UC Santa Barbara
UC Santa Barbara Previously Published Works

Title
Multiscale stochastic simulation algorithm with stochastic partial equilibrium assumption for chemically reacting systems

Permalink
https://escholarship.org/uc/item/6n16g21k

Journal
Journal of Computational Physics, 206(2)

ISSN
0021-9991

Authors
Cao, Y
Gillespie, D
Petzold, L

Publication Date
2005-07-01

Peer reviewed

eScholarship.org Powered by the California Digital Library
University of California

Multiscale Stochastic Simulation Algorithm with Stochastic Partial Equilibrium Assumption for Chemically Reacting Systems *

Yang Cao † Dan Gillespie ‡ Linda Petzold §

October 27, 2004

Abstract

In this paper we introduce a multiscale stochastic simulation algorithm (MSSA) which makes use of Gillespie's stochastic simulation algorithm (SSA) together with a new stochastic formulation of the partial equilibrium assumption (PEA). This method is much more efficient than SSA alone. It works even with a very small population of fast species. Implementation details are discussed, and an application to the modeling of the heat shock response of E. Coli is presented which demonstrates the excellent efficiency and accuracy obtained with the new method.

*This work was supported by the California Institute of Technology under DARPA Award No. F30602-01-2-0558, by the Molecular Sciences Institute under contract No. 244725 with the Sandia National Laboratories and the Department of Energy's "Genomes to Life" Program, by the U. S. Department of Energy under DOE award No. DE-FG02-04ER25621, by the National Science Foundation under NSF award CCF-0326576 and ACI00-86061, and by the Institute for Collaborative Biotechnologies through grant DAAD19-03-D-0004 from the U. S. Army Research Office.

†Department of Computer Science, University of California Santa Barbara, Santa Barbara, CA 93106, ycao@cs.ucsb.edu.

‡Dan T. Gillespie Consulting, GillespieDT@mailaps.org.

§Department of Computer Science, University of California Santa Barbara, Santa Barbara, CA 93106. petzold@engineering.ucsb.edu.

1 Introduction

In microscopic systems formed by living cells, the small number of reactant molecules can result in dynamical behavior that is discrete and stochastic rather than continuous and deterministic.[1–4] To study the influence of this stochastic behavior, stochastic simulation of the chemically reacting system is needed. Gillespie's stochastic simulation algorithm (SSA)[5,6] is well known as an essentially exact numerical simulation method for well-stirred chemically reacting systems and is widely used in the simulation of biochemical systems.

Because the SSA simulates every reaction event, it is inefficient for many realistic problems. The main reason for the low efficiency of the SSA is related to the multiscale nature of the underlying problem. Multiscale behavior appears in a wide range of problems. The multiscale problem in biochemical simulation has two aspects. The first is the timescale. Some reactions are much faster than others. Often the fast reactions quickly reach a stable state and the dynamics of the system are driven by the slow reactions. The SSA simulates every reaction and thus puts a great deal of effort into the more frequently occurring fast reactions, even though they do not contribute much to the dynamics and stochasticity of the system. This multiscale problem in time is known in the deterministic regime as *stiffness*.[7] Second, the populations of different species are of widely different magnitude. Some species are present with a large population while other species have only very few copies in a cell. Species with a small population should be modeled by a discrete stochastic process, whereas species with a large population can be efficiently modeled by a deterministic ordinary differential equation (ODE). SSA treats all of the species as discrete stochastic processes.

In this paper we use a model of the heat shock response (HSR) of E. Coli[8,9] as an example. The HSR system describes the mechanism of how the bacteria E. Coli responds to a temperature increase. When exposed to temperatures high enough to induce the denaturing (unfolding) of its constituent proteins, the *E. Coli* bacterium derives some measure

of protection from an elaborate heat shock response mechanism. One of several important components of this mechanism is the heat shock sigma factor, σ_{32}. Elevated temperatures in the bacterium cause σ_{32} to be produced through transcription and translation at a very rapid rate. A free σ_{32} molecule can bind to RNA polymerase (RNAP), and the resultant complex σ_{32}:RNAP initiates the transcription of genes that encode a variety of chaperone enzymes. These chaperones take care of denatured proteins, either by refolding them or else by degrading them so that they will not cause problems by aggregating. But a newly produced σ_{32} molecule is usually much more likely to be promptly sequestered by DNAK, one of those chaperone enzymes, an occurrence that precludes its binding to RNAP. The number of free σ_{32} molecules in the cell at any instant typically fluctuates over small (integer) values, with a time average that may well be less than one. Both multiscale aspects are present in this problem: (1) stiffness results from the presence of reactions which operate at vastly different timescales ($\sigma_{32}+DNAK \longleftrightarrow \sigma_{32}$:DNAK is very fast, whereas $\sigma_{32}+RNAP \longleftrightarrow \sigma_{32}$:RNAP is comparatively slow), and (2) there is a need to include in the simulation species that are present in very small quantities, for example free σ_{32} at typically around one molecule and bound σ_{32} at around 30-100 molecules, and species that are present with very large populations such as unfolded protein at more than 10^5 molecules. The details of the deterministic model for the HSR system can be found in Ref.[8] and a stochastic version was discussed in Ref.[9] In our stochastic model, 28 species participate in 61 chemical reactions. It takes 90 seconds for a 1.4Ghz Pentium IV Linux workstation to run one SSA simulation, which means more than 10 days for $10,000$ runs. (Usually we desire to run $10,000$ SSA simulations to obtain a reasonably accurate distribution.) The size of the HSR system is only moderate.

Important progress has been made towards efficient algorithms for discrete stochastic simulation. The tau-leaping method[10] has been proposed which can take time steps much larger than the time for a single reaction. By using a Poisson approximation, the tau-leaping

method can *"leap over"* many fast reactions and approximate the stochastic behavior of the system very well. The tau-leaping method makes a natural connection between the SSA in the discrete stochastic regime, the explicit Euler method for the chemical Langevin equation in the continuous stochastic regime, and the explicit Euler method for the reaction rate equation (RRE) in the continuous deterministic regime. The implicit tau leaping method[11] has been proposed to overcome the stability problems of the tau-leaping method when it is applied to stiff stochastic systems. The leaping methods show a promising direction to solve the multiscale stochastic problem. But they have a limitation, in that they are based on the assumption that the propensity function $a_j(x)$, defined so that $a_j(x)dt$ gives the probability that reaction R_j will fire in the next infinitesimal time interval $[t, t+dt)$, should not change significantly during each time step.[10] This requires the population of the species to be large relative to one. Thus when the system involves species with populations smaller than around ten, the leaping methods must be replaced by SSA. To the best of our knowledge, there has been no justifiable way to apply leaping methods to systems involving species that are present in very small quantities.

Another possibility is the hybrid methods,[12,13] which have recently been proposed to simulate multiscale chemically reacting systems. The hybrid methods combine the traditional deterministic reaction rate equation (RRE), or alternatively the chemical Langevin equation, with the SSA. The idea is to split the system into two regimes: the continuous regime and the discrete regime. RRE is used to simulate the fast reactions between species with large populations. SSA is used for slow reactions or species with small populations. The conditions[13] for the continuous regime are: *"the number of instances of each molecular species in a reaction in the continuous regime must be large relative to one, and the number of reaction events of each reaction occurring within one time step of the numerical solver must be large relative to one."* If either condition is not satisfied for a reaction channel, that

reaction channel must be handled in the discrete regime.

The hybrid methods efficiently utilize the multiscale properties of the problem. They are advantageous over the tau-leaping method when there are species with very small populations. But there are still some unsolved problems. One is that these methods do not have a rigorous theoretical foundation. Another important problem arises in the case when a reaction is fast but one of the corresponding reactants has a small population. It does not satisfy the second requirement for the deterministic regime. Thus this reaction will still be handled by the SSA, resulting in a very slow simulation. A typical such reaction is given by

$$S_1 + S_2 \longleftrightarrow S_3, \tag{1}$$

where S_1 has a small population but S_2 has a large population, and the reactions for both directions are fast. This situation occurs in the stochastic simulation of the HSR model, in which the important sigma factor σ_{32}, which plays a crucial role in the heat shock response of E. Coli, has a small population but is involved in fast reactions. If we must use SSA to simulate these fast reactions, the simulation will be very slow. Neither the tau-leaping methods or the hybrid simulation methods can efficiently handle this situation. The difficulty is that the reaction affects the population of S_1 significantly. The most extreme case is when the state of S_1 changes between zero and one. This actually occurs in the simulation of the HSR model. The important question is, how can we treat this reaction efficiently but maintain the stochasticity? Our MSSA method can handle this type of problem.

Quasi steady state and partial equilibrium assumptions have often been used in the simulation of deterministic kinetics systems.[14–16] In the deterministic case, the quasi steady state assumption assumes that on the time scale of interest, the instantaneous rates of change for some intermediate species are approximately equal to zero. The partial equilibrium assumption assumes that some fast reactions are always in equilibrium. In many cases these two assumptions are equivalent. The quasi-steady state assumption focuses on the state

while the partial equilibrium assumption concentrates on the reactions. These assumptions are often used to reduce the problem size or the stiffness.

The quasi steady state assumption was extended to the discrete stochastic case, and used together with SSA. The stochastic quasi steady state assumption (SQSSA)[17] was shown to be very useful in simplifying stochastic reaction kinetics. In this paper we will focus on the partial equilibrium assumption, although the discussion can also be applied to the quasi steady state assumption.

There is another practical problem with the hybrid methods and with the SQSSA method. Some molecular species are present in both the discrete and the continuous regimes, one in integer and one in floating point. In those methods, the floating point values from the deterministic regime are rounded to integers in the discrete regime. According to our analysis, this is not the best choice for the *"fast reaction low population"* situation. We propose to use the mean value. This does not lead to a big difference when the population is large. But if the population is very small, the difference between the floating point value and the integer value is relatively large and cannot be neglected.

The purpose of this paper is to introduce the main ideas of the stochastic partial equilibrium assumption and the MSSA algorithm, and to demonstrate the potential for accuracy and efficiency with this technique. Rigorous proofs have been omitted here. Interested readers can see the related paper[18] for a more rigorous discussion by probability theory. In contrast to the present paper, which concentrates on practical issues with regard the use of MSSA on moderate to large size real-world problems, Ref.[18] provides a theoretical analysis, focusing on several simple test problems that can be thoroughly understood. Thus the two papers are complementary.

Similar ideas have been proposed in the literature. A method called adiabatic elimination of fast variables,[19] has been studied in the Markov approximation for the reduced dynamics

in open systems. A theory is presented on the elimination of fast variables to obtain an effective description in terms of a reduced set of variables. A difference between that approach and ours is that the theory of adiabatic elimination of fast variables is framed in terms of continuous Markov processes (described by a Langevin and Fokker-Planck equation), not a jump Markov process (described by a master equation), which is the case we deal with.

It has been observed that *"it is the average occupancy over the individual microscopic events which determines the probability of gene expression "*[13]. Our paper presents one way to deal with those "average" characteristics, which can also be evaluated using thermodynamic properties, for example the entropy and Gibbs energy. Ackers et al.[20,21] developed a methodology for approximating statistical thermodynamic binding events, and applied it to simulation of the λ phage. Although the theoretical foundation for that method has not been carefully studied, in practice this approximation works well in reproducing the important properties of gene expression regulation. Our work is related to this approximating method. By generalizing the partial equilibrium assumption to the statistical thermodynamic assumption, our work can justify the use of the time-average effect from the formula of the propensities of the slow reaction channels. We further address this relationship in the discussion in Section 7.

The outline of this paper is as follows. In Section 2 we briefly review the SSA method. In Section 3 we introduce the stochastic partial equilibrium assumption (SPEA). In Section 4 we present the theory and the formulae to approximate the propensity functions of the slow reactions, based on SPEA. In Section 5 we discuss some implementation issues. Finally, in Section 6 numerical results are presented which demonstrate the accuracy and efficiency of this method. A conclusion and further discussion are given in Section 7.

2 Background

Suppose the system involves N molecular species $\{S_1, \ldots, S_N\}$. The state vector will be denoted by $X(t) = (X_1(t), \ldots, X_N(t))$, where $X_i(t)$ is the number of molecules of species S_i in the system at time t. M reaction channels $\{R_1, \ldots, R_M\}$ are involved in the system. We assume that the system is well-stirred and in thermal equilibrium. The dynamics of reaction channel R_j is characterized by the *propensity function* a_j and by the *state change vector* $\nu_j = (\nu_{1j}, \ldots, \nu_{Nj})$: $a_j(x)dt$ gives the probability that one R_j reaction will occur in the next infinitesimal time interval $[t, t+dt)$, and ν_{ij} gives the change in the S_i molecular population induced by one R_j reaction.

The dynamics of the system obeys the *chemical master equation* (CME):

$$\frac{\partial P(x,t|x_0,t_0)}{\partial t} = \sum_{j=1}^{M} [a_j(x-\nu_j)P(x-\nu_j,t|x_0,t_0) - a_j(x)P(x,t|x_0,t_0)], \tag{2}$$

where the function $P(x,t|x_0,t_0)$ denotes the probability that $X(t)$ will be x, given that $X(t_0) = x_0$. The CME is hard to solve, both theoretically and numerically. An equivalent simulation method is the SSA.[5,6] Let

$$a_0(x) = \sum_{j=1}^{M} a_j(x). \tag{3}$$

The time τ to the next occurring reaction is the exponentially distributed random variable with mean $1/a_0(x)$. The index j of that reaction is the integer random variable with probability $a_j(x)/a_0(x)$. SSA is a Monte Carlo method based on these distributions. In each step, SSA generates two random numbers r_1 and r_2 in $U(0,1)$, the uniform distribution on the interval $(0,1)$. The time for the next reaction to occur is given by $t+\tau$, where τ is given by

$$\tau = \frac{1}{a_0(t)} \log(\frac{1}{r_1}). \tag{4}$$

The index j for the next reaction is given by the smallest integer satisfying

$$\sum_{l=1}^{j} a_l(t) > r_2 a_0(t). \tag{5}$$

The system states are updated by $X(t+\tau) = X(t) + \nu_j$. Then the simulation proceeds to the next occurring time, until it reaches the final time.

3 Stochastic Partial Equilibrium Assumption

Traditionally the partial equilibrium assumption assumes that some fast reactions are always in equilibrium. In the deterministic regime, the equilibrium among the fast reactions is formulated as algebraic constraints. The involved states always satisfy these constraints. In the stochastic regime, even when the reactions are in equilibrium, the states may not satisfy the constraints. Instead, the *distributions* of these states are temporarily steady.

In order to distinguish the fast reaction channels and slow reaction channels, we re-label them so that $R = \left(R^f, R^s\right)$, where $R^f = \{R_1^f, \ldots, R_{M_f}^f\}$ represents the set of fast reaction channels and $R^s = \{R_1^s, \ldots, R_{M_s}^s\}$ represents the set of slow reaction channels, where $M_f + M_s = M$. We define a *fast species* to be any species whose population gets changed by some fast reaction. Otherwise it is called a *slow species*. We re-label the N species: $S = \left(S^f, S^s\right)$, where $S^f = \{S_1^f, \ldots, S_{N_f}^f\}$ is the set of fast species and $S^s = \{S_1^s, \ldots, S_{N_s}^s\}$ is the set of slow species, where $N_f + N_s = N$. The state vector is similarly subscripted as $X = \left(X^f, X^s\right)$. Note that the populations of fast species can be altered by slow reactions, but the populations of slow species cannot be altered by fast reactions. The fast and slow reaction propensity functions will similarly be denoted as

$$\begin{aligned} a_j^f(x) &= a_j^f(x^f, x^s), \quad j = 1, \ldots, M_f, \\ a_j^s(x) &= a_j^s(x^f, x^s), \quad j = 1, \ldots, M_s. \end{aligned} \tag{6}$$

The corresponding fast and slow reaction state-change vectors now appear as

$$\begin{aligned} \nu_j^f &= \left(\nu_{1j}^{ff}, \ldots, \nu_{N_f j}^{ff}\right), \quad j = 1, \ldots, M_f, \\ \nu_j^s &= \left(\nu_{1j}^{fs}, \ldots, \nu_{N_f j}^{fs}, \nu_{1j}^{ss}, \ldots, \nu_{N_s j}^{ss}\right), \quad j = 1, \ldots, M_s. \end{aligned} \tag{7}$$

Note that by definition, $\nu_{ij}^{sf} = 0$. We have dropped those zero components from ν^f.

Fast reactions occur much more frequently than slow reactions. To simplify the analysis we will first focus on the influence of fast reactions. We construct a *virtual system* V with only fast reactions R^f and fast species S^f. The populations of the fast species in the virtual system are represented by the virtual state variable $\hat{X}^f(t)$. By definition, $\hat{X}^f(t)$ is composed of the same fast state variables as $X^f(t)$ but it evolves under the influence of *only* the reactions R^f; in other words, $\hat{X}^f(t)$ is $X^f(t)$ with all the slow reaction channels *turned off*. $\hat{X}^f(t)$ is a more tractable process than $X(t)$ because it has fewer species and fewer reaction channels.

The reactions in the virtual system occur very fast. We assume that the *stochastic partial equilibrium* is quickly reached in the virtual system. This assumption involves two aspects. First, the virtual system has a *stochastic partial equilibrium* state. Note that the terminology *stochastic partial equilibrium* differs somewhat in meaning from partial equilibrium in the deterministic regime. On the one hand, it is *stochastic*. The virtual state variable $\hat{X}^f(t)$ keeps changing. But the distribution of $\hat{X}^f(t)$ remains unchanged. On the other hand, it is a *partial* equilibrium for the original system. Although the distributions of $\hat{X}^f(t)$ remain unchanged by the fast reactions, they can be changed by the occurrence of slow reactions. The reason is that slow variables are considered as parameters in the virtual system. Upon the occurence of a slow reaction event, slow variables get changed. The equilibrium state of $\hat{X}^f(t)$ is also disturbed. After that, the stochastic partial equilibrium is quickly resumed at a new level. We use $\hat{X}^f(\infty)$ to represent the fast variables which have unchanged distributions at the stochastic partial equilibrium state of the virtual system.

Second, compared with the occurrence time of the slow reactions, the transient time τ_{relax} (called the relaxation period) for the virtual system V to reach equilibrium is negligible. τ_{relax} is an important characteristic of the virtual system. In the virtual system, when the fast

reactions reach stochastic partial equilibrium, an important stochastic property is that after a relaxation period τ_{relax}, $\hat{X}^f(t)$ and $\hat{X}^f(t+\tau_{relax})$ can be considered as independent random variables with the same distribution as $\hat{X}^f(\infty)$. If τ_{relax} is negligible compared with the timescale of the slow reactions, the virtual system V can be considered to always remain at the stochastic equilibrium state. We then treat $\hat{X}^f(t)$ at different times t as independent random variables with the same distribution as $\hat{X}^f(\infty)$.

To illustrate the idea of stochastic partial equilibrium, Example 3.1 shows a simple virtual system and the corresponding stochastic equilibrium behavior.

Example 3.1 *A typical biochemical reaction has the form*

$$S_1 + S_2 \longleftrightarrow S_3, \tag{8}$$

with the corresponding propensity functions $a_1(X) = k_1 x_1 x_2$, $a_2(X) = k_{-1} x_3$, *where* $X_i(t)$ *is the state variable for the population of species* S_i. *We choose the coefficients* $k_1 = 1$, $k_{-1} = 10$ *and initial conditions* $x_1(0) = 10$, $x_2(0) = x_3(0) = 100$. *Suppose these two reaction channels are fast reactions. When* $a_1(X) \approx a_2(X)$, *the system is at the stochastic partial equilibrium state. Figure 1 shows the plot of* $X_1(t)$ *from time 0 to time 3 in one SSA simulation. The state keeps changing with time. Figure 2 shows the histogram of* X_1 *at time* $T = 1$ *(plot with 'o') and* $T = 2$ *(plot with '+') from an ensemble of* $10,000$ *SSA simulations. The distribution of the state remains the same at different times, which is also the distribution of* $\hat{X}_1(\infty)$. *Moreover, Figure 2 also shows the distribution of* $X_1(t_i)$ *with* $t_i = 0.01 * i$ *for* $i = 0, \ldots, 10000$ *(plot with '*'). It has the same distribution as* $\hat{X}_1(\infty)$.

4 Multiscale Stochastic Simulation Algorithm

Utilizing the stochastic partial equilibrium assumption, we can avoid detailed simulation of the fast reactions. Two methods can be applied. One is to find the distribution at the partial equilibrium and generate random numbers according to that distribution. For some simple systems, this method is rigorous and efficient.[18] But in the general case, finding the distribution for the partial equilibrium state is not easy. In this section we will show that, with the stochastic partial equilibrium assumption, only the mean value of the propensity functions are needed for the slow reaction channels.

Our purpose is to perform stochastic simulation on the slow reactions without an expensive simulation of the fast reactions. The critical step is to calculate the propensity functions for the slow reactions. If a slow reaction involves only slow species, the propensity function is calculated exactly the same as for the original SSA. But when a fast species is involved, how can we use the stochastic partial equilibrium assumption to calculate the corresponding propensity function? To solve this problem, we begin with a new formula for the propensity function for slow reaction channels when at least one of the reactants is a fast species.

Define $P_0^s(\tau|x^f, x^s, t)$ = the probability that, given $X(t) = (x^f, x^s)$, no slow reaction channel fires during $[t, t+\tau)$, and $a_0^s(X) = \sum_{j=1}^{M_s} a_j^s(X)$. During the time interval $[t, t+\tau)$ many fast reactions fire, which alters the fast variable X^f. We will not simulate the fast reactions. Instead we take X^f at time τ as a random variable. (Since no slow reaction fires during this period, $X^f(t)$ is the same as $\hat{X}^f(t)$.) According to the definition of a_0^s, under the condition that $X^f(\tau) = x^{f'}$, the probability that one slow reaction will occur in the next infinitesimal time interval $[\tau, \tau + d\tau)$ is $a_0^s(x^{f'}, x^s)d\tau$. Thus the total probability that one slow reaction will occur in the next infinitesimal time interval $[\tau, \tau + d\tau)$ is given by

$$E(a_0^s(X^f, x^s)|x^f, x^s)d\tau = \sum_{x^{f'}} P(X^f(\tau) = x^{f'}|x^f, x^s)a_0^s(x^{f'}, x^s)d\tau, \tag{9}$$

where $E(\cdot|x^f, x^s)$ and $P(\cdot|x^f, x^s)$ are the conditional mean and probability, under the condition that $X(t) = (x^f, x^s)$. Let the time $d\tau$ be so small that in the next infinitesimal time interval $[\tau, \tau + d\tau)$, at most one slow reaction can fire. Then we have

$$P_0^s(\tau + d\tau|x^f, x^s, t) = P_0^s(\tau|x^f, x^s, t)(1 - E\left(a_0^s(X^f(\tau), x^s)|x^f, x^s)d\tau\right). \tag{10}$$

Solving (10), we obtain

$$P_0^s(\tau|x^f, x^s, t) = \exp\left(-\int_t^{t+\tau} E\left(a_0^s(X^f(\mu), x^s)|x^f, x^s\right) d\mu\right). \tag{11}$$

We define the *next slow reaction density function* $p'(\tau, j|x^f, x^s, t)$. $p'(\tau, j|x^f, x^s, t)d\tau$ is the probability that, given $X(t) = (x^f, x^s)$, the next *slow* reaction will occur in the infinitesimal time interval $[t + \tau, t + \tau + d\tau)$ and will be an R_j^s reaction. Then (11) leads to

$$p'(\tau, j|x^f, x^s, t) = E\left(a_j^s(X^f(\tau), x^s)|x^f, x^s\right) \exp\left(-\int_t^{t+\tau} E\left(a_0^s(X^f(\tau), x^s)|x^f, x^s\right)\right). \tag{12}$$

Next we apply the stochastic partial equilibrium assumption. Assuming that the virtual system is at partial equilibrium and τ_{relax} is small compared to τ, $X^f(\tau)$ can be taken as $\hat{X}^f(\infty)$. Thus (12) becomes

$$p'(\tau, j|x^f, x^s, t) = E(a_j^s(X^f(\infty), x^s)|x^f, x^s) \exp\left(-\tau E(a_0^s(X^f(\infty), x^s)|x^f, x^s)\right). \tag{13}$$

Remark 4.1 (12) *is a general formula, which does not depend on the stochastic partial equilibrium assumption. It shows that the firing probability of the slow reactions is related to the mean value of the fast species, which can be approximated by a deterministic system if the corresponding fast species are present with a large population. Thus* (12) *provides a theoretical foundation for the hybrid methods.*[12,13] (13) *is the simplified formula after applying the stochastic partial equilibrium assumption. It is valid when τ_{relax} of the fast reactions is small compared to τ. When τ is close to τ_{relax}, (13) is biased. The magnitude of the bias is related to the variance of the fast species. The important point concerning (13)*

is that for a small population, we can use this method by taking the proper mean value. This allows us to generalize the work[12,13] *on hybrid methods to models where some species present with a small population are involved in fast reactions, which is exactly the case for the HSR model.*

The multiscale stochastic simulation algorithm (MSSA) is based on (13). Let $\bar{a}_j^s(x^s) = E\left(a_j^s(\hat{X}^f(\infty), x^s)|x^f, x^s\right)$, $\bar{a}_0^s(x^s) = \sum_{j=1}^{M_s} \bar{a}_j^s(x^s)$. Then (11) can be rewritten as

$$P_0(\tau|x^f, x^s, t) = \exp(-\bar{a}_0^s(x^s)\tau). \tag{14}$$

Thus the time τ that the next slow reaction will occur, starting at $X(t) = (x^f, x^s)$, follows an exponential distribution with probability density function $\bar{a}_0^s(x^s)\exp(-\bar{a}_0^s(x^s)\tau)$. We note that this is very similar to Gillespie's SSA method except that $a_0(x)$ is replaced by $\bar{a}_0^s(x^s)$. Once we obtain the value of $\bar{a}_0^s$, we can use the same Monte Carlo method to generate the random variable τ. For a uniform random number r, τ is given by

$$\tau = \frac{1}{\bar{a}_0^s(x^s(t))} \log\left(\frac{1}{r}\right). \tag{15}$$

The corresponding index j for the slow reaction channel which occurs at time $t + \tau$ is generated according to the distribution

$$p'(j|x_f, x_s, t) = \frac{\bar{a}_j^s(x^s)}{\bar{a}_0^s(x^s)}. \tag{16}$$

Thus the MSSA is given as follows. We will discuss the details of each step in the next section.

Multiscale Stochastic Simulation Algorithm

Given initial time t, initial state x_0 and final simulation time T,

Step 1: Compute the partial equilibrium state for the virtual system V. Update the fast variables $X^f = \hat{X}^f(\infty)$.

Step 2: For $j = 1, \ldots, M_s$, calculate $\bar{a}_j^s(x^s)$ and $\bar{a}_0^s(x^s)$.

Step 3: Generate two random numbers r_1 and r_2 in $U(0,1)$. The time for the next slow reaction to fire is given by $t + \tau$, where τ is given by

$$\tau = \frac{1}{\bar{a}_0^s(x^s)} \log\left(\frac{1}{r_1}\right). \tag{17}$$

The index j of the next slow reaction is given by the smallest integer satisfying

$$\sum_{j'=1}^{j} \bar{a}_{j'}^s(x^s) > r_2 \bar{a}_0^s(x^s). \tag{18}$$

Step 4: If $t + \tau > T$, stop. Otherwise update the time $t = t + \tau$ and states $x = x + \nu_j$. Go to Step 1.

5 Implementation of the MSSA method

5.1 Estimating $\bar{a}_j^s(x^s)$

Estimating the $\bar{a}_j^s(x^s)$ is the most difficult task in MSSA. Only for a very simple system can we derive an analytic solution for the distribution of $\hat{X}^f(t)$.[18] In the general case, we have to use an approximation. There are five possible functional forms for $\bar{a}_j^s(x^s)$:

$$\text{If } a_j^s(x) \text{ is independent of } x^f, \quad \text{then} \quad \bar{a}_j^s(x^s) = a_j^s(x). \tag{19}$$

$$\text{If } a_j^s(x) = c_j^s x_i^f, \quad \text{then} \quad \bar{a}_j^s(x^s) = c_j^s \left\langle \hat{X}_i^f \right\rangle. \tag{20}$$

$$\text{If } a_j^s(x) = c_j^s x_i^f x_{i'}^s, \quad \text{then} \quad \bar{a}_j^s(x^s) = c_j^s x_{i'}^s \left\langle \hat{X}_i^f \right\rangle. \tag{21}$$

$$\text{If } a_j^s(x) = \frac{c_j^s}{2} x_i^f (x_i^f - 1), \quad \text{then} \quad \bar{a}_j^s(x^s) = \frac{c_j^s}{2} \left\langle \hat{X}_i^f (\hat{X}_i^f - 1) \right\rangle. \tag{22}$$

$$\text{If } a_j^s(x) = c_j^s x_i^f x_{i'}^f, \quad \text{then} \quad \bar{a}_j^s(x^s) = c_j^s \left\langle \hat{X}_i^f \hat{X}_{i'}^f \right\rangle. \tag{23}$$

Here we use the notation

$$\left\langle f(\hat{X}^f) \right\rangle = E\left(f(\hat{X}^f(\infty)) | x^f, x^s\right) = \sum_{x^{f'}} f(x^{f'}) P\left(\hat{X}^f(\infty) = x^{f'} | x^f, x^s\right). \tag{24}$$

Note that $\left\langle \hat{X}^f(t) \right\rangle$ is defined under the conditional probability of $X(t) = (x^f, x^s)$. Since the (x^f, x^s) is stochastic, $\left\langle X^f(t) \right\rangle$ also has a distribution. But when $X(t) = (x^f, x^s)$ is fixed, $\left\langle \hat{X}^f(t) \right\rangle$ is a deterministic variable.

From all the possible forms, it is apparent that we need only the first two moments of $\hat{X}^f(\infty)$. But since the second moment is hard to calculate, if possible we will use only the mean value. For (19)-(21), we can directly use the mean for the virtual fast variable $\left\langle \hat{X}^f \right\rangle$. For (22) and (23), an approximation is needed. We will use the mean value for the virtual fast variable. For (22), we use

$$\bar{a}_j^s(x^s) \approx \frac{c_j^s}{2} \left\langle \hat{X}_i^f \right\rangle^2 . \tag{25}$$

This is for the reaction type $S_i^f + S_i^f \longrightarrow S_j$ (dimerization reaction). Note that we use $\left\langle \hat{X}_i^f \right\rangle^2$ instead of $\left\langle \hat{X}_i^f \right\rangle (\left\langle \hat{X}_i^f \right\rangle - 1)$. When the population of the fast variable S_i^f is large, this is a good approximation. Special attention must be paid when the population of S_i^f is very small. In particular the mean value can be less than one. Thus $\left\langle X_i^f \right\rangle (\left\langle X_i^f \right\rangle - 1)$ can be negative. There is no general way to efficiently compute $\bar{a}_j^s(x^s)$ in this case. In many practical problems, we can assume a Poisson distribution for X_i^f. Then $\left\langle [X_i^f]^2 \right\rangle = \left\langle X_i^f \right\rangle^2 + \left\langle X_i^f \right\rangle$. We have

$$\bar{a}_j^s(x^s) = \frac{c_j^s}{2} \left\langle \hat{X}_i^f (\hat{X}_i^f - 1) \right\rangle = \frac{c_j^s}{2} \left\langle \hat{X}_i^f \right\rangle^2 . \tag{26}$$

Thus we can use (25) to replace (22). Interestingly, this is the function that is used in the deterministic case. The resulting system is then very like the one given by the hybrid methods. But this Possion distribution assumption may not be always true. Further study is needed in that case. For (23), we use

$$\bar{a}_j^s(x^s) \approx c_j^s \left\langle \hat{X}_i^f \right\rangle \left\langle \hat{X}_{i'}^f \right\rangle . \tag{27}$$

This approximation is valid when at least one of the reactants has a large population. In the case where both reactants are fast species with small population and they are strongly

correlated, this approximation can lead to some error. So far we do not have an efficient way to deal with that case. Fortunately, this seems unlikely to occur, which has so far been bourne out in our experience with practical problems.

With (22) and (23) replaced by (25) and (27), we need only to estimate $\left\langle \hat{X}^f \right\rangle$. Algebraic equations are derived for $\left\langle \hat{X}^f \right\rangle$ according to the *equilibrium law* and *conservation law*. The *equilibrium law* requires the fast reactions to reach equilibrium. The *conservation law* applies to the virtual system. These two laws are usually obvious and are widely used in many applications. Note that now the equations are very similar to the equations in the deterministic regime. But the probability interpretation is different. The variables here are the means of $\hat{X}^f(\infty)$, not the concentrations for the state variables. Also, they are applied only to the virtual system, which usually has only mono-stable distributions[1]. Moreover, special attention should be paid if the fast reactions involve dimerization reactions. $S_1 + S_1 \longleftrightarrow S_2$, where the propensity functions are given by $a_1(x) = \frac{c_1}{2}x_1(x_1 - 1)$ and $a_2(x) = c_2 x_2$. The equilibrium law given by the deterministic equations is

$$\frac{c_1}{2}x_1^2 = c_2 x_2, \tag{28}$$

while the MSSA method requires

$$\frac{c_1}{2}x_1(x_1 - 1) = c_2 x_2. \tag{29}$$

When x_1 is large, (28) is not much different from (29). But when x_1 is small, (28) is no longer valid. We will see such an example in Section 6.

Many other technical details are also involved in the implementation of the MSSA method. In the following we will briefly discuss some important issues. We will use the

[1] This is a consequence of the assumption that τ_{relax} is small. For a bistable system, the relaxation time τ_{relax} will be large.

example from the stochastic simulation of the HSR model described earlier to illustrate the main points.

5.2 Solving the Partial Equilibrium Equations

5.2.1 Identifying the Fast Reaction Channels

How should we determine which are the fast reaction channels? In general, this remains an open question. For a biologist, those channels can be identified from experiments or experience. In the general case, the magnitude of the propensity function a can be used. But, for example, when one of the fast species switches between one and zero very rapidly, the corresponding propensity alternates between a large number and zero. When it is at zero, that reaction channel could be put into the group of slow reaction channels. Our practical approach is to run the full SSA simulation once or a few times and record the number of times that each reaction channel occurs. The most frequently occurring channels are the fast reaction channels. Determining how many fast reaction channels we should pick to guarantee the validity of the partial equilibrium assumption is a topic for future research.

There are 61 reaction channels and 28 species in the HSR model[2]. We ran the SSA once and recorded the number of times each reaction channel fired. 6 reaction channels were determined to be fast reactions. These 6 reaction channels fired 5.9×10^7 times, while the total number of firings for all the reaction channels was 6.2×10^7. The next most frequent reaction fired 7.7×10^5 times. Thus the 6 most frequently firing reaction channels take 94.4%

[2]We obtained the reaction channels and the corresponding parameters for this model from H. Kurata.[8] Many of the parameters were obtained from experiments, and are not known with a great deal of certainty.

of the total number of reaction firings. These 6 reactions are:

$$\sigma^{32} + HsLvu \longleftrightarrow \sigma^{32} : HsLvu, \tag{30}$$

$$\sigma^{32} + DNAK \longleftrightarrow \sigma^{32} : DNAK, \tag{31}$$

$$RNAP + D \longleftrightarrow RNAP : D, \tag{32}$$

where σ^{32} and $RNAP$ have small populations.

5.2.2 The Stochastic Partial Equilibrium Approximation

Once the fast reaction channels have been determined, the algebraic equations are given by the *equilibrium law* and the *conservation law*.

For the HSR model, eight species are involved in the six (three pairs) fast reaction channels. There are three equations which arise from the equilibrium law.

$$\begin{aligned} \langle \sigma^{32} \rangle \cdot \langle HsLvu \rangle &\approx K_1 \langle \sigma^{32} : HsLvu \rangle, \\ \langle \sigma^{32} \rangle \cdot \langle DNAK \rangle &\approx K_2 \langle \sigma^{32} : DNAK \rangle, \\ \langle RNAP \rangle \cdot \langle D \rangle &\approx K_3 \langle RNAP : D \rangle, \end{aligned} \tag{33}$$

where K_i are the ratios of the reaction rates (the backward reaction rates over the forward reaction rates) for the reactions (30-32). We have five equations from the conservation laws of the six reaction channels.

$$\begin{aligned} \langle \sigma^{32} \rangle + \langle \sigma^{32} : HsLvu \rangle + \langle \sigma^{32} : DNAK \rangle &= C_1, \\ \langle HsLvu \rangle + \langle \sigma^{32} : HsLvu \rangle &= C_2, \\ \langle DNAK \rangle + \langle \sigma^{32} : DNAK \rangle &= C_3, \\ \langle RNAP \rangle + \langle RNAP : D \rangle &= C_4, \\ \langle D \rangle + \langle RNAP : D \rangle &= C_5, \end{aligned} \tag{34}$$

where $C_i, i = 1, \ldots, 5$ are constants for the virtual system. They could be changed by the

slow reactions. The conservation law is valid only for the virtual system of the 6 fast reaction channels.

Equations (33) and (34) can be solved by Newton iteration. Because the virtual system is solved at almost every MSSA step, it is important to make the system size as small as possible. Since the conservation law yields linear equations, it is often used to reduce the size of the system. We chose $\langle DNAK \rangle$, $\langle \sigma^{32} \rangle$ and $\langle RNAP \rangle$ as free variables. Then

$$\begin{aligned}
\langle \sigma^{32} : DNAK \rangle &= C_3 - \langle DNAK \rangle \,, \\
\langle \sigma^{32} : HsLvu \rangle &= C_1 - C_3 + \langle DNAK \rangle - \langle \sigma^{32} \rangle \,, \\
\langle HsLvu \rangle &= C_2 - C_1 + C_3 - \langle DNAK \rangle + \langle \sigma^{32} \rangle \,, \\
\langle RNAP : D \rangle &= C_4 - \langle RNAP \rangle \,, \\
\langle D \rangle &= C_5 - C_4 + \langle RNAP \rangle \,.
\end{aligned} \tag{35}$$

Substituting (35) into (33), we obtained three nonlinear equations with three variables. Newton's method was then applied to this system. In the general case, the equations above can be automatically generated by symbolic computation.

5.2.3 Michaelis-Menten Approximation

If we have further knowledge of the system we may be able to simulate it more efficiently. As introduced in Section 1, the populations of the species have a multiscale nature. Based on a knowledge of the scales of the species populations, we can reduce the cost for solving the partial equilibrium equations. For example, for $\langle RNAP \rangle \cdot \langle D \rangle = K_3 \langle RNAP : D \rangle$, since we know that the population of D is several orders of magnitude larger than the free RNAP, we may treat $\langle D \rangle$ as a constant temporarily. Then from $\langle RNAP \rangle + \langle RNAP : D \rangle = C_4$ and $\langle RNAP \rangle \cdot \langle D \rangle = K_3 \langle RNAP : D \rangle$, we have

$$\langle RNAP \rangle = \frac{K_3}{K_3 + \langle D \rangle} C_4. \tag{36}$$

This is also known as the Michaelis-Menten[22,23] approximation. Then we can directly solve for $\langle RNAP \rangle$ without Newton iteration.

We note that this procedure requires prior knowledge about the scales of the system, so we will present numerical results both for using this approximation and not.

5.3 The Stochasticity of the Fast Species

In the MSSA method, after solving the algebraic equations of the virtual system we obtain $\left\langle \hat{X}^f \right\rangle$ instead of $X^f(t)$ itself for the fast species. As mentioned before, $\left\langle \hat{X}^f \right\rangle$ is affected by the slow reactions. For an ensemble of many MSSA simulations, $\left\langle \hat{X}^f \right\rangle$ will have a stochastic distribution. At this point, the distribution will not be accurate because it does not reflect the randomness of the fast reaction channels. But for the full SSA model, the stochasticity of the fast species is mostly affected by the slow reactions. Since the MSSA method captures the stochasticity of the slow reactions very well, for the fast species the distribution obtained from the MSSA method will be close to the distribution obtained from the SSA method. In most cases we are concerned only with the distributions of some slow species. The value we have for $\left\langle \hat{X}^f \right\rangle$ is good enough for that purpose. If in some special situations we need an accurate distribution of the fast species, we can use the *Down Shifting method*, which was proposed in Ref.[11] In the down shifting method, we use the MSSA method until the time is close to the output time. Then we change to the full SSA method for a short period of time (about twice the relaxation time τ_{relax}) just prior to the output time. This procedure regenerates the distribution for the fast species. The fact that the underlying kinetics is Markovian or "past-forgetting" is important in being able to apply such a procedure.

6 Numerical Examples

Numerical results for two examples are presented here to demonstrate the efficiency and accuracy of the MSSA method. The simulations were performed on a 1.4Ghz Pentium IV Linux workstation.

Example 6.1 *Heat Shock Response Model* As described in Section 5, we took 6 reaction channels as the fast reactions for the HSR model. These 6 most frequently occurring reaction channels account for 94.4% of the total number of reaction firings. With $T = 500$ as the final time, the original SSA simulation takes 90 seconds per run (9.00×10^6 seconds, or about 10 days for $10,000$ runs). Assuming the 6 reaction channels to be in stochastic partial equilibrium, we applied the MSSA method. We used Newton's method to solve the nonlinear equations, as described in Section 5.2.2. The average simulation time was 6.70 seconds for one MSSA simulation, yielding 6.70×10^4 seconds, or about 18 hours for $10,000$ runs. After further simplification using the Michaelis-Menton approximation, the time was 5.25 seconds for one MSSA simulation. We will present the accuracy comparison in the case of 12 fast reactions described below.

To obtain an even more efficient simulation of the HSR model, in the next series of numerical results we chose 12 reaction channels as the fast reactions. These 12 reactions fire 6.14×10^7 times, which is 99% of the total number of times for all reaction channels. Thus we can expect to reduce the simulation time to about 1% of the original simulation time. Conservation laws and equilibrium laws were used to construct the algebraic equations. The Michaelis-Menton approximation was also applied, based on our knowledge of the scales of the species. The MSSA method takes about 1 second per run ($10,000$ runs of the MSSA take about 3 hours). Note here that the time is very close to 1% of the original time. The overhead from solving the nonlinear algebraic equation is minimized by the Michaelis-Menton

approximation. In a further experiment, in order to obtain an accurate distribution for the fast species we applied the down shifting method. To do this, we used MSSA until $t = 499$ and then changed to the original SSA from $t = 499$ to 500. The MSSA method with down shifting takes 1.4 seconds per run ($10,000$ runs take about 4 hours).

The time comparison shows that the MSSA method is much more efficient than the original SSA method. To verify the accuracy of the MSSA method, we generated a histogram plot of $10,000$ runs for each method. We used 12 fast reactions for the MSSA method in the following comparison. For the slow species, MSSA always generates a histogram very close to the original SSA. For the fast species, MSSA without downshifting generates a close distribution but some difference is observed. With the down shifting method, this difference is no longer observed. Figure 3 shows the histogram for a slow species $mRNA(DNAK)$ and a fast species $DNAK$ with $10,000$ runs of the original SSA and the MSSA methods with and without down shifting. The distribution of the fast species is totally restored with the down shifting method.

Example 6.2 *Slow dimerization reaction of a fast species with a small population*

We consider

$$\begin{aligned} S_1 &\longleftrightarrow S_2, \\ S_1 + S_1 &\longrightarrow S_3, \end{aligned} \tag{37}$$

where the propensity functions are given by $a_1 = 200x_1$, $a_{-1} = x_2$ and $a_2 = x_1(x_1 - 1)$, $x_1(0) = 0$, $x_2(0) = 100$, $x_3(0) = 0$. Here the first two reversible reactions are the fast channels. The final time is $T = 20$. The algebraic equations for this simple system are given by

$$200\,\langle X_1\rangle = \langle X_2\rangle\,, \quad \langle X_1\rangle + \langle X_2\rangle = 100 - 2X_3(t). \tag{38}$$

Note that $X_3(t)$ is treated as a constant in the virtual system but its value changes with every occurrence of the slow reaction. Solving (38) gives $\langle X_1\rangle = \frac{100-2X_3(t)}{201} < 1$. Since $a_2 =$

$X_1(X_1 - 1)$, if we use the mean value of X_1 we can obtain a negative propensity. Assuming a Poisson distribution for X_1, we have $\langle a_2 \rangle = \langle X_1 \rangle^2$. Figure 4 shows the distribution of X_3 computed by the SSA method and the MSSA method with stochastic partial equilibrium assumption. They match very well. Figure 5 shows the distributions of X_2 generated by the original SSA and the hybrid SSA method with or without the down shifting method. We can see that with the down shifting method, the distribution of X_2 has been totally restored.

Example 6.3 *Fast dimerization: a case where the deterministic equation fails*

Consider

$$\begin{aligned} A &\longrightarrow S_1, \\ S_1 + S_1 &\longleftrightarrow S_2, \\ S_2 &\longrightarrow S_3, \end{aligned} \tag{39}$$

where A represents a species with constant population, the propensity functions for the reactions are given by $a_1 = 1$, $a_2 = 100x_1(x_1 - 1)$, $a_{-2} = 10x_2$ and $a_3 = x_2$, $x_1(0) = 0$, $x_2(0) = 0$, $x_3(0) = 0$. Here the reversible dimerizations $S_1 + S_1 \longleftrightarrow S_2$ are the fast channels. The final time is $T = 20$. The equilibrium law given by the deterministic equation is

$$x_2 = 10x_1^2, \tag{40}$$

while the MSSA requires

$$x_2 = 10x_1(x_1 - 1). \tag{41}$$

Equation (40) gives an incorrect value for x_2 when x_1 is small. The final distribution for S_3 is then wrong. As shown in Figure 6, (41) generates the correct distribution for S_3 while (40) doesnot. This example indicates that we have to be very careful when applying the deterministic equations to the low population case.

7 Conclusion and Further Discussion

In this paper we have introduced the MSSA method with the stochastic partial equilibrium assumption. This extends the hybrid methods[12,13] to the case where the population of the fast species may be small. Implementation details have been briefly discussed. The MSSA is a promising method for efficient and accurate simulation of stochastic biochemical systems.

In the current setup of this paper, to solve the stochastic partial equilibrium states we need a full list of fast reaction channels. But in many practical problems such a detailed list is not always available a priori. We note that the stochastic partial equilibrium assumption does not actually depend on the knowledge of a detailed list of fast reactions. When the possible states of the virtual system is finite, there is an alternative way to calculate the distribution of equilibrium states which applies the thermodynamic theory and uses a thermodynamic potential, such as the entropy or the Gibbs energy (see Ackers et al.[20,21]). Since the Gibbs energy may be determined through experiments, this can be a practical way to apply the stochastic partial equilibrium assuption. Our analysis provides a theoretical foundation for applying the thermodynamic method.

References

[1] A. Arkin, J. Ross, and H.H. McAdams. Stochastic kinetic analysis of developmental pathway bifurcation in phage λ-infected E. Coli cells. *Genetics*, 149:1633–1648, 1998.

[2] N. Fedoroff and W. Fontana. Small numbers of big molecules. *Science*, 297:1129–1131, 2002.

[3] H.H. McAdams and A. Arkin. Stochastic mechanisms in gene expression. *Proc. Natl. Acad. Sci. USA*, 94:814–819, 1997.

[4] H.H. McAdams and A. Arkin. It's a noisy business! *Trends in Genetics*, 15(2):65–69, 1999.

[5] D. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *J. Comput. Phys.*, 22:403–434, 1976.

[6] D. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 81:2340–61, 1977.

[7] Brenan, K.E., Campbell, S.L., and Petzold, L.R. *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations.* SIAM, Philadelphia, Pa, 1996 (second edition).

[8] H. Kurata, H. El-Samad, T. Yi, M. Khammash and J. Doyle. Feedback regulation of the heat shock response in E. Coli. *Proceedings of the 40th IEEE conference on Decision and Control*, pages 837–842, 2001.

[9] H. Kurata, M. Khammash and J. Doyle. Stochastic analysis of the heat shock response in E. Coli. *3rd International Conference on Systems Biology*, 2002.

[10] D. Gillespie. Approximate accelerated stochastic simulation of chemically reacting systems. *J. Chem. Phys.*, 115:1716, 2001.

[11] M. Rathinam, L. Petzold, Y. Cao, D. Gillespie. Stiffness in stochastic chemically reacting systems: the implicit tau-leaping method. *J. Chem. Phys.*, 119:12784–94, 2003.

[12] E. Haseltine and J. Rawlings. Approximate simulation of coupled fast and slow reactions for stochastic chemical kinetics. *J. Chem. Phys.*, 117:6959–6969, 2002.

[13] T. Mattheyses and M. Simmons. Hybrid simulation of cellular behavior. *Bioinformatics*, 20:316–22, 2004.

[14] J. D. Ramshaw. Partial chemical equilibrium in fluid dynamics. *Phys. Fluid*, 23, 1980.

[15] M. Rein. The partial-equilibrium approximation in reacting flows. *Phys. Fluids A*, 4, 1992.

[16] L. A. Segel and M. Slemrod. The quasi-steady-state assumption: a case study in perturbation. *SIAM Review*, 31, 1989.

[17] C. Rao and A. Arkin. Stochastic chemical Kinetics and the quasi steady-state assumption: application to the Gillespie algorithm. *J. Chem. Phys.*, 118:4999–5010, 2003.

[18] Y. Cao, D. Gillespie and L. Petzold. The slow-scale stochastic simulation algorithm. to appear, J. Chem. Phys.

[19] C. W. Gardiner. *Handbook of Stochastic Methods, for Physics, Chemistry and the Natual Sciences.* Springer, 2003 (third edition).

[20] G. Ackers, A. Johnson and M. Shea. Quantitative model for gene regulation by a lambda phage repressor. *Proc. Natl. Acad. Sci.*, 1982.

[21] M. Shea and G. Ackers. The o_r control system of bacteriophage lambda. *J. Mol. Biol.*, 1985.

[22] C. Lin and L. Segel. *Mathematics Applied to Deterministic Problems in the Natural Sciences.* SIAM, Philadelphia, 1988.

[23] J. M. Murray. *Mathematical Biology.* Springer-Verlag, Berlin, 1993, 2nd ed.

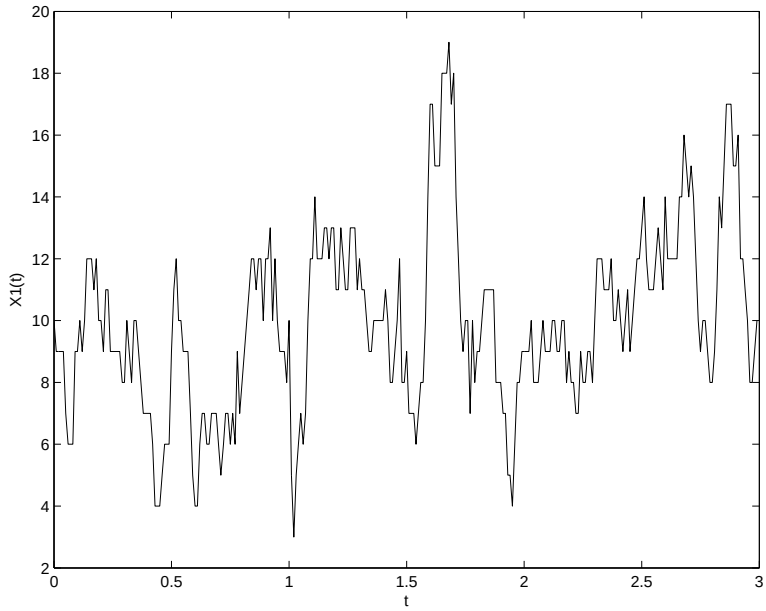

Figure 1: $X_1(t)$ for model (8) from time 0 to time T = 3 in one SSA simulation.

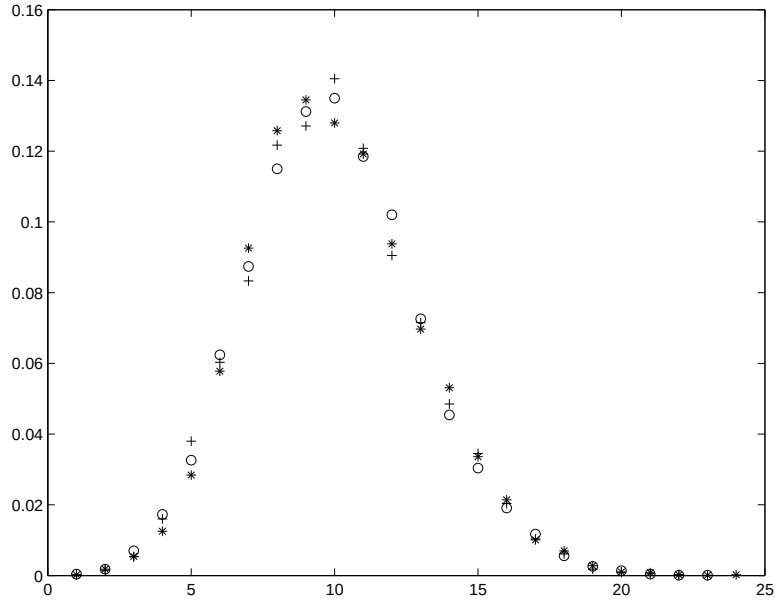

Figure 2: The histogram of X_1 for model (8) at time $T = 1$ (plot with 'o') and $T = 2$ (plot with '+') for $10,000$ SSA simulations. The plot with '*' is the distribution of $X_1(t_i)$ with $t_i = 0.01 * i$ for $i = 0, \ldots, 10000$.

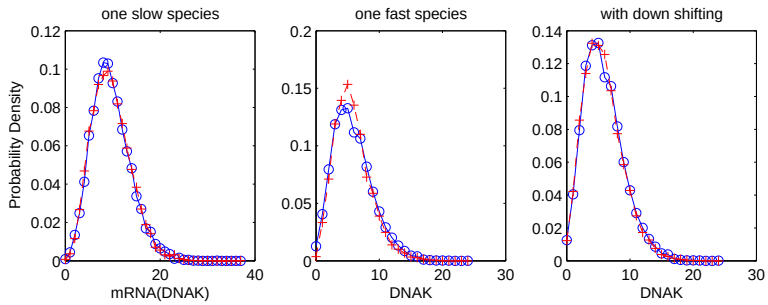

Figure 3: The histogram (10,000 samples) of a slow species mRNA(DNAK) (left) and a fast species DNAK with (right) and without down shifting method (middle) solved by the original SSA method (solid line with 'o') and the MSSA method (dashed line with '+') for the HSR model.

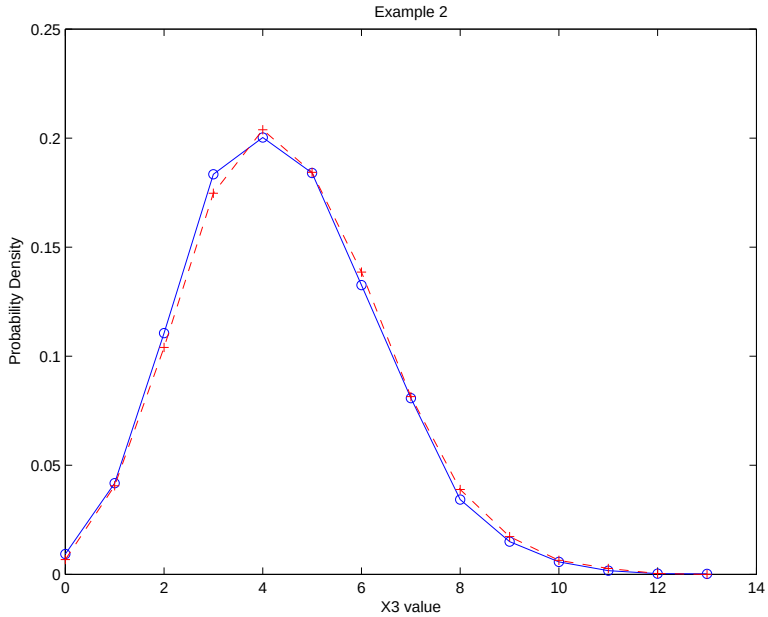

Figure 4: The histogram (10,000 samples) of X_3 solved by the SSA method (solid line) and the MSSA method (dashed line) for Example 2.

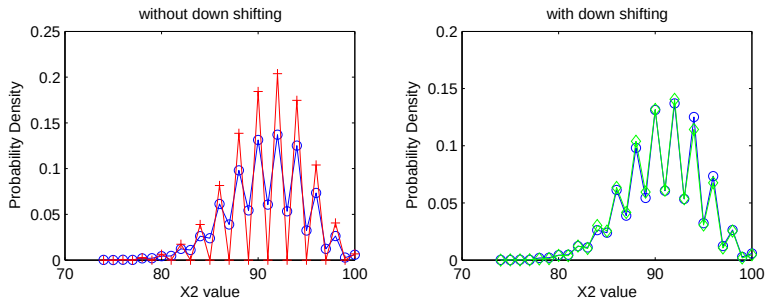

Figure 5: The histogram (10,000 samples) of X_2 solved by the SSA method (solid line) and the MSSA method with (right) and without (left) down shifting method for Example 2.

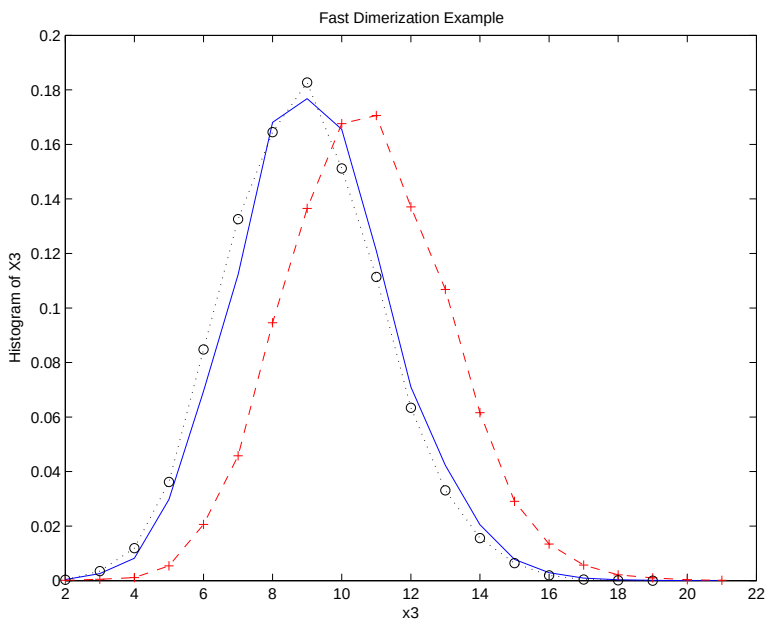

Figure 6: The histogram (10,000 samples) of S_2 solved by the SSA method (solid line), the formula (40) given by the deterministic equation (dashed line with '+') and the MSSA formula (41) (dotted line with 'o') for Example 3.