays have proven able to create long transients between attractors, even for simple genetic networks (Ribeiro et al., 2006a).

More importantly, the introduction of delays makes the complex network systems much easier to theoretically study at the molecular level. This allows us to easily study the kinetics of the systems by exploring the delay effects. This not only can help us to understand better the properties of real genetic networks, but also may help guide the design of robust genetic devices. Since transcription and translation are two core processes in genetic regulatory networks, in this work we will focus on the time delays involved in these two processes.

To simulate delayed reactions, the commonly used Gillespie SSA, must be modified. There are several options available. Gibson and Bruck proposed a method to impose the delay effect into their version of Gillespie's algorithm (Gibson and Bruck, 2000). Ramsey et al. (2005) also reported that single-output delayed reactions can be introduced in stochastic simulations performed in their software Dizzy. Very recently, Bratsun et al. (2005) provided a detailed delayed Gillespie algorithm. Here, we use a generalized delay algorithm (Roussel and Zhu, 2006), which can handle more than one delayed output event for one input event.

The focus of this article is put on the stochastic reaction modeling approach. Yet, the Boolean network method (Kauffman, 1993), which has a higher level of description in models of genetic networks, comes with a tightly related issue of great interest: in the Boolean network description, how should one interpret "0" and "1" in the stochastic time series obtained from the corresponding stochastic model? Accordingly, we investigate the connection between the two kinds of models through two examples. The text of this article is organized as follows. In Section 2, we introduce two kinds of delayed reactions: delayed effective reactions and delayed virtual reactions, and describe the methods we use to construct two types of genetic network models from these two kinds of reactions, respectively. In Section 3, we first propose a single-gene expression model and employ it to study the dynamics of a gene which is expressed in a certain repressed condition. Next, we derive various simplified versions from this model and evaluate their performances under given conditions. Then, we study two simple genetic regulatory networks, a two-gene toggle switch and a three-gene oscillator, using the proposed methods. Finally, the studied stochastic models are compared with the corresponding Boolean network models. In Section 4, we conclude with the key features of the delayed models and the issues of using these models in further studies of gene networks. For simplicity, extrinsic fluctuations and genetic variations are not considered in this work.

2. Methods

As stated in Section 1, we are exploring genetic systems using the stochastic reaction kinetics method. This method requires a chemical reaction model and an SSA. We first provide in detail two types of delayed models and then briefly describe the SSA used here to perform stochastic simulations. All the models are reaction-based models which consist of only simple chemical reactions, such as zeroth-order, first-order (unimolecular), and second-order (bimolecular) reactions.

2.1. Process-based effective chemical reactions

Our effective chemical reactions are designed to model multiple-step biochemical processes with single-step delayed reactions. A regular reaction equation has two sides: the reactant side and the product side. The delayed reaction equation has the same reactant side as a regular reaction does, but has a different product side. We start this section from a single-gene expression model that explicitly includes two parts, i.e. transcription and translation. For the transcriptional part, we have

$$\mathrm{RNAP}(t)+\mathrm{Pro}(t)\xrightarrow{k_1}\mathrm{Pro}(t+\tau_1)+\mathrm{RBS}(t+\tau_1)+\mathrm{RNAP}(t+\tau_2)+R(t+\tau_2). \quad (1)$$

This reaction is represented using the delayed mass-action representation of chemical mechanisms (Roussel, 1996). On the reactant side, an RNA polymerase (RNAP) binds to the promoter region (Pro), forming an elongation complex (an intermediate product, not given explicitly). This process is the first step of transcription initiation which is modeled as a bimolecular reaction with the probability rate constant k_1. Here, RNAP and Pro are the inputs of the transcription process; their reaction is the corresponding input event. The elongation complex is then ready to walk along the DNA strand. On the product side, the corresponding output events are: τ_1 time units later, the promoter is cleared; $\tau_2>\tau_1$ later, the RNAP is ready to release the DNA, producing a primary mRNA transcript (R). Also, the leader region of the transcript containing the ribosome binding site (RBS) is assumed to be produced at the same time as the promoter is cleared. Pro, RBS, RNAP, and R are the outputs of the transcription process.

Note that the transcription initiation frequency, which determines the input rate of the transcription process, is controlled by k_1 and τ_1. The former determines how frequently the binding event happens if the Pro is available (assuming that the RNAP is abundant), and the latter determines how long it takes for the Pro to be available again after its shut-off due to the binding event. Statistically, after a transient of τ_2, the output rates of RNAP and R reach the input rate if the input rate is kept unchanged. For a non-delay reaction, on the contrary, all the output rates reach the input rate immediately. This is how the production delay τ_2 makes the original difference compared with the non-delay case. *If the average time for one input event (determined by* $(RNAP\times k_1)^{-1}+\tau_1$, *where RNAP is the number of free RNAPs) is much larger than* τ_2, τ_2 *can be neglected.* This follows because the input rate is so slow that it takes a long time compared with the production delay to obtain the outputs. In such situations, the effect of the production delay may not be significant. In addition, τ_1 can be omitted as well if it is much smaller than $(\mathrm{RNAP}\times k_1)^{-1}$.

In fact, the transcription process involves many reaction steps which can be classified into three stages: initiation, elongation and termination (von Hippel, 1998). Especially in the elongation stage, there are at least two repeating steps: A phosphodiester bond forms between two ribonucleotides, generating chemical energy for the RNAP to step

forward to the next position; one NTP paired with the nucleotide in that position binds to the RNAP. The two elongation steps repeat until the last nucleotide of the gene is reached. In most dynamic studies concerning systems biology, these detailed intermediate reactions are not of interest. What interests us is to know how long/often, statistically, the two key species, i.e. the Pro and the RBS, would be produced since the populations of these species are tightly related to the population of expressed proteins (see below for the explanation). The delayed transcription reaction (1) is designed for this purpose.

For the translational part, we have a delayed reaction analogous to reaction (1)

$$\text{Ribosome}(t)+\text{RBS}(t)\xrightarrow{k_2}\text{RBS}(t+\tau_3)+\text{Ribosome}(t+\tau_4)+P(t+\tau_5). \quad (2)$$

Once a free ribosome successfully binds to the RBS, after τ_3, the RBS is cleared; and $\tau_4>\tau_3$ time units after the binding, a protein (P) is produced and the ribosome releases the mRNA. k_2 is the probability rate constant for Ribosome–RBS binding events. The parameters, k_2, τ_3, and τ_4, have the same relationships as k_1, τ_1, and τ_2 do for the transcription reaction. In addition, the RBS loses its ability of binding to ribosomes through the following degradation process of RBS:

$$\text{RBS}\xrightarrow{k_3}\text{decay}. \quad (3)$$

The degradation mechanism is that the RBS is attacked by ribonuclease E (Yarchuk et al., 1992). For simplicity, we assume that the population of ribonuclease E is constant. The degradation of proteins can be modeled as another unimolecular reaction

$$P\xrightarrow{k_4}\text{decay}. \quad (4)$$

We propose reactions (1)–(4) for a model of gene expression mainly based on the model given by Kierzek et al. (2001) for prokaryotes. The main biological feature of the prokaryotic gene expression is shown in the model: mRNA is translated during transcription. For eukaryotes, one of the major differences in gene expression from prokaryotes is that the mRNA is processed before transport to the cytoplasm where it is translated. Thus, we could also have a simple model for eukaryotic gene expression by replacing reaction (1) with

$$\text{RNAP}(t)+\text{Pro}(t)\xrightarrow{k_1}\text{Pro}(t+\tau_1)+\text{RBS}(t+\tau'_1)+\text{RNAP}(t+\tau_2)+R(t+\tau_2), \quad (5)$$

where τ'_1 is the time it takes to form an RBS, including the time for the processing of primary transcripts and transport of the processed transcripts to cytoplasm imposing always $\tau'_1>\tau_2$. Note that the models mentioned below refer to prokaryotic cases only unless otherwise stated.

As seen from reactions (1) and (2), the transcription and translation processes are modeled in a similar way due to their biological similarity: an RNAP binds to the Pro in transcription, while a ribosome binds to the RBS in translation. Note that the Pro represents the state of a gene. For the current single-gene case, its value can be either one or zero, denoting the on or off states of the gene. Here, the "on state" means that the Pro can freely bind to an RNAP, while the "off state" means that the Pro is unavailable for binding due to some reason, for example, the Pro is not released yet or it is blocked by a repressor. On the product side of reaction (1), what really dominates the kinetics of proteins is RBS rather than R. Once an RBS forms, proteins can start being produced through reaction (2). Thus, the Pro controls the production of RBSs which successively controls the production of proteins. Besides the Pro and the RBS, another two species, RNAP and ribosome, also affect the gene expression. However, only Pro and RBS play a dominant role in gene expression kinetics because their populations are much smaller than those of RNAPs and ribosomes, and therefore act as the limiting factors for the gene expression process. It is biologically reasonable that only the leader regions of a gene and an mRNA are most associated with the kinetics of gene expression. The other two species, R and P, are included so that we can compare the kinetics of R and P directly with those of experimental systems or other models.

We can derive various simplified versions from the model (reactions (1) and (2)). For example, by assuming that RNAPs and ribosomes both are in a steady state, $\tau_2\gg\tau_1$, and that $\tau_4\gg\tau_3$, reactions (1) and (2) become, respectively,

$$\text{Pro}(t)\xrightarrow{k_1}\text{Pro}(t)+\text{RBS}(t)+R(t+\tau_2), \quad (6)$$

$$\text{RBS}(t)\xrightarrow{k_2}\text{RBS}(t)+P(t+\tau_4). \quad (7)$$

The effects of RNAP and ribosome are absorbed into k_1 and k_2, respectively, since their populations are fixed. We note that Pro and RBS in reactions (6) and (7) behave like catalysts: One Pro/RBS is repeatedly used to produce more than one product. For reaction (6), it means the gene is always "on", so it is actually a zeroth-order reaction

$$\xrightarrow{k_1}\text{RBS}, \quad (8)$$

where we delete R and keep the dynamics-related species only. In reaction (7), the production rate of proteins depends on the number of RBSs which changes over time. Thus, reaction (7) is a first-order self-catalytic reaction. It is worth mentioning that the reactions commonly used in the literature for transcription and translation are:

$$\xrightarrow{k_1}\text{RNA}, \quad (9)$$

$$\text{RNA}\xrightarrow{k_2}P. \quad (10)$$

Comparing them with our simplified model (reactions (8) and (7)), we get the following results. RNA in the literature model (reactions (9) and (10)) stands for RBS in our model.

Reaction (9) is then the same as reaction (8), but reaction (10) is quite different from reaction (7). There exist two differences: Unlike reaction (7), reaction (10) is not a self-catalytic reaction so that one transcript can only produce one product; once the input rate of translation drops due to the decrease in the number of transcripts, the production rate of proteins drops at the same time (no delay/transient). To improve the literature model, for the former difference, we introduce the protein burst size, n, the average production rate of proteins per transcript (Ozbudak et al., 2002); for the latter, we need to add a delay for the protein production. Reaction (10) is thus updated to, in terms of reaction (7)

$$\mathrm{RBS}(t) \xrightarrow{k_2} nP(t+\tau_4). \tag{11}$$

Reactions (8) and (11) are a simplified version of the gene expression model for prokaryotes that we directly derive from our first model (reactions (1) and (2)). It is worth emphasizing that commonly the protein production delay, τ_4, cannot be neglected since the time it takes for one input event to happen is commonly smaller than τ_4. For example, in *Escherichia coli*, the interval between translation initiations ranges from 1.7 to 16 s (Kennell and Riezman, 1977); the speed for ribosomes to walk along the mRNA ranges from 8 to 15 amino acids/s (Talkad et al., 1976), and the average length of an mRNA molecule is about 1200 bases (Stryer, 1988), giving $\tau_4 = 26$–50 s. In addition, τ_4 would be much larger if it includes the time for the post-translational protein assembly process (Yu et al., 2006). Reaction (11) could work for eukaryotes as well, but the transcription reaction (8) should be modified to

$$\xrightarrow{k_1} \mathrm{RBS}(t+\tau_1'). \tag{12}$$

Reaction (12) accounts for the long delay to form an RBS in eukaryotes, compared to prokaryotes.

In addition, we can simplify our first gene expression model by combining transcription and translation in a single-step delayed reaction as follows (Ribeiro et al., 2006b):

$$\mathrm{RNAP}(t) + \mathrm{Pro}(t) \xrightarrow{k_1} \mathrm{Pro}(t+\tau_1) + \mathrm{RNAP}(t+\tau_2) + R(t+\tau_2) + nP(t+\tau_3). \tag{13}$$

In this delayed reaction, the translation process is viewed as a delayed output event of the transcription process. Those parameters in reaction (13) except n and τ_3 are the same in reaction (1). n is used to model the bursts of protein production and τ_3 is the delay of protein production in this reaction. The two parameters are obtained by fitting the average kinetics of reaction (13) with that of the first model. With the same reasoning used above, we can have a further simplified model obtained from reaction (13)

$$\xrightarrow{k_1} nP(t+\tau_3). \tag{14}$$

Here, k_1 is the transcription initiation probability rate constant, n determines the burst size of protein production, and τ_3 is the protein production delay.

We presented above several versions of models of gene expression. The next question is: how can we connect up these basic components to build a reasonable genetic regulatory network model? Many theoretical and experimental techniques have been used to study this issue. In our framework of chemical reactions, we assume that all the genetic regulations are realized through chemical reactions and that signals are transmitted via chemical species. We provide below a strategy to construct simple genetic networks by including some additional reactions and reacting species.

Generally there are two basic regulatory mechanisms in genetic networks: gene inhibition and gene activation (Alberts et al., 2002). Regulations can happen in transcription and/or translation. We consider only transcriptional regulation due to its dominant role in gene expression. In a regular transcriptional inhibition, a repressor binds to an operator site which is close to the promoter region, blocking the RNAP from walking on the DNA strand (Cowell, 1994), while in a regular transcriptional activation process, an activator binds to some part of the DNA resulting in faster transcription initiation through a recruitment mechanism (Ptashne and Gann, 1997). One can have several distinct effectors (repressors and activators) involved in a single gene regulation process. For simplicity, only one effector per gene regulation is considered at this point. We note that, in both the inhibition and activation processes, one common point is that the state of the promoter is changed by an effector. We thus include the following two reactions of gene regulation:

$$\mathrm{Pro} + P \xrightarrow{k_3} \mathrm{Pro}P, \tag{15}$$

$$\mathrm{Pro} + P \xrightarrow{k_4} \mathrm{Pro}P, \tag{16}$$

where P denotes effector which is a protein expressed from some gene and Pro$P$ denotes the changed state of the promoter. Reaction (16) allows the changed state to be reversible to the normal state. For a gene inhibition process, Pro$P$ is the repressed state of the Pro. Once the Pro$P$ forms through reaction (15), transcription stops until the Pro (normal state) recovers through reaction (16). For a case of gene activation, Pro$P$ is the activated state of the Pro; besides, we need to add another "high-speed" synthetic channel for this activated Pro. For example, once this activated Pro forms, reaction (1) is replaced by the following reaction:

$$\mathrm{RNAP}(t) + \mathrm{Pro}P(t) \xrightarrow{k_1'} \mathrm{Pro}P(t+\tau_1) + \mathrm{RBS}(t+\tau_1) + \mathrm{RNAP}(t+\tau_2) + R(t+\tau_2), \tag{17}$$

where $k_1' > k_1$. In addition, if the effector P is abundant so that its population is almost fixed, we can simplify

reactions (15) and (16) to a first-order reversible state change reactions like Pro $\Leftrightarrow$ Pro$P$. For this simplified case, we may also assign a "low-speed" channel like (17) with $k'_1 < k_1$ to the repressed Pro. This is another way to model gene inhibition, which is commonly used in the literature (Kærn et al., 2005). In our strategy, we do not consider this channel since we assume that the repressed Pro is completely turned off. A slightly simpler version of the above regulatory equations was also proposed (Ribeiro et al., 2006b) to build gene regulatory networks of any size, which can be used to do an ensemble approach study (Kauffman, 2004) instead of Boolean networks (Kauffman, 1969).

In summary, the two basic transcriptional regulation mechanisms are implemented by this strategy: the transcription initiation frequency is regulated by changing the state of the Pro. Note that this strategy only applies to those models explicitly including the Pro. It is worth emphasizing that the natural transcriptional regulation in cells is far more complex than the mechanisms proposed above. Using this strategy is a sensible first step toward constructing simple synthetic networks. We will use this method in the next part to build a model of single gene expression in a repressed condition and a model of toggle switch.

The above proposed reactions for gene expression and genetic regulations are simple chemical reactions. Besides being used in the framework of stochastic reaction kinetics, these reactions seem to serve in the framework of deterministic reaction kinetics as well. In the deterministic framework, ordinary differential equations (ODEs) for non-delay reactions and delayed differential equations (DDEs) for delayed reactions can be easily written down based on the mass-action law. However, for those reactions consisting of Pro, the deterministic kinetics will give an incorrect description. This problem arises because the deterministic equations can provide a good approximation to the average kinetics of a stochastic description only if the number of reacting molecules of each type is large enough (Gillespie, 2001). This requirement can never be met when we study genetic systems, especially single-gene systems, where the number of each Pro is either one or zero at any given time. Even in real gene networks, it is unusual to have more than two copies of the same gene (Alberts et al., 2002). Accordingly, the above transcription reactions involving a Pro actually are not workable in the corresponding deterministic kinetics studies. Some deterministic models (also called mean-field models) of genetic regulatory networks employ mathematical functions obtained from biochemical reaction equations under some assumptions to model the regulations among genes without explicitly considering single gene expression. These mean-field models appear to be helpful in understanding some simple genetic regulatory networks, such as the genetic toggle switch (Gardner et al., 2000) and the genetic oscillator (Elowitz and Leibler, 2000). We note that these models are easily converted to virtual chemical reactions, which can be studied in the same stochastic kinetics framework mentioned above. We next introduce the virtual chemical reactions and use them to construct stochastic models of genetic regulatory networks.

2.2. Function-based virtual chemical reactions

The second type of gene network models is based on virtual chemical reactions. We illustrate this modeling method with a simple example, a two-gene toggle switch system. The toggle switch consists of two repressors and two genes. Each gene is inhibited by the repressor transcribed by the opposing gene. A synthetic genetic toggle switch in *E. coli* has been constructed and shows robust flipping between two stable states in certain conditions (Gardner et al., 2000). The following ODE model was used to predict these conditions, reasonably explaining the experimental observations:

$$\frac{dP_1}{dt} = \frac{a_1}{1 + P_2^{b_1}} - P_1, \tag{18}$$

$$\frac{dP_2}{dt} = \frac{a_2}{1 + P_1^{b_2}} - P_2, \tag{19}$$

where P_1 and P_2 are the concentrations of the two repressors, a_1 and a_2 are their synthesis rate constants, and b_1 and b_2 are the corresponding cooperativities of repression. The second terms in Eqs. (18) and (19) are degradation of the repressors. In this mean-field model, the Hill function, $a/(1 + P^b)$, is employed to model the repression interaction between the two genes. The larger the value of b, the higher is the cooperativity. b can also be a self-repression cooperativity in a single gene expression system where the Hill function serves as a self-inhibition mechanism (Monk, 2003).

We now transform Eqs. (18) and (19) into a set of chemical reactions. It is reasonable to convert the degradation terms to unimolecular decay reactions like reaction (4). Yet, one must be careful when doing the conversion for the inhibition parts due to the existence of the Hill functions. A simple, direct method is to convert them to zeroth-order *virtual* reactions with time-dependent probability rate constants obeying the Hill functions. We call them virtual reactions because they do not explicitly correspond to any real biochemical process. We can now write down the following reactions for the above mean-field model:

$$\xrightarrow{\kappa_1(t)} P_1(t), \quad \kappa_1(t) = \frac{a_1}{1 + P_2(t)^{b_1}}, \tag{20}$$

$$P_1 \xrightarrow{\kappa_2} \text{decay}, \quad \kappa_2 = 1, \tag{21}$$

$$\xrightarrow{\kappa_3(t)} P_2(t), \quad \kappa_3(t) = \frac{a_2}{1 + P_1(t)^{b_2}}, \tag{22}$$

$$P_2 \xrightarrow{\kappa_4} \text{decay}, \quad \kappa_4 = 1. \tag{23}$$

These reactions can be reconverted to the mean-field model in terms of the mass-action law. With delays included, virtual reactions (20) and (22) are rewritten as

$$\xrightarrow{\kappa_1(t)} P_1(t+\tau_1), \quad \kappa_1(t)=\frac{a_1}{1+P_2(t)^{b_1}}, \tag{24}$$

$$\xrightarrow{\kappa_3(t)} P_2(t+\tau_2), \quad \kappa_3(t)=\frac{a_2}{1+P_1(t)^{b_2}}. \tag{25}$$

While the above virtual reactions correspond to no real biochemical processes, they seem dynamically reasonable since the inhibition relation between the two genes is embedded in the varying probability rate constants. Of course, the extent to which these virtual reactions are reasonable should be justified by comparing the stochastic kinetics of this model with that of the corresponding process-based model. We will do the comparison in the next part of this paper.

Compared with the effective reactions, the virtual reactions have two main different features. First, an effective reaction is chemically designed to model a certain biochemical process like transcription, while a virtual reaction is derived from a mathematical function, modeling a certain biological function like inhibition. Thus, the latter is at a higher modeling level than the former. This indicates that fewer virtual reactions than effective reactions are used to model the same genetic networks. Second, the probability rate constant of an effective reaction is kept constant, while that of a virtual reaction is time dependent. The time-dependent term is in fact a correction factor to the fixed constant for the virtual reaction. However, the purposes of using effective and virtual reactions are the same: to allow the use of simple reactions to reasonably model complex biochemical processes. Furthermore, our effective and virtual reactions include delays. This feature is absent in regular reactions where the outputs are obtained at the same time as inputs are consumed. Since the Gillespie SSA was originally designed to handle regular reactions, we briefly describe a general delay SSA below that we use to simulate the delayed models. This algorithm was recently proposed by Roussel and Zhu (2006).

2.3. The delay SSA

Compared with the non-delay SSAs, the main difference is that we need to create a waiting list to store the delayed output events.

SSA with time delays:

1. Initialize: set initial numbers of molecules; set $t \leftarrow 0$; form a group of input events and a separate group of output events from the list of reactions; create an empty waiting list L for delayed output events.
2. Do an SSA step for the input events to get the next reacting event R_1 and the corresponding occurrence time t_1.
3. Compare t_1 with the least time, τ_{min}, in L.

→ If $t_1 < \tau_{min}$, generate the delay τ for R_1 (Note that there may be several delayed output events for one input event.) Then,

a. set $t \leftarrow t + t_1$;
b. update the number of molecules by performing R_1;
c. if $\tau = 0$, do G_1;
e. decrement all the delays in L by t_1;
d. if $\tau \neq 0$, add $\{G_1\ \tau\}$ into L.

→ If $t_1 > \tau_{min}$,

a. set $t \leftarrow t + \tau_{\min}$;
b. update the number of molecules by performing the output event G_{min}, which is associated with τ_{min};
c. delete $[G_{\min}\ \tau_{\min}]$ from L;
d. decrement all the delays in L by $\tau_{\min}$.

4. Go to step 2.

Note that the SSA step in step 2, i.e. the selection of the next reacting event and occurrence time, can be carried out by any exact version of the SSA (Gillespie, 1976, 1977; Gibson and Bruck, 2000). Naturally the delay corresponding to one delayed output event of a reaction is a random variable, fluctuating based on some probability distribution, with a mean value that can be tuned from experimental data. Here, we use fixed delays for simplicity, unless otherwise stated. In addition, the SSA steps for the case with varying probability rate constants are simply carried out by this means: at each time step, update the probability rate constants in terms of the corresponding mathematical functions.

In the next part, we first use this delay SSA to study the gene expression models presented above, and then to study two simple genetic networks. All stochastic simulations in this work are calculated using MATLAB (2005).

3. Results and discussion

We focus here on modeling prokaryotic genetic systems, especially *E. coli*.

3.1. Single-gene expression simulations

Very recently, the real-time production of single protein molecules under the control of a repressed *lac* promoter in individual *E. coli* cells was directly monitored through an epifluorescence microscope (Yu et al., 2006). It was found that the protein molecules are produced in bursts, with the distribution of the bursts per cell cycle fitting well a Poisson distribution, and that protein numbers in the bursts follow a geometric distribution. The model of gene expression proposed by Roussel and Zhu (2006), which is the same as ours (reactions (1)–(3)) except that it involves more transcriptional details, has been demonstrated to match well the observed stochastic kinetics. We show below that the reaction information in our simpler model is enough to reproduce the stochastic kinetics. To model this repressed condition, we simply use the following two reactions:

$$\mathrm{Pro} + \tilde{P} \xrightarrow{k_4} \mathrm{Pro}\tilde{P}, \tag{26}$$

$$\mathrm{Pro}\tilde{P} \xrightarrow{k_5} \mathrm{Pro} + \tilde{P}, \tag{27}$$

where $\tilde{P}$ is the repressor. Once the repressed state of the promoter, $\mathrm{Pro}\tilde{P}$, forms through reaction (26), transcription stops until the normal state of the promoter recovers through reaction (27).

Now we can build a model with reactions (1)–(3), (26) and (27) to model the gene expression in the repressed condition. We denote this by "Model 1". The parameter values and initial conditions are given in Table 1. Some of the parameters and initial conditions estimated from experimental data are kept fixed; others are tuned to match the average kinetics of the experimental system following the methods below. In the experiments, the average number of gene expression bursts per cell cycle is 1.2. The bursts last from 3 to 15 min, which are much

Table 1
Parameters and initial conditions used in the gene expression models

Parameter		Initial conditions
Model 1[*]		
$\mathrm{RNAP}(t) + \mathrm{Pro}(t) \xrightarrow{k_1} \mathrm{Pro}(t+\tau_1) + \mathrm{RBS}(t+\tau_1) + \mathrm{RNAP}(t+\tau_2) + R(t+\tau_2)$ (1)		
$\mathrm{Ribosome}(t) + \mathrm{RBS}(t) \xrightarrow{k_2} \mathrm{RBS}(t+\tau_3) + \mathrm{Ribosome}(t+\tau_4) + P(t+\tau_5)$ (2)		
$\mathrm{RBS} \xrightarrow{k_3} \mathrm{decay}$ (3)		
$\mathrm{Pro} + \tilde{P} \xrightarrow{k_4} \mathrm{Pro}\tilde{P}$ (26)		
$\mathrm{Pro}\tilde{P} \xrightarrow{k_5} \mathrm{Pro} + \tilde{P}$ (27)		
k_1	$0.01\,\mathrm{s}^{-1}$	RNAP = 40, Pro = 1, $R = 0$, Ribosome = 100, RBS = 0, P = 0, $\mathrm{Pro}\tilde{P} = 0$, $\tilde{P} = 100$
k_2	$0.00042\,\mathrm{s}^{-1}$	
k_3	$0.01\,\mathrm{s}^{-1}$	
k_4	$1\,\mathrm{s}^{-1}$	
k_5	$0.1\,\mathrm{s}^{-1}$	*Note*: We renumber the delays in this model and also in the following models. Some delays in different models have the same names but may have different meanings. All the delays are fixed except that the delay for protein production is set to $\tau_5 = 420 \pm 140$ s with a normal distribution
τ_1	40 s	
τ_2	90 s	
τ_3	2 s	
τ_4	58 s	
τ_5	420 ± 140 s	
Model 2 is Model 1 excluding reactions (26) and (27) *Note*: A fixed τ_5 of 420 s is used		
Model 3		
$\xrightarrow{k_1} \mathrm{RBS}$ (8)		
$\mathrm{RBS}(t) \xrightarrow{k_2} nP(t+\tau_1)$ (11)		
$\mathrm{RBS} \xrightarrow{k_3} \mathrm{decay}$ (3)		
k_1	$0.023\,\mathrm{s}^{-1}$	RBS = 0, $P = 0$.
k_2	$0.042\,\mathrm{s}^{-1}$	
k_3	$0.01\,\mathrm{s}^{-1}$	*Note*: The transcript R is not explicitly shown in the model. The formation of an RBS indicates the production of an R
n	5	
τ_1	500 s	
Model 4 is Model 3 except that $n = 1$ and $\tau_1 = 0$		
Model 5		
$\mathrm{RNAP}(t) + \mathrm{Pro}(t) \xrightarrow{k_1} \mathrm{Pro}(t+\tau_1) + \mathrm{RNAP}(t+\tau_2) + R(t+\tau_2) + nP(t+\tau_3)$ (13)		
k_1	$0.01\,\mathrm{s}^{-1}$	RNAP = 40, Pro = 1, $R = 0$, $P = 0$
n	4	
τ_1	40 s	
τ_2	50 s	
τ_3	500 s	

Table 1 (*continued*)

Parameter		Initial conditions	
Model 6			
$\xrightarrow{k_1} nP(t+\tau_3)$			(14)
k_1	$0.023\,s^{-1}$	$P=0$	
n	4		
τ_3	500 s		

*The settings of parameters and initial conditions for Model 1 are described as follows:
We study single gene expression, so the number of Pro is initially set to 1. The initial number of RNAPs is set to 40 so that the average number of available RNAPs in the simulations is the order of magnitude observed in experiments (Record et al., 1996). The number of repressors is set larger than the number of RNAPs, considering the inhibition is very strong in the studied system. The value of k_1 is set to $0.01\,s^{-1}$ so that the average time it takes for an RNAP to bind the promoter is $1/(k_1 \times 40\,s) = 2.5\,s$, which is reasonably less than the reported transcription initiation interval of 3.3 s (Kennell and Riezman, 1977). Note that the transcription initiation process includes the transition from the closed promoter complex to the open promoter complex (in Model 1, τ_1 accounts for this process), which is usually slower than the binding event (McClure, 1980). Thus, we set the value of τ_1 to 40 s, which is among the range from 10 s to several min (McClure 1980).
A similar model (Kierzek et al., 2001) used the average number of free ribosomes of 350 to produce the protein synthesis rate of $26\,s^{-1}$. An initial number of ribosomes of 100 in Model 1 reasonably produces the low protein synthesis rate of $11\,s^{-1}$. The time of the RBS clearance in translation initiation, τ_3, is set to 2 s, which is close to the experimentally estimated rate of RBS clearance (Draper, 1996).
The length of the gene studied in *E. coli* is ~2500 nt (Yu et al., 2006). The average rate of transcriptional elongation in *E. coli* is ~50 nt/s. We get $\tau_2 = \tau_1 + 2500\,nt/(50\,nt/s) = 90\,s$. The average translation rate is ~15 amino acids/s. So, $\tau_4 = \tau_3 + 2500\,nt/(45\,nt/s) = 58\,s$. The other parameters, k_2–k_5, are tuned to match the average kinetics of the experimental system. See the text for details.

smaller than the cell cycle (about 55 min). That means most of the bursts are finished in one cell cycle. Thus, it is reasonable to do the stochastic simulations without necessarily considering the cell division. We do each stochastic simulation for 50 min, corresponding to a cell cycle, and then count the promoter binding occurrences (transcription initiation) by reaction (1) for each simulation. According to the experimental study, the promoter binding frequency should be close to the burst frequency since each burst arises from one transcript. Accordingly, we tune k_4 for reaction (26) and k_5 for reaction (27) to attain the average promoter binding frequency close to $1.2\,s^{-1}$. For the translational part, the average number of protein molecules per burst is 4.2 in the experiments. In the simulations, we count the events of ribosome binding to RBS in reaction (2) (translation initiation) between the generation and decay of an RBS. We match the average of this binding frequency close to $4.2\,s^{-1}$ by tuning k_2 for reaction (2) and k_3 for reaction (3).

Another important experimental observation is that the bursts display particular temporal spreads. Yu et al. (2006) attributed these spreads to the long post-translational protein assembly process which takes about 7.0 ± 2.5 min. Rather, Roussel and Zhu have showed that the statistical distribution of the protein production delay causes the wide temporal spreads (unpublished results). Following this line, we now separate in reaction (2) the ribosome release delay from the protein production delay, and name the latter τ_5. We set $\tau_5 = 420 \pm 140$ s with a normal distribution. Note that we choose the normal distribution since we do not know exactly what the distribution is. The other delays are kept fixed.

We show three example time traces of gene expression in Fig. 1 and statistical analyses of simulated gene expression in Fig. 2. In good agreement with the experimental observations, the protein molecules are produced in bursts with particular temporal spreads, and the transcription of one transcript is responsible for one burst. Besides the common gene expression bursts shown in Fig. 1, we also observe cases where more than one transcript is responsible for a burst (not shown) and that no protein is produced for a transcription initiation (for example, the transcription initiation that happened between the first and second bursts in the lower panel of Fig. 1). However, these events happen with low probabilities under the simulating conditions. In addition, just as obtained in the experiments, the distribution of transcription initiations per 50 min fits well a Poisson distribution, and the number of translation initiations for each transcription follows a geometric distribution as well. The above simulations reproduce the key features of the stochastic kinetics of the experimental system. It also indicates that Roussel et al.'s model can be further simplified to the current version without losing stochastic kinetics of interest.

We now remove the repressed reactions, i.e. reactions (26) and (27), from Model 1 and build a pure single-gene expression model, called Model 2. Using the same parameters and initial conditions as used in Model 1, we get an unrepressed gene expression kinetics characterized by the transcript and protein productions. We find that in the unrepressed condition the gene expression kinetics is not sensitive to the statistical distribution of protein production delay. Thus, all the delays are set fixed in the following simulations. We evaluate below the simplified gene expression models presented above, by comparing their kinetics with that of Model 2.

First, Model 2 is simplified to Model 3 which includes reactions (8), (11), and (3) (see Table 1). k_1 and k_2 for

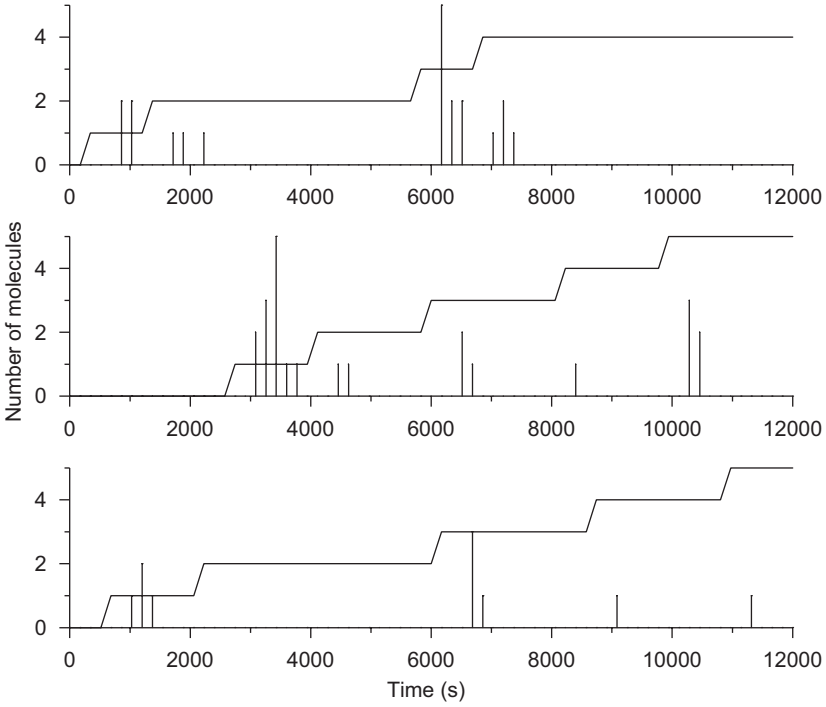

Fig. 1. Three examples of protein burst production for the gene expression under the repressed condition (Model 1) for 200 min (about 4 cell cycles). The vertical axis is the number of protein molecules newly produced during the last 3 min. The step lines indicate the total numbers of transcriptional initiations accumulated over time. One step denotes one transcriptional initiation. See Table 1 for the parameter values used.

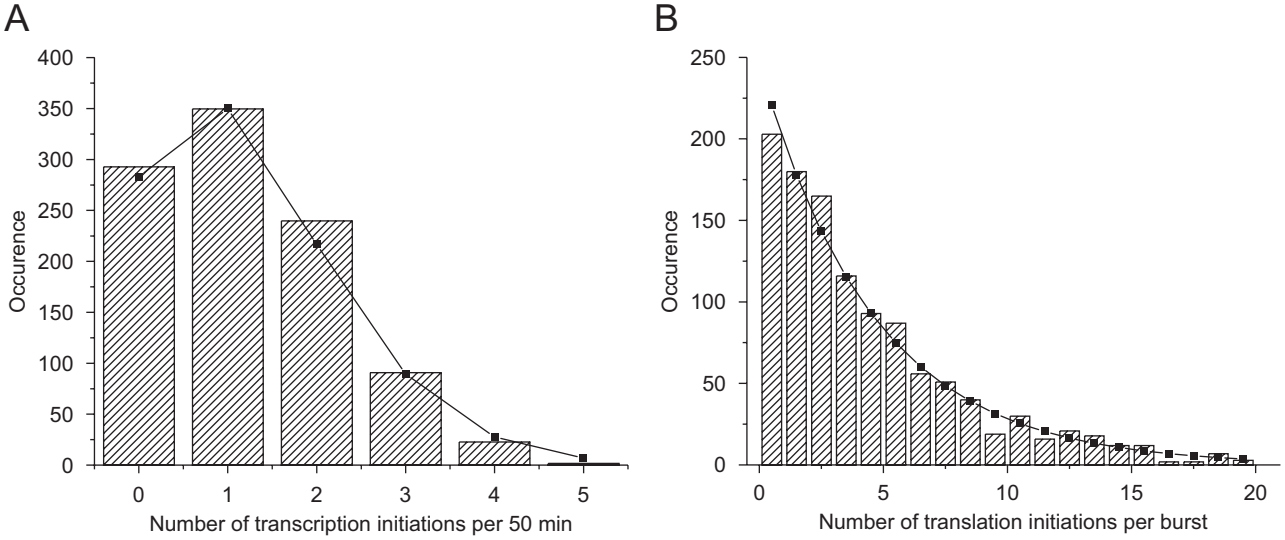

Fig. 2. Statistics of the protein bursts for the gene expression shown in Fig. 1. (A) Distribution of the number of transcription initiations per 50 min from 1000 simulations. It fits a Poisson distribution (solid line) with an average of 1.2. (B) Distribution of the number of translation initiations between the birth and death of an RBS from 1148 RBSs. It follows a geometric distribution (solid line) with an average of 4.1.

reactions (8) and (11) in Model 3 are different from those for reactions (1) and (2) in Model 2, but they are easily tuned to allow Model 3 to have close transcription and translation initiation frequencies to Model 2. k_3 is kept the same in both models. To match the average kinetics of protein production, the burst size and the protein production delay in Model 3 are carefully tuned. As shown in Fig. 3, the average gene expression kinetics (both protein and transcript) of Models 2 and 3 match very well, and the standard deviations of expressed proteins also have a reasonable agreement, but the standard deviations of transcripts have large deviations. This indicates that in this case the simplified Model 3 correctly describes protein production, but does not give a good description for transcript production. Model 4 is a special case of Model 3, where the burst size and the protein production delay are

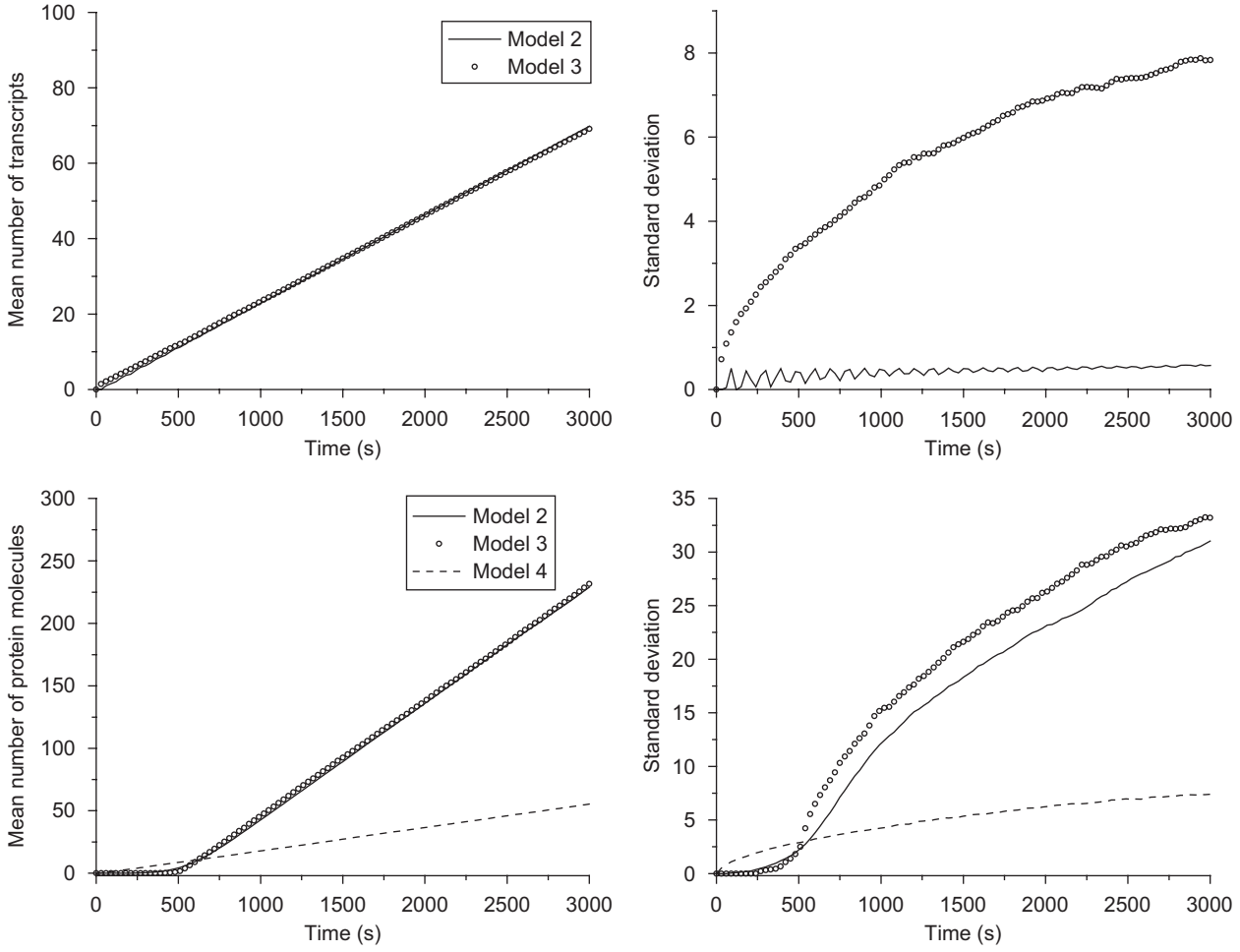

Fig. 3. Comparison of gene expression in Models 2–4. Model 2 is Model 1 used in Fig. 1 without the repressed reactions. Model 3 is a simplified version of Model 2. Model 4 is a special case of Model 3. The averages and standard deviations of transcripts and proteins versus time are presented for Models 2 and 3. For Model 4, only the average and standard deviation of proteins are given; its transcript production is similar to that of Model 3. See Table 1 for the details of the models and the parameter values used in each model.

simply set to 1 and 0, respectively. The kinetics of protein for Model 4 is also given in Fig. 3 as a comparison. Neither the average nor the standard deviation looks reasonable. We can predict that the deviations in the kinetics between Models 4 and 2 would get larger as the protein synthesis rate of Model 2 increases. Note that the average rate of protein production of Model 2 is about 11 protein molecules per s, which is low compared with the value of 20 protein molecules per s for regular gene expression in *E. coli* (Kennell and Riezman, 1977). We present the kinetics of Model 4 since this model is a commonly-used model of gene expression in the current literature. This comparison implies that Model 4 may cause significant errors in quantitative studies.

Second, we evaluate another type of gene expression model proposed above, i.e. one single reaction of gene expression. Model 5 consists of reaction (13) only. Compared with Model 2, Model 5 keeps k_1 and all the delays for the transcriptional part. Only two parameters, the burst size and the protein production delay, need to be tuned to match the average kinetics of Model 2. Note that this protein production delay is involved in the whole gene expression process, which is longer than the previously mentioned protein production delay involved only in the translational process. The comparison results are given in Fig. 4. Since Model 5 maintains the transcriptional part of Model 2, its kinetics of transcript matches well with that of Model 2. However, their standard deviations of proteins are very different even though their averages match well. It indicates that Model 5 is good for transcription modeling, but not valid for translation modeling in this case. Finally, Model 5 is simplified to Model 6, which contains protein kinetics only. This is the simplest model of gene expression presented in this work. Interestingly, it describes well the protein kinetics.

From the above comparisons, we conclude that Models 3 and 6 both are good in protein production modeling, Model 5 is good in transcription modeling, and Model 4 is the worst one among the simplified models. Model 3 may be better than Model 6 since the former maintains the average kinetics of transcripts even though the stochastic kinetics of transcripts is not well described. Model 5

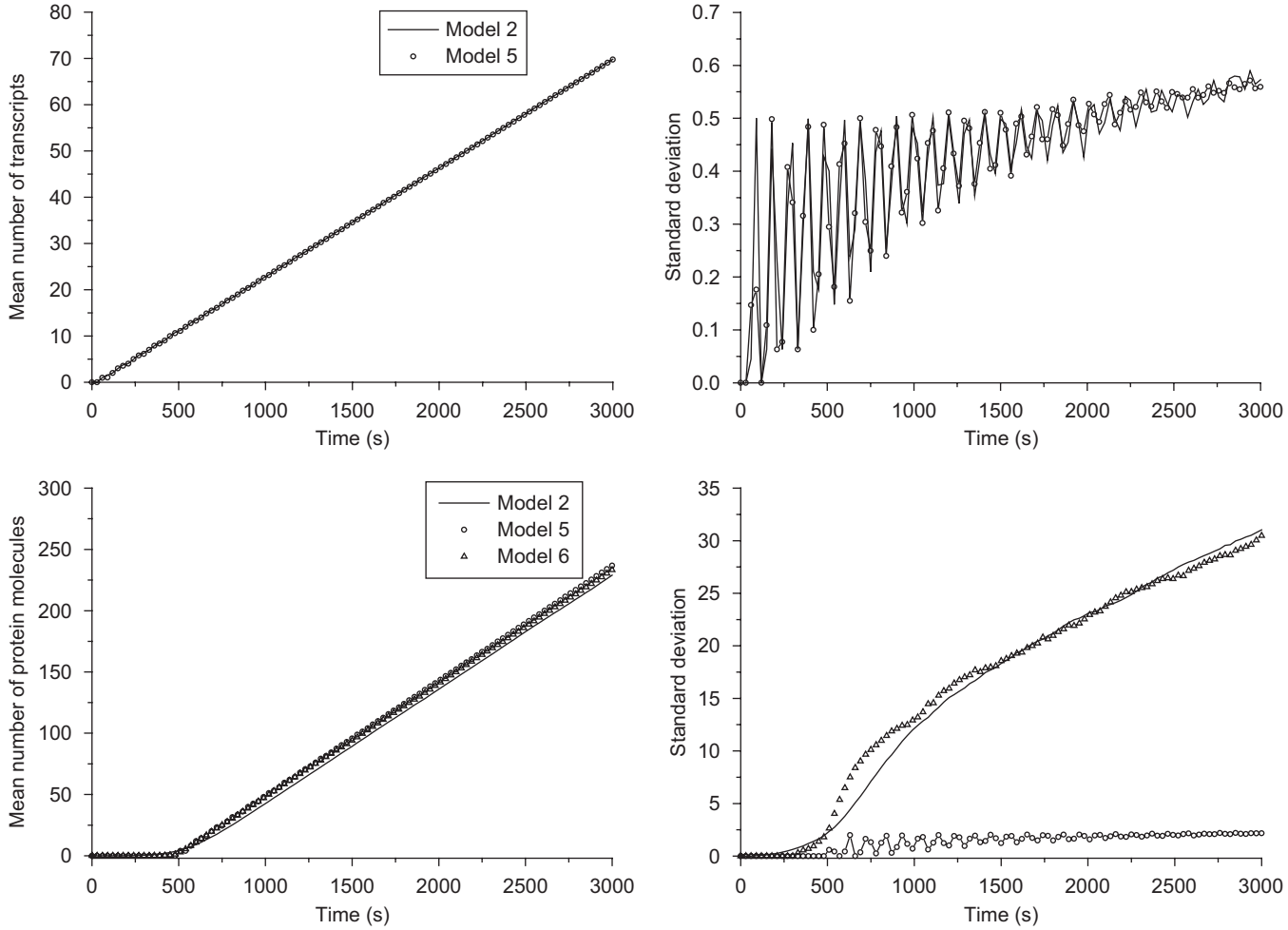

Fig. 4. Comparison of gene expression in Models 2, 5 and 6. Model 5 is a single-reaction gene expression model. Model 6 is a simplified version of Model 5. The averages and standard deviations of transcripts and proteins against time are presented for Models 2 and 5. For Model 6, only the average and standard deviation of proteins are given since it does not concern transcripts. See Table 1 for the details of the models and the parameter values used in each model.

captures well the transcriptional kinetics, but the fluctuations of proteins are underestimated in the studied case. We find that the fluctuations of proteins grow with the protein burst size, so the protein production modeling of Model 5 can be improved for cases with high protein production rate. In summary, all these simplified models could be used to model single gene expression, causing some errors compared with Model 2. Choosing a suitable model depends on what the study is focused on. One should keep in mind the features of these models when using them.

We intend to use Models 2–6 to construct genetic regulatory networks. To this end, we also need to include additional reactions for transcriptional and translational regulations. Since the Pro and the RBS are two important kinetics-controlling species in transcription and translation, respectively, the models including Pro (Models 2 and 5) could be directly used for transcriptional regulations and the models including RBS (Models 3 and 4) could be directly used for translational regulations (not studied in this work). As for Model 6 with neither Pro nor RBS, it could be used to model stable or basal gene expression in gene networks.

3.2. The genetic toggle switch

We are now ready to study the genetic toggle switch, a simple genetic network. The switch consists of two genes, where each gene can be inhibited by the repressor transcribed by the other gene. Based on Model 1, we can readily build a toggle switch model. We assume that RNAPs and ribosomes are both in a steady state for simplicity, so we get two first-order reactions (28) and (29) listed in Table 2 from reactions (1) and (2). To model the repression, we add, like in Model 1, the reversible inhibition reaction pair (32) and (33) for each gene. Also, the protein decay reactions (31) are included. Reactions (28)–(33) constitute the toggle switch model 1. A genetic toggle switch of this kind was experimentally synthesized in *E. coli* (Gardner et al., 2000). Two conditions were found from the mean-field model (Eqs. (18) and (19)) to achieve bistable switching: Both repressors must act cooperatively,

Table 2
Parameters and initial conditions used in the two symmetrical toggle switch models

Parameter		Initial conditions
Toggle switch model 1		
$\mathrm{Pro}_i(t) \xrightarrow{k_1} \mathrm{Pro}_i(t+\tau_1) + \mathrm{RBS}_i(t+\tau_1)$		(28)
$\mathrm{RBS}_i(t) \xrightarrow{k_2} \mathrm{RBS}_i(t+\tau_2) + P_i(t+\tau_3)$		(29)
$\mathrm{RBS}_i \xrightarrow{k_3}$ decay		(30)
$P_i \xrightarrow{k_4}$ decay		(31)
$\mathrm{Pro}_i + P_j \xrightarrow{k_5} \mathrm{Pro}_i P_j$		(32)
$\mathrm{Pro}_i P_j \xrightarrow{k_6} \mathrm{Pro}_i + P_j$		(33)
k_1	$0.4\,\mathrm{s}^{-1}$	$\mathrm{Pro}_i = 1$
k_2	$0.042\,\mathrm{s}^{-1}$	$\mathrm{RBS}_i = 0$
k_3	$0.01\,\mathrm{s}^{-1}$	$\mathrm{Pro}_i P_j = 0$, $P_i = 0$
k_4	$0.001, 0.01\,\mathrm{s}^{-1}$	$i, j = 1$ and 2; for Pro_i P_j, the values of i and j are different
k_5	$0, 0.0001, 0.01\,\mathrm{s}^{-1}$	
k_6	$0.01\,\mathrm{s}^{-1}$	
τ_1	1 s	*Note*: Some parameters have multiple values used in different cases shown in Fig. 5
τ_2	2 s	
τ_3	4, 420 s	
Toggle switch model 2		
$\xrightarrow{\kappa_1} P_1(t+\tau), \quad \kappa_1 = \frac{a}{1+P_2(t)^b}$		(24)
$P_1 \xrightarrow{\kappa_2}$ decay		(21)
$\xrightarrow{\kappa_3} P_2(t+\tau), \quad \kappa_3 = \frac{a}{1+P_1(t)^b}$		(25)
$P_2 \xrightarrow{\kappa_4}$ decay		(23)
a	6, 100	$P_1 = 0$, $P_2 = 0$
b	0.5, 1, 2	
τ	0.001, 9	*Note*: Some parameters have multiple values used in different cases shown in Fig. 6. Time is in arbitrary units (t) and probability rate constants are in units of t^{-1}
κ_2	1	
κ_4	1	

and the effective rate of synthesis of the two repressors must be balanced. Here, we study the toggle switch model 1 in a different way. We consider only the symmetrical case where both genes are identical. In this particular case, we would also ask: What are the conditions for bistability of the symmetrical toggle switch? Various toggle switch kinetic models were recently studied to explore this issue without considering any delays (Cherry and Adler, 2000; Warren and ten Wolde, 2004, 2005; Kaznessis, 2006; Lipshtat et al., 2006). In contrast to these studies, we mainly focus on effects of the delays present in our model.

There are three different delays involved in the toggle switch model 1: the Pro clearance/RBS production delay, the RBS clearance delay, and the protein production delay. To perform the following comparisons between short- and long-delayed cases, we start with small values for these delays. Specifically, they are set to 1, 2, and 4 s, respectively. With these delays, we first study the effect of inhibition strength by changing the promoter–repressor binding probability constant, k_5. As shown in Figs. 5A–C, we increase k_5 from 0 to $0.01\,\mathrm{s}^{-1}$, keeping the other parameters fixed. As the inhibition is weak, the switching between the two stable states happens frequently due to molecular fluctuations (the single run in Fig. 5B); as the inhibition gets strong, the switching appears to happen rarely (the single run in Fig. 5C). On the other hand, the standard deviation in Figs. 5A and B soon reaches an equilibrium state, while it takes a long transient for Fig. 5C to do so. The equilibrium standard deviation is increased rapidly, from 66 (Fig. 5A) to 139 (Fig. 5B) to 307 (Fig. 5C), indicating the increasingly stronger effect of repression between the two genes. Note that in Fig. 5C the single run deviates far from the mean plus standard deviation, also implying large fluctuations in this case. While the case in Fig. 5C does not often display switching spontaneously, we can force the system to change to a new state by temporarily turning off the repression function of the dominant gene and then, after turning on the gene, the system remains in the new state. See the inset of Fig. 5C for an example of such manual switching, showing the key feature of bistability. Moreover, for this case, as we increase the protein decay rate, the switching probability seems to increase again (Fig. 5D). Though the switching stochastic dynamics are not statistically studied here, we draw these results by carefully observing time patterns from many individual stochastic simulations.

What happens as the protein production delay grows? Fig. 5E shows a case which is the same as in Fig. 5D except that the protein production delay increases from 4 to 420 s. Figs. 5F and C are another pair of this comparison but with a lower protein decay rate. Both Figs. 5E and F show an interesting behavior: the two genes seem to be both on and off at some moments. For the low protein decay rate case, the long protein production delay leads to a long transient, as illustrated in Fig. 5F, before the system reaches a stable state. The long protein production delay does not affect the switching dynamics much but the transient dynamics in this case. For the high protein decay rate case, however, the long protein production delay totally changes the dynamics as shown in Fig. 5E, which is characterized by short-period, high- and low-amplitude peaks. The bistability in this case is almost destroyed by the long protein production delay. Another interesting phenomenon that can be clearly seen in Figs. 5E and F and also in Figs. 5B and D is that the two expressed proteins seem to oscillate. There are two modes of oscillations

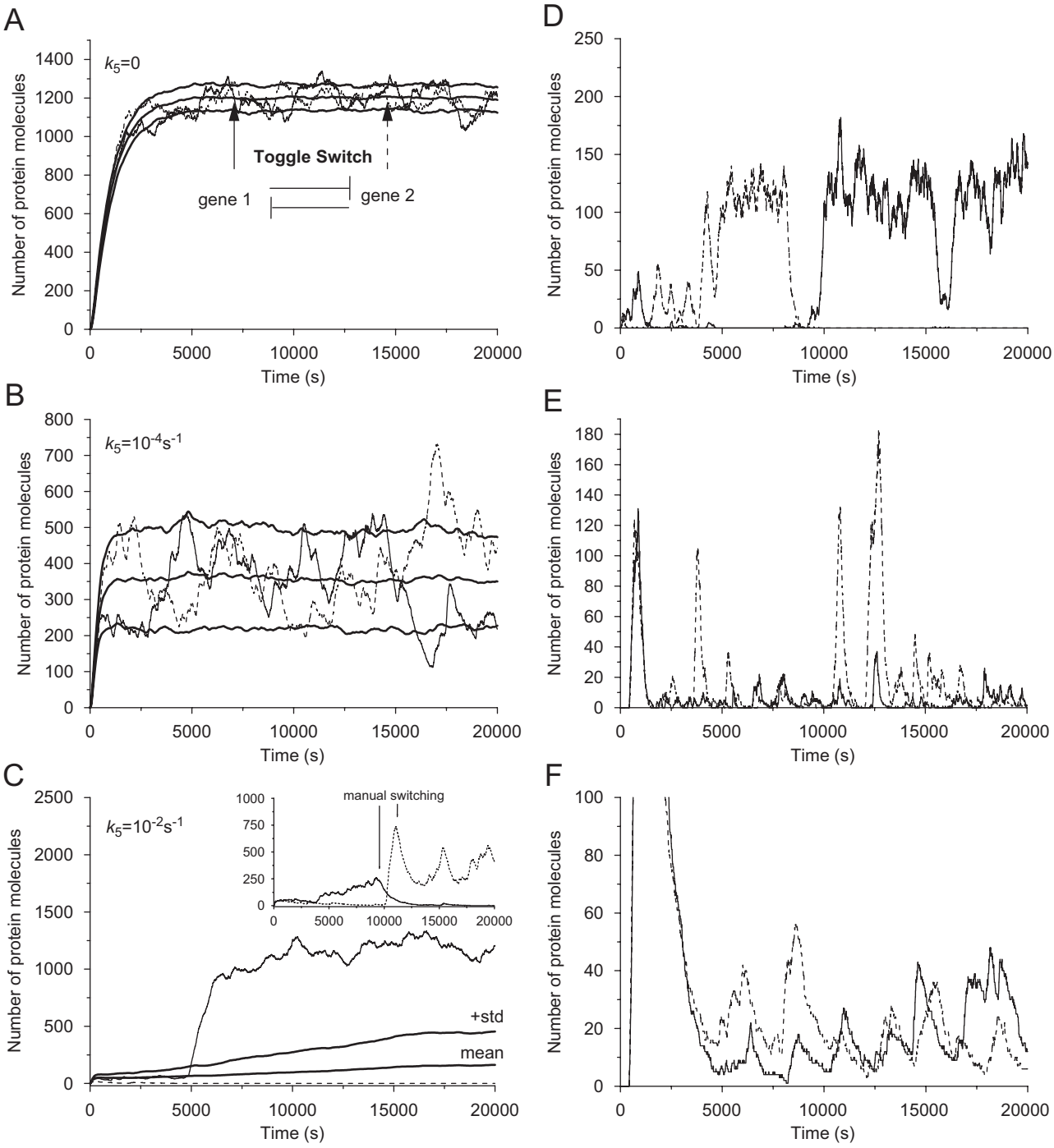

Fig. 5. Examples of protein time series simulated from the toggle switch model 1. (A)–(C) All the parameters are the same for each case except the repression binding constant k_5 which is specified on each panel; the protein decay rate constant $k_4 = 0.001\,\mathrm{s}^{-1}$. Bold lines are mean and $\pm$standard deviation (std) values of gene 1 over 100 runs except for (C) which was averaged over 500 runs. The corresponding values of gene 2 are the same since gene 2 is identical to gene 1. (A)–(D) The delay for protein production is set to a small value: $\tau_3 = 4\,\mathrm{s}$. (D) The same case as (C) except that $k_4 = 0.01\,\mathrm{s}^{-1}$. (E) The same case as (D) except that $\tau_3 = 420\,\mathrm{s}$. (F) The same case as (C) except that $\tau_3 = 420\,\mathrm{s}$. The inset in (C) shows an example of manual switching, where the first arrow indicates that the value of k_5 for gene 1 is changed from 0.01 to 0, and the second arrow that it is set back to 0.01. See Table 2 for the details of the model and the other parameter values used.

observed: antiphase oscillations and inphase oscillations (previously referred to as the both on and off state). These oscillations are originally caused by molecular fluctuations as they can also more or less be observed in the non-repression case shown in Fig. 5A. The two repression cases shown, respectively, in Figs. 5B and D indicate that the strong inhibition between the two genes assists the antiphase state to last longer; the fact that antiphase oscillations dominate in Figs. 5B and D and inphase oscillations dominate in Figs. 5E and F implies that a

long protein production delay makes the inphase state relatively stable.

As for the Pro clearance/RBS production delay, we find that a long delay would destroy the bistability as well. For example, as this delay is set to 40 s, the same value used in Model 1, the toggle switch does not show bistability (results not shown). Since this delay controls the transcription initiation frequency, increasing the delay leads to a decrease in the synthesis rate of proteins. Similar to the Pro clearance delay, the RBS clearance delay also affects the dynamics since it also controls the protein synthesis rate. However, it does not influence the dynamics as much as the Pro clearance delay does, because the number of available RBSs is larger than that of available Pro for the gene expression of each gene.

The above results indicate three conditions for bistability of the symmetrical toggle switch: (i) the inhibition must be strong, (ii) the synthesis rate of proteins must be high, and (iii) the protein production delay must be short. Specifically, weak inhibitions and high protein decay rates can cause frequent switching due to intrinsic molecular fluctuations. Note that one condition for bistability from the mean-field toggle switch model is that each repressor must act cooperatively, i.e. the cooperative binding of at least two proteins is required to get bistability. Since we do not include cooperative effects of multimers here, our first condition means that the bistability robustness of the system grows with the inhibition strength. This indicates that strong inhibition enables bistability of the toggle switch without cooperative binding, which agrees with a recent chemical master analysis of a similar toggle switch model (Lipshtat et al., 2006). The other condition from the mean-field model is that the effective rates of synthesis of the two repressors must be balanced. In our model, they are always balanced since the two genes are identical. Besides that, conditions (ii) and (iii) indicate that to build a robust toggle switch with fast switching time, the two genes must also have a high protein production rate and a fast response to repressors.

We next study a second toggle switch model which is converted from the mean-field model (Eqs. (18) and (19)) by using the second method proposed above. The model, called the toggle switch model 2, is given in Table 2. It is worth stressing that the first model is not quantitatively comparable with the second one since the latter loses a lot of reaction details compared with the former. We intend to qualitatively compare the dynamics of the two toggle switch models. Thus, the parameters for the second model are given a little arbitrarily, and the populations of protein molecules are not matched in the two models. Note that time in this model is in arbitrary units (t) and probability rate constants are in units of t^{-1}. In the toggle switch model 2, there exists only one delay, i.e. the protein production delay. We first set the protein production delay to 0.001 and increase only the cooperativity of repression from 0.5 to 2. Three cases are shown in Figs. 6A–C. As in Fig. 5C, an example of manual switching is illustrated in the inset of Fig. 6C. Interestingly, it seems that the cooperativity plays a similar role as the promoter–repressor binding constant does in the first model. In addition, as in Figs. 5A–C, the equilibrium standard deviation is increased rapidly, from 4.7 (Fig. 6A) to 18.5 (Fig. 6B) to 50.4 (Fig. 6C). However, the long transient feature of the first toggle-switch model is lost in Fig. 6C. We then *reduce the protein synthesis rate* from 100 to 6. We get similar patterns illustrated in Fig. 6D to those obtained by *increasing the protein decay rate* in the toggle switch model 1 (Fig. 5D). Note that although we used two different methods in the two models to change the dynamics of protein production, they caused similar effects.

As we did for the toggle switch model 1, we now increase the protein production delay for both the two cases (Figs. 6C and D). Fig. 6E shows a case which is the same as in Fig. 6D except that the protein production delay increases to 9. Figs. 6F and C are another pair of this comparison but with a higher protein synthesis rate. We obtain the similar dynamic behavior in Figs. 6E and F as we got in Figs. 5E and F: the two genes seem to oscillate inphase. For the high protein synthesis case, the long protein production delays do not affect much the switching dynamics but affect significantly the transient dynamics. It should be mentioned that this effect of time delay is quite similar to that obtained from a delayed mean-field model of gene network in multicells (Veflingstad et al., 2005), which can also produce the homogenous oscillations as shown in Fig. 6F before the system reaches the steady state. For the low protein synthesis case, however, the bistability is almost destroyed by the long protein production delay (Fig. 6E).

In conclusion, the two toggle switch models share the following key dynamic characteristics: strong interactions between the two genes assist to stabilize the antiphase state, while long protein production delays assist to stabilize the inphase state. Both the antiphase and inphase states are induced by molecular fluctuations. However, there are also some dynamic distinctions between them. For example, compared with the first model, the second model has short switching times, and its inphase state is relatively regular. These results indicate that it is reasonable to use function-based virtual reactions to study genetic networks even though they lose much reaction detail.

3.3. The repressilator

The repressilator consists of three genes. The protein from the first gene inhibits the second gene, whose protein product in turn inhibits the third gene, and finally the protein from the third gene inhibits the first gene, completing the cycle. Each gene transcribes the repressor of one of the other genes. A mean-field model showed that this configuration can produce oscillating levels of each repressor (Elowitz and Leibler, 2000). In the same paper, following the modeling work, Elowitz and Leibler reported a repressilator, which was constructed in *E. coli*, to show

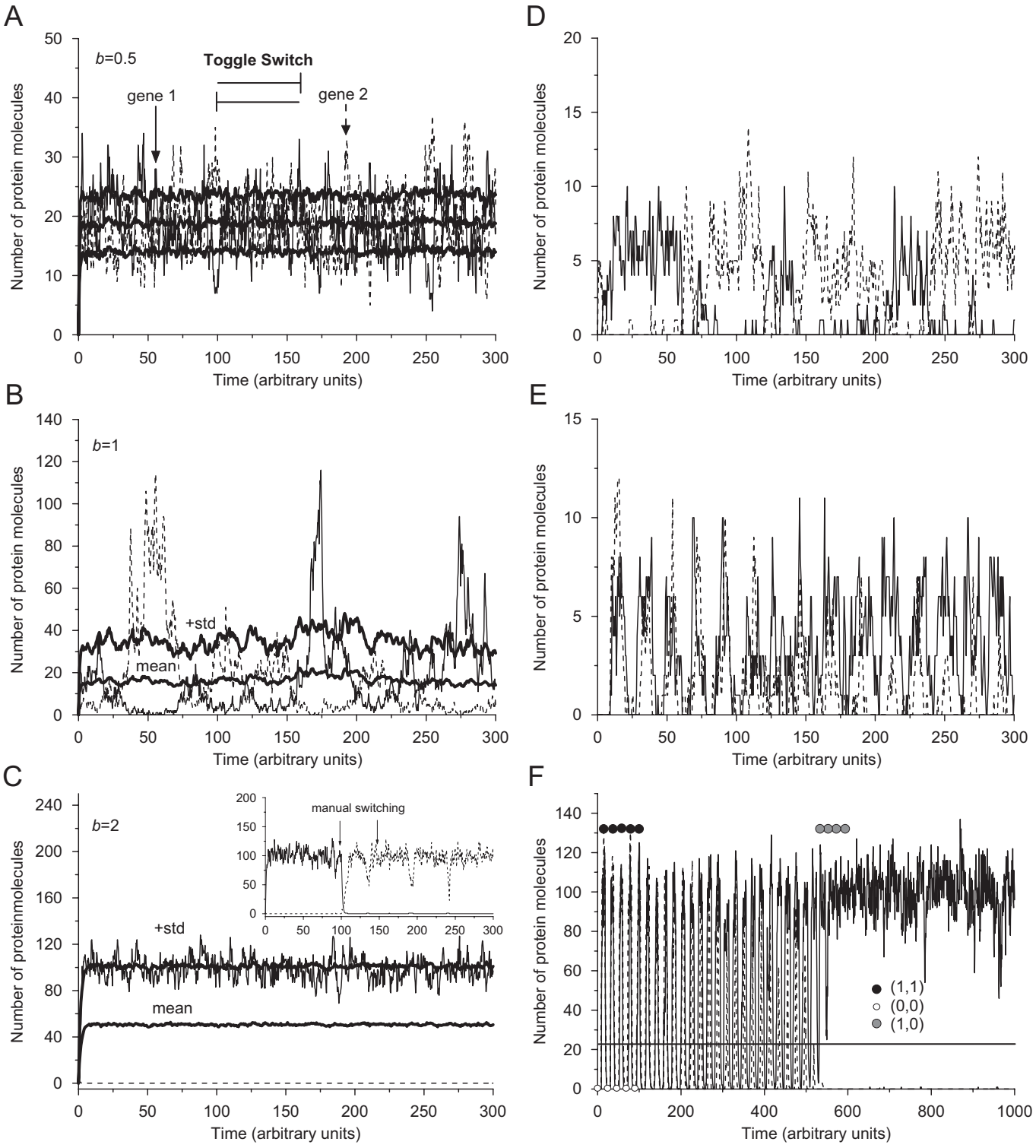

Fig. 6. Examples of protein time series simulated from the toggle switch model 2. (A)–(C) All the parameters are the same for each case except the cooperativity of repression, b, which is specified on each panel; the protein synthesis rate $a = 100$. Bold lines are mean and $\pm$standard deviation values of gene 1 over 100 runs. (A)–(D) The delay for protein production is set to a small value: $\tau = 0.001$. (D) The same case as (C) except that $a = 6$. (E) The same case as (D) except that $\tau = 9$. (F) The same case as (C) except that $\tau = 9$. The inset in (C) gives an example of manual switching, where the first arrow indicates that the value of b for gene 1 is changed from 2 to 0.5, and the second arrow that it is set back to 2. The system state represented by the binary code is also illustrated in (F). The straight line denotes a threshold, below which, state "0" is assigned and above which, state "1" is assigned. The circle symbols denote binary states at given time points. See Table 2 for the details of the model and the other parameter values used.

Table 3
Parameters and initial conditions used in the symmetrical repressilator model

Parameter		Initial conditions
The repressilator model		
$\xrightarrow{\kappa_1} P_1(t+\tau)$,		$\kappa_1 = \dfrac{a}{1+P_3(t)^b}$ (34)
$P_1 \xrightarrow{\kappa_2}$ decay		(35)
$\xrightarrow{\kappa_3} P_2(t+\tau)$,		$\kappa_3 = \dfrac{a}{1+P_1(t)^b}$ (36)
$P_2 \xrightarrow{\kappa_4}$ decay		(37)
$\xrightarrow{\kappa_5} P_3(t+\tau)$,		$\kappa_5 = \dfrac{a}{1+P_2(t)^b}$ (38)
$P_3 \xrightarrow{\kappa_6}$ decay		(39)
a	100	$P_1 = 0$, $P_2 = 0$, $P_3 = 0$
b	2	
τ	0.001, 1, 3, 6	*Note*: The delay τ has multiple
κ_2	1	values used in different cases
κ_4	1	shown in Fig. 7. Time is in
κ_6	1	arbitrary units (t) and probability rate constants are in units of t^{-1}

oscillations. To qualitatively study this genetic oscillator, we can use the Hill function to construct a repressilator model the same way as we constructed the toggle switch model 2. The model is given in Table 3. For simplicity, we still consider only the symmetrical case where all the three genes are identical. A detailed repressilator model, constituting transcriptional and translational elongation, was intensively studied, but omitting exploring the effect of the time delay for protein production (Tuttle et al., 2005). We focus on this effect by simply varying the protein production delay in our model.

As the protein production delay is as low as 0.001, the system shows random oscillations with a characteristic frequency about 0.28, as illustrated in Fig. 7A. As the delay increases to 1 (Fig. 7B), the oscillations become regular, and the characteristic frequency drops to 0.067. Increasing the delay to 3 (Fig. 7C) shows there exists two different rhythms of oscillations: high-frequency, small-amplitude oscillations as time <40 and low-frequency, big-amplitude oscillations as time >40. The former oscillations are a transient, featured with the "all-on-and-all-off" behavior of the three genes (the inphase state). The latter are robust oscillations with the characteristic frequency of 0.037. This two-rhythm dynamical behavior appears more clearly as the delay grows to 6 (Fig. 7D). Note that the interesting transients are obtained by using the initial conditions where none of the three repressors exist. We thus summarize the effect of the protein production delay in the repressilator model as follows: if the delay is small, only irregular oscillations with one characteristic frequency are shown; as the delay grows to a certain value, two types of oscillations appear. One type is of transient dynamics, and the other is of robust dynamics. Transient oscillations last longer for longer delays. The characteristic frequency of robust oscillations decreases with the increase of the delay.

Since this stochastic repressilator model is converted from a mean-field model, the results obtained above could be understood by analyzing the corresponding delayed mean-field model. The bifurcation diagram of the mean-field model (not given) shows two bifurcations when the delay is used as the control parameter and the other parameters are fixed to the values given in Table 3. One bifurcation happens as the delay is between 0 and 0.2 and the other between 1 and 2. The oscillations born from the first bifurcation are stable, and the oscillations from the second one are unstable. This explains the two rhythms shown in Figs. 7C and D. The mean-field model shows only one stable steady state before the system reaches the first bifurcation. However, in this case the stochastic model shows irregular oscillations (Fig. 7A). This is because the system is very close to the first bifurcation so that molecular noise randomly causes the oscillations (Hou and Xin, 2003; Li and Zhu, 2004). Another point worth noting is that the mean-field model does not show oscillations at all as the delay is zero, but a small protein production delay will trigger big oscillations (the first bifurcation). We note that the mean-field model used to explain the oscillations for the synthetic repressilator does not include any delay (Elowitz and Leibler, 2000), but it does include the translation process, which would take some time. Our mean-field model does not explicitly include translation, only considering the time it takes for a protein to mature. This strongly implies that the protein production time is a significant factor in controlling the oscillating dynamics of the repressilator, which is in agreement with the stability analysis of a delayed version of the Elowitz and Leibler mean field (Chen and Aihara, 2002).

3.4. Connection with Boolean network models

Some time series of proteins shown above for the two genetic networks display clear expressing (on) and non-expressing (off) states of genes. For example, we can extract, by manually choosing a suitable threshold, the following discrete temporal binary code (gene1, gene2) of the toggle switch model 2 from the time series shown in Fig. 6F: (0, 0), (1, 1), (0, 0), (1, 1)…, (1, 0), (1, 0),… . See the figure for details on how the binary code is obtained. Also, "0" can be defined such that the number of proteins reaches the bottom of a valley and, "1" that the number reaches the top of a peak. We can see that there are two different repeating patterns, {(0, 0), (1, 1)} and {(1, 0)}. In fact, there exists another equivalent pattern {(0, 1)} since the two genes are identical. It is interesting that the corresponding Boolean network model for the toggle

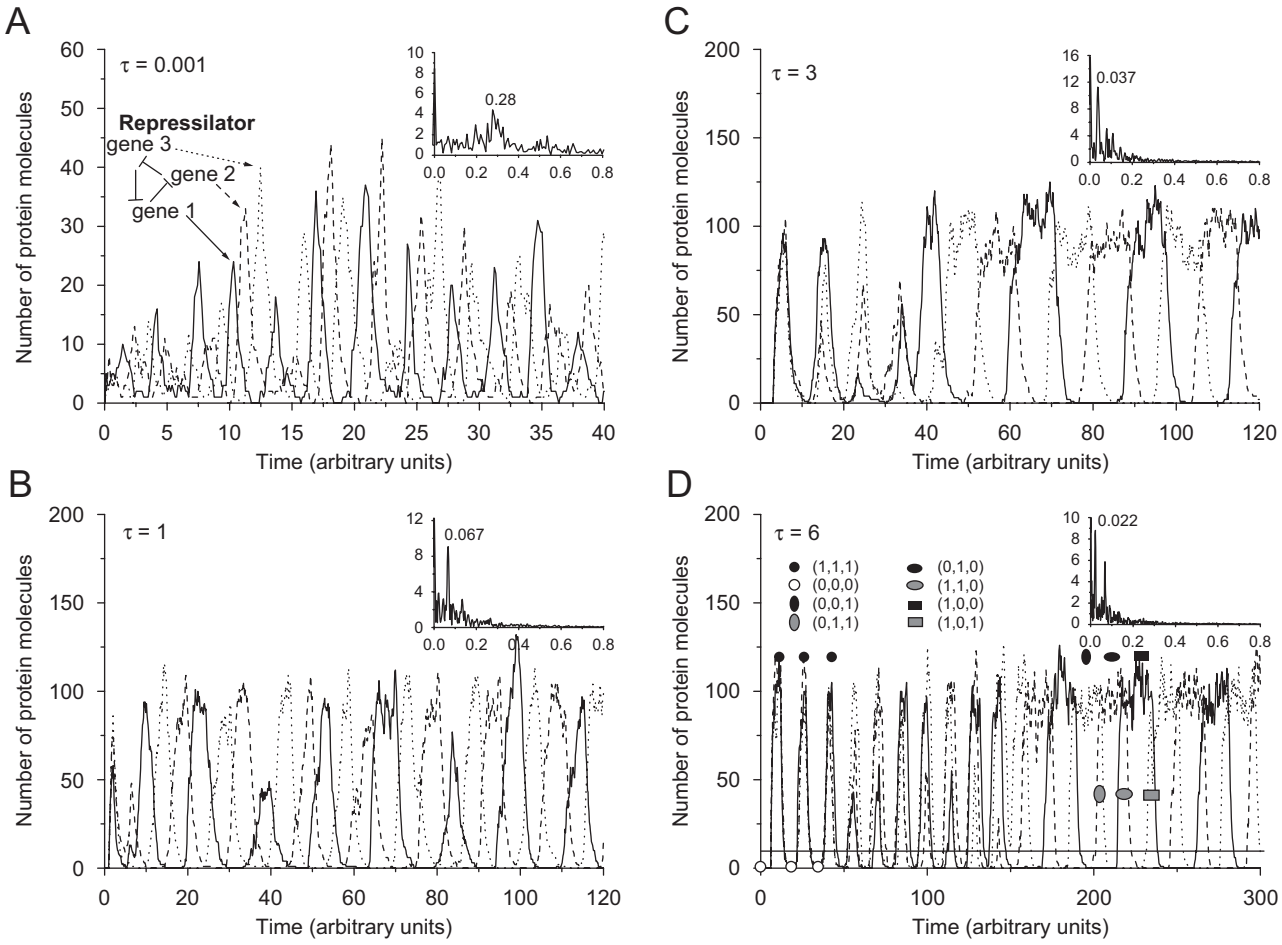

Fig. 7. Examples of protein time series simulated from the repressilator model. All the parameters are the same for each case except the protein synthesis delay τ which is specified on each panel. The binary state of the system is also illustrated in (D) as in Fig. 6 (F). See Table 3 for the details of the model and the other parameter values used. Insets show the FFT of the first repressor species. Numbers indicate the values of main frequencies.

Table 4
Boolean network models for the toggle switch and the repressilator

Toggle switch
Gene 1 -| Gene 2 -| Gene 1 (-| denotes repression)
Boolean functions and attractors

Input to gene 1	Output from gene 2	Input to gene 2	Output from gene 1
1	0	1	0
0	1	0	1
Attractor 1	(0, 0)→(1, 1)→(0, 0)…		
Attractor 2	(0, 1)→(0, 1)→(0, 1)…		
Attractor 3	(1, 0)→(1, 0)→(1, 0)…		

Repressilator
Gene 1 -| Gene 2 -| Gene 3 -| Gene 1

Boolean functions and attractors

Input to gene 1	Output from gene 2	Input to gene 2	Output from gene 3	Input to gene 3	Output from gene 1
1	0	1	0	1	0
0	1	0	1	0	1
Attractor 1	(0, 0, 0)→(1, 1, 1)→(0, 0, 0)…				
Attractor 2	(0, 0, 1)→(0, 1, 1)→(0, 1, 0)→(1, 1, 0)→(1, 0, 0)→(1, 0, 1)→(0, 0, 1)…				

switch provides the same three patterns (Table 4). For the repressilator, we can, using the same procedure, get the following binary code for (gene1, gene2, gene3) from the time series in Fig. 7D: (0, 0, 0)→(1, 1, 1)→(0, 0, 0)→(1, 1, 1)…(0, 0, 1)→(0, 1, 1)→(0, 1, 0)→(1, 1, 0)→(1, 0, 0)→(1, 0, 1)… . The extracting method is illustrated in Fig. 7D. Two distinct patterns are shown in this case. The corresponding Boolean network model gives the same two patterns (Table 4). It is worth noting that for the second repeating set of the repressilator, there are actually three different "1"s. This set can be refined to (0, 0, 1)→(0, ↑, ↓)→(0, 1, 0)→(↑, ↓, 0)→(1, 0, 0)→(↓, 0, ↑)→… "↓" means that the number of proteins is dropping down from a peak and "↑" means the number is growing from a valley. The states, "↓"and "↑", which are both viewed as "1" in the Boolean model, are two intermediate states between the above-defined steady states "1" and "0". This helps us understand that in the Boolean model, states (0, 1, 1), (1, 1, 0), and (1, 0, 1) are in fact, respectively, transient states to (0, 1, 0), (1, 0, 0), and (0, 0, 1).

The Boolean models capture in this case the correct path of states and all the attractors of the two stochastic systems illustrated above. The mean-field models not only can detect all the attractors but also can provide information on stability (robustness) of the attractors. Boolean models do not seem to be able to directly give the information of robustness for a given attractor. However, we can reasonably guess such information using the following method (Klemm and Bornholdt, 2005): if the number of states involved in one attractor (including all states in the basin of attraction) is much larger than that in the other, the former attractor may be more stable than the latter one, for example, the repressilator model. If the two attractors have similar number of states, like the toggle switch, we can use the minimal number of simultaneous flipping events to guess which is more stable (Klemm and Bornholdt, 2005). In stochastic situations, things get more complex. Figs. 6F and 7D are two special stochastic cases that can show the same discrete states as the Boolean models give. This agreement strongly depends on two factors: the implied Boolean functions in the stochastic models are faithful (unchanged) and the stochastic fluctuations are not strong.

For a Boolean network with faithful Boolean functions, the attractors are intrinsically deterministic even though the network exists in a stochastic situation. A deterministic attractor means that it includes deterministic states and each of the states only goes to a deterministic next state. In stochastic situations, such system may either jump from one attractor to another or stay on an attractor, depending on the strength of molecular fluctuations and the intrinsic robustness of the attractors (Figs. 6C–F and 7A–D, $b = 2$). If the Boolean functions are not faithful, we are dealing with probabilistic Boolean network (PBN) models (Shmulevich et al., 2002). These Boolean models have fuzzy attractors. In contrast to intrinsically deterministic attractors, fuzzy attractors are intrinsically noisy. We do not know exactly what states are involved in the attractor, and one state may go to different states in different situations. In this case, the molecular fluctuations would make things even fuzzier (Fig. 6A, $b = 0.5$). If the Boolean functions are just "so–so" in faith (a common PBN), one has an intermediate case between the two cases stated above. Fig. 6B with $b = 1$ illustrates such a case. We provided the examples above for the three different cases in term of the Hill coefficients used, since we found that the Hill coefficient, b, is an important faith index here. The higher the Hill coefficient, the more faithful are the implied Boolean functions in the stochastic models.

It is worth noting that the stochastic models will not show the synchronization of genes if the protein production delays are set to zero. The Boolean models do capture those special cases of synchronization, implying that Boolean models include some delay effects. The Boolean modeling method has shown some success to explore gene networks (Kauffman, 1993) as a first approach, disregarding its limitations. The key question is: are the implied Boolean functions in real genetic regulatory networks faithful? If so, how faithful are they? What are the dominant factors/mechanisms in controlling faithfulness? We can try to answer these questions by studying more complex models of genetic networks based on experimental data using the methods proposed here.

4. Conclusions

We present a general model of gene expression (reactions (1)–(3)) and its various simplified versions based on the delayed effective chemical reactions. The key feature of these reactions is that the complex multiple-step biochemical processes, such as transcription, translation, and even the whole gene expression, are simplified to single-step time delayed reactions. We verified our general gene expression model by showing that it correctly predicts the stochastic kinetics of a real gene expression in *E. coli* under the repressed condition. Using this model as a standard, those simplified versions were evaluated by comparing their kinetics. One of the simplified models is a commonly used gene expression model in the literature. Unfortunately, its average and stochastic kinetics are quite different from those of the standard model in the studied case. However, including two parameters, the protein burst size and the protein production delay, to the literature model would greatly improve its modeling performance.

We then proposed a strategy to construct simple genetic regulatory networks based on these gene expression models. In particular, we constructed a symmetrical genetic toggle switch model. We studied the conditions for bistability of the toggle switch. It was found that in order to build a robust toggle switch with fast switching time, the two genes must have strong inhibition ability, a high protein production rate, and short protein synthesis time. These conditions cannot be obtained from the mean-field model (Eqs. (18) and (19)). Especially, the stochastic model shows that strong inhibition enables bistability without

cooperative binding, which disagrees with the condition for bistability derived from the mean-field model, i.e. the cooperative binding of at least two proteins is required to get bistability. In addition, we find that the long protein production delay would cause an interesting long transient, where the two genes appear to synchronize before the system reaches a stable state.

Besides, we proposed a second method of constructing gene networks by using the corresponding mean-field models. Using this method, we built a second version of the toggle switch model. This model contains delayed virtual reactions which model the function of gene inhibition based on the Hill function. This deterministically related stochastic model gives kinetics similar to that of the first toggle switch model. We also constructed a three-gene oscillator model using this approach and studied its kinetics. The protein production delay is found to be a key parameter to control the oscillating behavior of the system.

Though the two methods are both intended to construct models of gene networks in the framework of stochastic reaction kinetics, they have quite different features. The first one is based on biochemical processes, while the second one is on biochemical functions. Therefore, the latter is at a higher level of modeling than the former. This means that fewer virtual reactions than effective reactions would be used to model the same genetic network. As a result, there are more details of biochemical processes in the former models than in the latter ones. Based on these features, the two methods can develop in two directions in further studies: the former is used for quantitatively studying simple gene networks, and the latter for systematically studying large-scale genetic regulatory networks.

Our gene expression model under the repressed condition captures detailed stochastic kinetics of the corresponding real system, demonstrating its ability to model gene expression quantitatively. This also displays the possibility of quantitatively modeling simple gene networks. To this end, there are at least two things we should focus on in future work: key biochemical reactions and their probability rate constants. This information now mainly comes from experiments. On the other hand, computational chemistry would be potentially another important tool. So far, the computational chemistry methods have not been widely used for this purpose. This is partly because most of the molecules involved in biochemical reactions are macromolecules which may consist of hundreds and thousands of atoms and also because most important interactions among biomolecules are weak interactions. However, the increasing power of computational chemistry might change this situation soon (Head-Gordon and Wooley, 2001).

The biochemical function-based stochastic models can be guided by the corresponding mean-field models since the two kinds of models are tightly related in dynamics. Our repressilator model is a good example. Also, the number of parameters of the model is much smaller than that of the biochemical process based model. This makes it easier to do the systematic analysis. In addition, our studies of the toggle switch and the repressilator indicate that this type of models can be connected with the corresponding Boolean models, which are easily analyzed due to their simplicity. This would make the systematic analysis much easier.

The introduction of delays into modeling makes it possible to efficiently study complex cellular processes at the molecular level based on simple chemical reactions. Naturally time delays mean intermediate processes. The simplification comes from omitting the dynamic details of the intermediate processes, keeping only the time they consume. This technique should prove very useful to study dynamical systems like gene networks, where the time dominantly controls the dynamics.

Acknowledgments

R.Z. would like to thank Dr. Josh Socolar for valuable discussions. This work was supported by a Grant (10174) from the Natural Sciences and Engineering Research Council of Canada. A.S.R. is funded by FCT, Portuguese Foundation for Science and Technology, with a Post Doctoral Fellowship (SFRH/BPD/14565/2003). All the authors thank iCORE, a funding agency of the Alberta government, and the Department of Physics and Astronomy of the University of Calgary.

References

Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., Walter, P., 2002. Molecular Biology of the Cell. Garland Publishing, New York.

Arkin, A., Ross, J., McAdams, H.H., 1998. Stochastic kinetic analysis of developmental pathway bifurcation in phage X-infected *Escherichia coli* cells. Genetics 149, 1633–1648.

Atkinson, M.R., Savageau, M.A., Myers, J.T., Ninfa, A.J., 2003. Development of genetic circuitry exhibiting toggle switch or oscillatory behavior in *E. coli*. Cell 113, 597–607.

Blake, W.J., Kærn, M., Cantor, C.R., Collins, J.J., 2003. Noise in eukaryotic gene expression. Nature 422, 633–637.

Bratsun, D., Volfson, D., Tsimring, L.S., Hasty, J., 2005. Delay-induced stochastic oscillations in gene regulation. Proc. Natl Acad. Sci. USA 102, 14593–14598.

Cai, L., Friedman, N., Xie, X.S., 2006. Stochastic protein expression in individual cells at the single molecule level. Nature 440, 358–362.

Chen, L.N., Aihara, K., 2002. Stability of genetic regulatory networks with time delay. IEEE Trans. Circuits—I 49, 602–608.

Cherry, J.L., Adler, F.R., 2000. How to make a biological switch. J. Theor. Biol. 203, 117–133.

Cowell, I.G., 1994. Repression versus activation in the control of gene-transcription. Trends Biochem. Sci. 19, 38–42.

Draper, D.E., 1996. In *Escherichia coli* and *Salmonella*. ASM Press, Washington, DC, pp. 902–908.

Elowitz, M.B., Leibler, S., 2000. A synthetic oscillatory network of transcriptional regulators. Nature 403, 335–338.

Gaffney, E.A., Monk, N., 2006. Gene expression time delays and Turing pattern formation systems. Bull. Math. Biol. 68, 99–130.

Gardner, T.S., Cantor, C.R., Collins, J.J., 2000. Construction of a genetic toggle switch in *Escherichia coli*. Nature 403, 339–342.

Gibson, M.A., 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.

Gillespie, D.T., 1977. Exact stochastic simulation of coupled chemical reactions. J. Phys. Chem. 81, 2340–2361.

Gillespie, D.T., 1992. A rigorous derivation of the chemical master equation. Physica A 188, 404–425.

Gillespie, D.T., 2001. Approximate accelerated stochastic simulation of chemically reacting systems. J. Chem. Phys. 115, 1716–1733.

Guido, N.J., Wang, X., Adalsteinsson, D., McMillen, D., Hasty, J., Cantor, C.R., Elston, T.C., Collins, J.J., 2006. A bottom-up approach to gene regulation. Nature 439, 856–860.

Head-Gordon, T., Wooley, J.C., 2001. Computational challenges in structural and functional genomics. IBM Syst. J. 40, 265–296.

Hooshangi, S., Thiberge, S., Weiss, R., 2005. Ultrasensitivity and noise propagation in a synthetic transcriptional cascade. Proc. Natl Acad. Sci. USA 102, 3581–3586.

Hou, Z.H., Xin, H.W., 2003. Internal noise stochastic resonance in a circadian clock system. J. Chem. Phys. 119, 11508–11512.

Kærn, M., Elston, T.C., Blake, W.J., Collins, J.J., 2005. Stochasticity in gene expression: from theories to phenotypes. Nat. Rev. Genet. 6, 451–464.

Kauffman, S., 2004. A proposal for using the ensemble approach to understand genetic regulatory networks. J. Theor. Biol. 230, 581–590.

Kauffman, S.A., 1969. Metabolic stability and epigenesis in randomly constructed genetic nets. J. Theor. Biol. 22, 437–467.

Kauffman, S.A., 1993. The Origins of Order: Self-organization and Selection in Evolution. Oxford University Press, New York.

Kaznessis, Y.N., 2006. Multi-scale models for gene network engineering. Chem. Eng. Sci. 61, 940–953.

Kennell, D., Riezman, H., 1977. Transcription and translation initiation frequencies of the *Escherichia coli lac* operon. J. Mol. Biol. 114, 1–21.

Kierzek, A.M., Zaim, J., Zielenkiewicz, P., 2001. The effect of transcription and translation Initiation frequencies on the stochastic fluctuations in prokaryotic gene expression. J. Biol. Chem. 276, 8165–8172.

Klemm, K., Bornholdt, S., 2005. Stable and unstable attractors in Boolean networks. Phys. Rev. E 72, 055101.

Kramer, B.P., Viretta, A.U., Baba, M.D.-E., Aubel, D., Weber, W., Fussenegger, M., 2004. An engineered epigenetic transgene switch in mammalian cells. Nat. Biotechnol. 22, 867–870.

Lewis, J., 2003. Autoinhibition with transcriptional delay: a simple mechanism for the zebrafish somitogenesis oscillator. Curr. Biol. 13, 1398–1408.

Li, Q.S., Zhu, R., 2004. Mesoscopic description of chemical supercritical Hopf bifurcation. Int. J. Bifurcat. Chaos 14, 2393–2397.

Lipshtat, A., Loinger, A., Balaban, N.Q., Biham, O., 2006. Genetic toggle switch without cooperative binding. Phys. Rev. Lett. 96, 188101.

MATLAB, 2005. The MathWorks.

McAdams, H.H., Arkin, A., 1997. Stochastic mechanisms in gene expression. Proc. Natl Acad. Sci. USA 94, 814–819.

McAdams, H.H., Arkin, A., 1998. Simulation of prokaryotic genetic circuits. Annu. Rev. Biophys. Biomol. Struct. 27, 199–224.

McClure, W.R., 1980. Rate-limiting steps in RNA chain initiation. Proc. Natl Acad. Sci. USA 77, 5634–5638.

Monk, N.A.M., 2003. Oscillatory expression of Hes1, p53, and NF-kB driven by transcriptional time delays. Curr. Biol. 13, 1409–1413.

Ozbudak, E.M., Thattai, M., Kurtser, I., Grossman, A.D., van Oudenaarden, A., 2002. Regulation of noise in the expression of a single gene. Nat. Genet. 31, 69–73.

Pedraza, J.M., van Oudenaarden, A., 2005. Noise propagation in gene networks. Science 307, 1965–1969.

Ptashne, M., Gann, A., 1997. Transcriptional activation by recruitment. Nature 386, 569–577.

Ramsey, S., Orrell, D., Bolouri, H., 2005. Dizzy: stochastic simulation of large-scale genetic regulatory networks. J. Bioinform. Comput. Biol. 3, 415–436.

Raser, J.M., O'Shea, E.K., 2005. Noise in gene expression: origins, consequences, and control. Science 309, 2010–2013.

Raser, J.M., O'Shea, E.K., 2004. Control of stochasticity in eukaryotic gene expression. Science 304, 1811–1814.

Record, T.M., Reznikoff, W.S., Craig, M.L., McQuade, K.L., Schlax, P.J., 1996. In *Escherichia coli* and *Salmonella*. ASM Press, Washington, DC, pp. 792–821.

Ribeiro, A.S., Zhu, R., Kauffman, S.A., 2006a. A general model for gene regulatory networks with stochastic dynamics. WSEAS Trans. Biol. Biomed. 3, 261–263.

Ribeiro, A.S., Zhu, R., Kauffman, S.A., 2006b. A general modeling strategy for gene regulatory networks with stochastic dynamics. J. Comput. Biol. 13, 1630–1639.

Rosenfeld, N., Young, J.W., Alon, U., Swain, P.S., Elowitz, M.B., 2005. Gene regulation at the single-cell level. Science 307, 1886–1888.

Roussel, M., Zhu, R., 2006. Validation of an algorithm for delay stochastic simulation of transcription and translation in prokaryotic gene expression. Phys. Biol. 3, 274–284.

Roussel, M.R., 1996. The use of delay differential equations in chemical kinetics. J. Phys. Chem. 100, 8323–8330.

Shmulevich, I., Dougherty, E.R., Kim, S., Zhang, W., 2002. Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks. Bioinformatics 18, 261–274.

Stryer, L., 1988. Biochemistry. W.H. Freeman and Company, New York.

Talkad, V., Schneider, E., Kennell, D., 1976. Evidence for variable rates of ribosome movement in *Escherichia coli*. J. Mol. Biol. 104, 299–303.

Turner, T.E., Schnell, S., Burrage, K., 2004. Stochastic approaches for modelling in vivo reactions. Comput. Biol. Chem. 28, 165–178.

Tuttle, L.M., Salis, H., Tomshine, J., Kaznessis, Y.N., 2005. Model-driven designs of an oscillating gene network. Biophys. J. 89, 3873–3883.

Veflingstad, S.R., Plahte, E., Monk, N.A.M., 2005. Effect of time delay on pattern formation: competition between homogenisation and patterning. Physica D 207, 254–271.

von Hippel, P.H., 1998. An integrated model of the transcription complex in elongation, termination, and editing. Science 281, 661–665.

Warren, P.B., ten Wolde, P.R., 2004. Enhancement of the stability of genetic switches by overlapping upstream regulatory domains. Phys. Rev. Lett. 92, 128101.

Warren, P.B., ten Wolde, P.R., 2005. Chemical models of genetic toggle switches. J. Phys. Chem. B 109, 6812–6823.

Yarchuk, O., Jacques, N., Guillerez, J., Dreyfus, M., 1992. Interdependence of translation, transcription and mRNA degradation in the *lacZ* gene. J. Mol. Biol. 226, 581–596.

Yu, J., Xiao, J., Ren, X., Lao, K., Xie, S., 2006. Probing gene expression in live cells, one protein molecule at a time. Science 311, 1600–1603.